SESSION

ROBOTICS AND APPLICATIONS

Chair(s)

TBA

Place Recognition and Topological Map Learning in a Virtual Cognitive Robot

Paul R. Smart[1] **and Katia Sycara**[2]
[1]Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ UK
ps02v@ecs.soton.ac.uk
[2]Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA
katia@cs.cmu.edu

Abstract—*An ACT-R cognitive model is used to control the spatial behavior of a virtual robot that is embedded in a three-dimensional virtual environment, implemented using the Unity game engine. The environment features a simple maze that the robot is required to navigate. Communication between ACT-R and Unity is established using a network-based inter-operability framework. The ability of the robot to learn about the spatial structure of its environment and navigate to designated goal locations serves as a test of the ability of the framework to support the integrative use of cognitive architectures and virtual environments in a range of research and development contexts.*

Keywords: virtual robotics; virtual environment; cognitive architecture; spatial cognition; spatial memory

1. Introduction

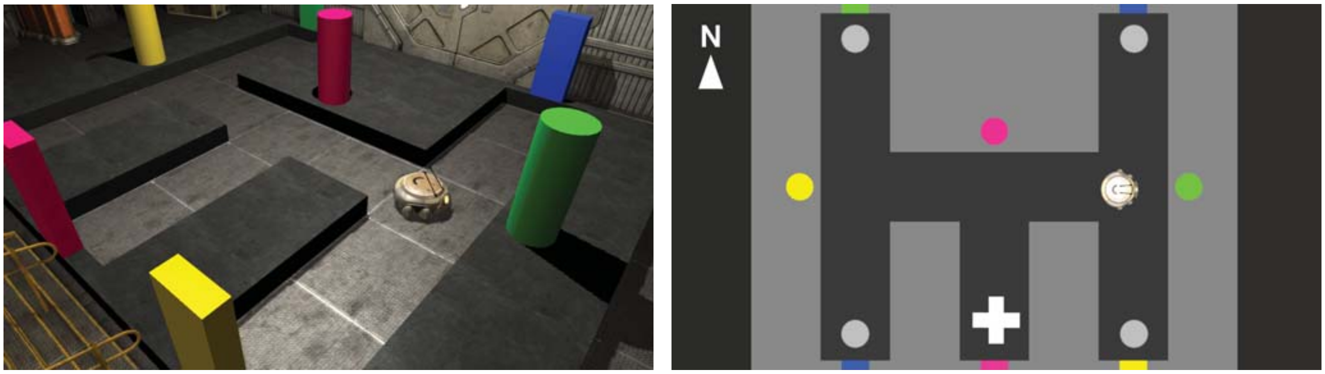
In the effort to develop computational models of cognitive behavior, it often helps to draw on the resources of a reusable framework that incorporates some of the representational structures and computational mechanisms that are assumed to be invariant across multiple cognitive tasks. Cognitive architectures are examples of such frameworks [1]. They can be used to develop models that test ideas relating to the cognitive mechanisms associated with aspects of human performance. In addition, they are sometimes used to implement agents that are capable of performing cognitive tasks or exhibiting signs of behavioral intelligence [2].

Work using cognitive architectures has typically involved the use of computational models that are highly limited with respect to the kinds of agent-world interaction they support. The perceptual inputs to cognitive models are typically very simple, as are the motor outputs. In addition, it is sometimes difficult to precisely simulate the effects that motor outputs have on subsequent sensory input, thereby limiting the extent to which cognitive models can productively incorporate motor actions into ongoing problem-solving processes (see [3]).

One way to address these issues involves the use of virtual environments, such as those encountered in contemporary video games. These can be used to create dynamic and perceptually-rich environments that serve as virtual surrogates of the real world. The cognitive models that are implemented using cognitive architectures can then be used to control the behavior of one or more virtual agents that inhabit these environments. By supporting the exchange of rich bodies of information between the virtual environment and the cognitive model, and by linking the cognitive model to the perceptuo-motor system of a particular virtual agent, it becomes possible to think of cognitive models as being effectively embedded and embodied within a virtual world.

In this paper, we describe a study that combines the use of a cognitive architecture with a virtual environment in order to study the maze learning and place recognition abilities of a virtual cognitive robot. The cognitive architecture used in the study is the ACT-R cognitive architecture [4]. This is one of the most widely used cognitive architectures within the cognitive scientific community. Although the design of ACT-R is inspired by the features of the human cognitive system (e.g., ACT-R consists of a number of modules that are associated with specific cognitive capabilities, such as the memorization and recall of declarative knowledge), it is possible to use ACT-R in the context of research efforts where the aim is not so much the modeling of human cognitive processes as the real-time control of a variety of intelligent systems. This is evidenced by recent work concerning the use of ACT-R in the design of real-world cognitive robots [5]. It is also possible to extend the core functionality of ACT-R in a variety of ways (e.g., via the addition of new modules).

Aside from ACT-R, the focus of the current integration effort is centered on the Unity game engine. This is a game engine, developed by Unity Technologies, that has been used to create a broad range of interactive 2D and 3D virtual environments. Despite its use as a research tool, as well as a platform for game development, the current attempt to integrate ACT-R with Unity is entirely novel: to our knowledge there have been no previous attempts to combine the use of Unity and ACT-R in the context of cognitive agent simulations. The closest approximation to the current integration effort is work by Best and Lebiere [6]. They used ACT-R to control the behavior of humanoid virtual characters in an environment implemented on top of the Unreal Tournament game engine. Our work differs from this previous work in the sense that we are targeting a different



(a) (b)

Fig. 1: Different views of the 'H' Maze environment. The robot is located on the right-hand side of the maze in both images. (a) View from a first-person camera situated external to the maze. (b) View from a top-down tracking camera situated directly above the maze. The tracking camera produces a simplified rendering of the scene in order to support the analysis and visualization of simulation results. The white cross represents the starting location of the robot on all training and testing trials. The grey circles within the maze represent goal locations for the robot during testing trials.

game engine (i.e., Unity) and we do not attempt to control a humanoid virtual character. Instead, we focus on a virtual robotic system that comes equipped with a set of (distinctly non-human) sensor and effector elements. These serve to make the perceptual and behavioral capabilities of the robot unlike those seen in the case of humanoid virtual characters. Another factor that differentiates our work from previous attempts to integrate cognitive architectures with virtual environments concerns our approach to sensor processing. While it is possible to rely on explicit knowledge about the features of virtual world objects (e.g., their position and geometry) as a means of directly calculating important perceptual information (e.g., distance and shape information), the robot in the current study is required to engage in the processing of low-level sensor data as a means of extracting cognitively-useful information from its environment.

As a means of testing the integrity of the ACT-R/Unity integration solution, we rely on the use of a spatial navigation task that requires an ability to (1) recognize spatial locations, (2) learn about the structure of a spatial environment and (3) navigate to specific goal locations. There are a number of factors that motivate the choice of this task in the context of the current work. Firstly, the topic of spatial cognition has been the focus of extensive research efforts in both the robotics and neuroscience communities [7], [8], [9]. This provides a wealth of data and knowledge that can be used to support the development of spatially-relevant cognitive models and associated cognitive processing capabilities. Secondly, spatial navigation is a task that is recognizably cognitive in nature, and it is one that may therefore benefit from the use of a cognitive architecture, such as ACT-R. In addition, the task is of sufficient complexity to require more than just the trivial involvement of the cognitive architecture. In fact, as will be seen below, the task requires the use of multiple existing ACT-R modules, the development of a new custom module, and the exploitation of over 100 production rules. Thirdly, the place recognition component of the task places demands on the perceptual processing capabilities of the robot. This helps to test the mechanisms used for the processing and interpretation of sensor data. Finally, the task requires the continuous real-time exchange of information between ACT-R and Unity in order to ensure that the behavior of the virtual robot is coordinated with respect to its local sensory environment. This serves as a test of the real-time information exchange capabilities of the proposed integration solution.

2. Method

2.1 Environment Design

A simple virtual maze was constructed from a combination of simple geometric shapes, such as blocks and cylinders. The design of the maze is based on that described by Barrera and Weitzenfeld [7] as part of their effort to evaluate bio-inspired spatial cognitive capabilities in a real-world robot. The maze consists of a number of vertically- and horizontally-aligned corridors that are shaped like the letter 'H'. An additional vertically-aligned corridor is used as a common departure point for the robot during training and testing trials (see Figure 1).

A number of brightly colored blocks and cylinders were placed around the walls of the maze to function as visual landmarks. These objects are used by the virtual robot to identify its location within the maze.

2.2 Virtual Robot

The virtual robot used in the current study is based on a pre-existing 3D model available as part of the Robot Lab project from Unity Technologies. The 3D structure of the virtual robot is defined by a conventional polygonal mesh of the sort typically used in game development. The robot was equipped with three types of sensors in order to support the processing of visual, tactile and directional information. Visual information is processed by the robot's eyes, which are implemented using Unity `Camera` components. For convenience, we refer to these components as 'eye cameras'. The eyes are positioned around the edge of the robot and are oriented at 0°, 90°, 180° and 270° relative to the Y or 'up' axis of the robot in the local coordinate system. This provides the robot with a view of the environment to its front, back, left and right. Given the elevated position of the visual landmarks in the maze (see Figure 1), the eyes were oriented slightly upwards at an angle of 15°. This enabled the robot to see the landmarks, even when it was positioned close to one of the walls of the maze.

In order to keep the visual processing routines as simple as possible, the eye cameras were configured so as to enhance the visibility of the visual landmarks within the scene. In particular, the far clipping plane of each eye camera's view frustum was set to 10 meters. This limited the range of the camera within the scene (although the range was still sufficient to encompass the entire extent of the 'H' Maze, irrespective of the robot's actual position in the maze). The culling mask of each eye camera was also configured so as to limit the rendering of scene objects that were external to the maze environment. Finally, a self-illuminated shader was used for the rendering of visual landmarks by the eye cameras. This shader used the alpha channel of a secondary texture to define the areas of the landmark that should emit light of a particular color. By simply omitting this secondary texture, the visual landmarks had the appearance of objects that emitted light uniformly across their surface. This served to enhance the contrast of the objects (from the robot's perspective) and reduced color variations resulting from different viewing angles. The result of applying these adjustments is shown in the four image insets at the top of Figure 2. These show the view of the maze environment from the perspective of each of the robot's eye cameras.

During the training and testing phases of the experiment (see Section 2.5), the output of each eye camera was periodically rendered to what is known as a `RenderTexture` asset. This is a special type of 2D image asset that captures a view of the virtual environment from the perspective of a particular camera. In essence, each `RenderTexture` asset effectively represents the state of one of the robot's 'retinas' at a particular point in time. The pixel data associated with these images can be processed in order to extract visual features, some of which may indicate the presence of particular objects in the scene. The visual processing routines used in the current study were relatively lightweight and focused on the attempt to detect the brightly colored objects (i.e., the visual landmarks) arrayed around the walls

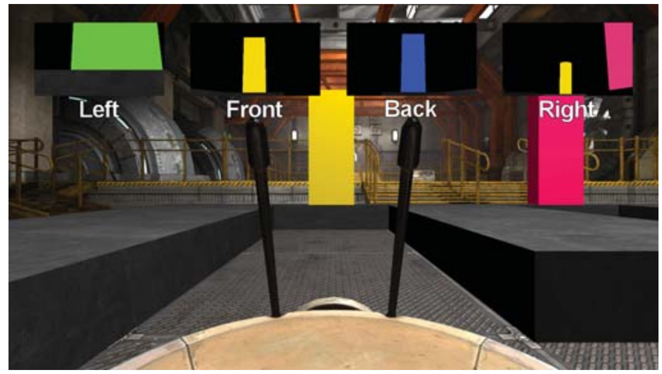


Fig. 2: View of the 'H' Maze from a forward-facing camera situated onboard the robot. The four image insets at the top of the image correspond to the views the robot has of the virtual environment via its eye cameras.

of the maze. These objects were detected by matching the luminance levels of image pixels in the red, green and blue (RGB) color channels to the colors of the objects as they appeared in the robot's eye cameras. A custom `RobotEye` component was developed to support the design-time configuration of the eye cameras with respect to the detection of the visual landmarks. This component supports the specification of target colors that should be detected by each eye camera during the post-rendering analysis of each `RenderTexture` asset. The component also provides access to two properties that control the sensitivity of the robot's eye cameras. These are the 'tolerance' and 'threshold' values. The tolerance value represents the range of luminance levels in each color channel that is recognized as a match to the target luminance level. A value of 0.01, for example, means that deviations of ± 0.01 from a target luminance level (in each color channel) will be recognized as a match to the target color[1]. The threshold value specifies the minimum number of matching pixels that must be present in the image in order for the `RobotEye` component to signal the detection of a particular color. For the purposes of the current study, the tolerance value was set to a value of 0.01 and the threshold value was set to a value of 1500. In addition, each retina was sized to 200×200 pixels to give a total of 40,000 pixels per eye camera on each render cycle.

In addition to the eye cameras, the robot was also equipped with 'whiskers' that functioned as tactile (or proximity) sensors. The aim of these sensors was to detect the presence of maze walls in the forward, left, right and backwards directions. The whiskers extended outwards from the robot's body in the same directions as the eye cameras and were of sufficient length to detect when the robot was adjacent to a maze wall. This enabled the robot to detect

[1] In Unity, the values of RGB channels range from 0 to 1, so a value of 0.01 represents 1% of the total value range.

the presence of particular situations, such as when it was in a corridor (e.g., the left and right whiskers were both in contact with maze walls) or when it had reached the end of one of the maze arms (e.g., the forward, left and right whiskers were all in contact with maze walls). The information provided by the whiskers assists in helping the robot to localize itself within the maze. The whiskers also function to provide affordances for action, helping the robot to decide when it needs to turn and what directions it can move in. From an implementation perspective, the whiskers were implemented using ray casting techniques: each time the robot was required to report sensory information to ACT-R, rays were projected from the robot's body and any collisions of the rays with the walls of the maze were recorded.

The final sensor used by the robot was a directional sensor. This functioned as an onboard compass. The sensor reading was based on the rotation of the robot's transform in the world coordinate system. A rotation of 0° thus corresponded to a heading value of 'NORTH'; a rotation of 90°, in contrast, corresponded to a heading value of 'EAST'.

For the purposes of this work, the directional movement of the robot was restricted to the north, south, east and west directions: these are the only directions that are needed to fully explore the 'H' Maze environment. The robot was also capable of making rotational movements to orient itself in the north, south, east, and west directions. Turning movements were implemented by progressively rotating the robot's transform across multiple update cycles using spherical linear interpolation techniques. Linear movements, in contrast, were implemented by specifying the velocity of the robot's `Rigidbody` component, a component that enabled the robot to participate in the physics calculations made by Unity's physics engine. Both movements occurred in response to the instructions received from an ACT-R model, and in the absence of this input, the robot was behaviorally quiescent.

2.3 Cognitive Modeling

The cognitive modeling effort involved the development of an ACT-R model that could support the initial exploration of the maze and the subsequent navigation to target locations. The requirements of the model were the following:

1) **Motor Control:** The model was required to issue motor instructions to the robot in response to sensory information in order to orient and move the robot within the maze.
2) **Maze Learning:** The model was required to detect novel locations within the maze and memorize the sensory information associated with these locations.
3) **Route Planning:** The model was required to use the memorized locations in order to construct a route to a target location.
4) **Maze Navigation:** The model was required to use route-related information in conjunction with sensory feedback in order to monitor its progress towards a target location.

In addition, in order to analyze the structure of the robot's spatial memories and compare navigational performance under different test conditions, it was important for the model to be able to serialize and deserialize memorized information to a persistent medium.

The ACT-R model developed for the current study consists of 126 production rules in addition to ancillary functions that control the communication with Unity (see Section 2.4). A key goal of the model is to memorize spatial locations that are distinguished with respect to their sensory properties (i.e., unique combinations of visual and tactile information). These locations are referred to as 'place fields' in the context of the model. Each place field is created as a chunk in ACT-R's declarative memory, and retrieval operations against declarative memory are used to recall the information encoded by the place field as the robot moves through the maze. The collection of place fields constitutes the robot's 'cognitive map' of the maze (see [9]). This map is structured as a directed graph in which the place fields act as nodes and the connections between the nodes are established based on the directional information that is recorded by the robot as it explores the maze. Any two place fields that are created in succession will be linked via a connection that records the direction the robot was moving in when the connection was made. For example, if the robot creates a place field (*PF1*) at the start of the simulation and then creates a second place field (*PF2*) while heading north from the start location, a connection will be established between *PF1* and *PF2* that records *PF1* as the source of the connection, *PF2* as the target of the connection and 'NORTH' as the direction of the connection. The cognitive map, as the term is used in the current study, is thus a representational structure that encodes information about the topological relationships between place fields based on the exploration-related movements of the virtual robot.

The productions of the ACT-R model were used to realize the motor control, maze learning and navigation functions mentioned above; the route planning function, however, was implemented using separate Lisp routines. In order to plan a route, the robot first needs to be given a target location. This was specified at the beginning of trials that tested navigational performance (see Section 2.5). The robot then needs to identify its current location within the maze. The robot achieved this by comparing current sensory information with that stored in memory (in the form of place field representations). Finally, the robot needs to compute a sequence of place fields that encode the path from the start location to the target location. This was achieved via the use of a spreading activation solution that operated over all the place fields in the robot's cognitive map (i.e., the contents of the robot's spatial memory). The spreading activation solution involved the initial activation of the place field corresponding to the robot's start location, and this activation was then propagated to neighboring place fields across successive processing cycles until the place field representing the target location was finally reached. The chain of activated place fields from the start location to the target location specifies the sequence of place fields (identified by combinations of sensory information) that must be detected by the robot as it navigates towards the target. Importantly, the connections between adjacent place fields in the computed route serves to inform the robot about the desired direction of travel as each place field is encountered. For example, if the connection between the first and second place fields in the route has an associated value of 'NORTH' and the robot is currently facing north, then the model can simply instruct the robot to move forward. If the robot is facing south, then the robot needs to implement a 180° turn before moving forward.

In order to avoid situations where the robot failed to detect successive place fields in the planned route (either as a result of delays in sensor feedback or the close proximity of topologically-adjacent place fields), the robot attempted to match received sensor information to *all* route-related place fields every time new sensor information was received. This enabled the robot to continually monitor its progress against the planned route and avoid confusion if some locations in the route were over-looked.

An initial pilot study using an earlier ACT-R model (see [10]) revealed a tendency for errors to sometimes occur in navigation-related decisions. Although this did not affect the ability of the robot in the pilot study to ultimately reach a particular goal destination, it did lead to inefficiencies in navigational behavior. An analysis of the structure of the robot's spatial memory in the context of this earlier study revealed that the problem originated from a failure to adequately discriminate between spatially-distinct locations during maze learning. Given the robot's perceptual capabilities, some of the locations in the maze can appear identical, and this can lead to situations where erroneous linkages are created between non-adjacent place fields. The result is a breakdown in the extent to which the cognitive map provides a faithful representation of the actual topological structure of the environment. In order to address this shortcoming, the current cognitive model attempted to categorize visual inputs based on the number of pixels of a particular color that were contained in the image generated by each eye camera. Pixel counts between 1500 and 6000 (for a particular color) were thus categorized as indicating the presence of 'small' colored objects, and pixel counts above 6000 were categorized as indicating the presence of 'large' colored objects[2]. The addition of this admittedly simple categorization scheme was sufficient to yield adequate discriminative capabilities in the context of the 'H' Maze; it is likely, however, that more refined schemes will be required in the case of more complex spatial environments.

[2]The detection threshold of the eye cameras was equal to 1500, so pixel counts below this value were treated as equal to zero.

2.4 ACT-R/Unity Integration Solution

In order for the ACT-R model to control the movements of the virtual robot in response to sensory information, it is necessary for the ACT-R environment and the Unity game engine to engage in bidirectional modes of communication. This is problematic because Unity is implemented in C++, while ACT-R is implemented in Lisp. In addition, the need to run Unity and ACT-R in parallel can place significant demands on the processing and memory resources of the host machine, and this can undermine the real-time responsiveness of both systems.

As a means of addressing these concerns, we developed a network-based solution to support the integration of ACT-R with the Unity game engine. The solution is based on an existing approach to integrating ACT-R with external environments that goes under the heading of the JSON Network Interface (JNI) [11]. The JNI enables ACT-R to exchange information with a variety of external environments using a combination of a TCP/IP connectivity solution and messages formatted using the JavaScript Object Notation (JSON) data interchange format. In order to make use of this approach in the context of environments built on top of the Unity game engine, we developed a set of components collectively referred to as the ACT-R Unity Interface Framework [10]. These components provide support for the automatic handling of connection requests made by ACT-R models using the JNI. They also enable Unity-based virtual characters to send information to specific ACT-R models and respond to ACT-R commands. The result is a generic solution for enabling ACT-R models to control the behavior of virtual characters in any Unity-based virtual environment (either 2D or 3D). By combining the framework with the JNI, we were able to run ACT-R and Unity on different machines (thus addressing performance issues) and establish bidirectional forms of communication between the two systems using a client-server model (with the ACT-R model acting as the client and Unity acting as the server). Further details of the integration solution can be found in Smart et al. [10].

At runtime, sensor information from the virtual environment was periodically posted to ACT-R as part of a 'sensor processing cycle'. For performance reasons, this was constrained to run at a frequency much lower than that of the game engine's main update loop (a frequency of 2Hz was used in the current study). During each sensor processing cycle, information from all of the robot's sensors was posted to ACT-R using a single JSON-formatted message. The ACT-R model received this information and responded to it by issuing motor commands that were posted back to Unity (again as JSON-formatted messages). These motor commands were themselves generated by a sequence of

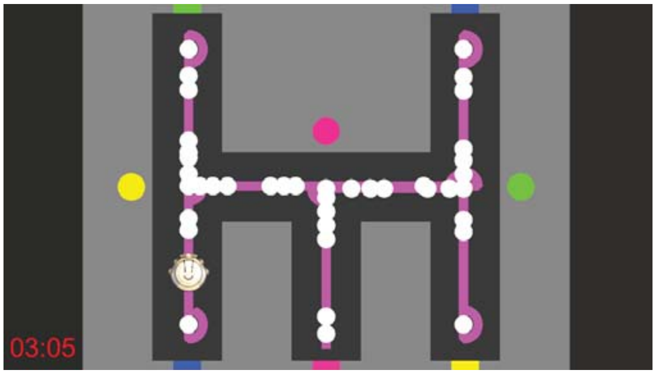


Fig. 3: A cognitive map of the environment formed during one of the training trials of the experiment. Each white circle symbolizes a place field that was created by the robot as it explored the maze. The place fields correspond to nodes in a topological map of the environment.

production firings corresponding to the cognitive processing steps implemented by the ACT-R model. On receipt of the motor commands, the Unity game engine dispatched the commands to the virtual robot, which then assumed responsibility for the actual implementation of motor actions.

2.5 Procedure

In order to test the integrity of the ACT-R/Unity integration solution, as well as the performance of the cognitive model, we performed a simple experiment involving a series of simulations. Each simulation consisted of two phases: a training phase and a testing phase. In the training phase, the robot was allowed to move around the maze and form a cognitive map based on its experiences. Once the robot had explored all of the maze, the training phase was terminated and the robot's cognitive map was saved to disk. In the subsequent testing phase, the cognitive map was loaded into declarative memory and the robot was given a series of target locations to navigate towards. These target locations were situated at the ends of each of the vertical corridors comprising the long arms of the 'H' Maze. The starting location of the robot was the same in all testing and training trials (see Figure 1b).

The simulation was repeated a total of five times in order to test the reliability of the model and the integrity of the ACT-R integration solution. This resulted in a total of five cognitive maps that were acquired on five separate training trials. It also resulted in data from (4 $\times$ 5) 20 testing trials that highlighted the navigational performance of the robot.

3. Results

The structure of one of the cognitive maps formed during one of the training phases of the experiment is shown in Figure 3. The white circles in this figure indicate the position of the place fields that were formed by the robot as it moved around the maze. The magenta trail represents the path of the robot and indicates the extent of the robot's exploratory activity.

Table 1: Table showing mean and standard deviation values for key dependent variables. Data was obtained from 5 simulations using identical conditions and parameters.

Dependent Variable	$\bar{X}$	σ
Training phase duration (seconds)	194.80	11.63
# place fields	41.00	2.45
# place field connections	44.60	3.13
Time to top-left target (seconds)	45.20	1.10
Time to bottom-left target (seconds)	45.40	1.52
Time to top-right target (seconds)	46.40	1.14
Time to bottom-right target (seconds)	46.40	1.95
# ACT-R messages (per minute) across all trials	123.66	1.78
# Unity messages (per minute) across all trials	274.07	4.69

Figure 4 shows the path followed by the robot as it navigated to one of the target locations (situated at the top left of the maze) in one of the testing phases of the experiment (the cognitive map, in this case, is the same as that shown in Figure 3). The robot was able to successfully navigate to each of the target locations in all test-related trials of the experiment. In addition, unlike the results that were obtained in an earlier pilot study (see [10]), the navigational performance of the robot was highly efficient, with no detours being made by the robot *en route* to the target locations. Table 1 summarizes some of the key results of the study. In addition, a video showing the behavior of the robot during the training and testing phases of the experiment is available for viewing from the YouTube website[3].

4. Conclusion

This study has shown how the ACT-R cognitive architecture can be used to control the behavior of a virtual robot that is embedded in a simulated 3D environment. A key aim of the study was to test the integration of ACT-R (which represents one of the most widely used cognitive architectures) with the Unity game engine (which represents one of the most widely used game creation systems). The integration solution builds on an existing approach to integrating ACT-R with external environments [11] in order to support bidirectional modes of information exchange between an ACT-R model and a Unity-based virtual environment. The two systems were hosted on separate machines during the course of the simulations, a strategy that serves to distribute the computational overhead associated with running both systems at the same time.

The task chosen to test the integration solution was a spatial navigation task that required an ability to learn about the spatial structure of a virtual 3D environment, recognize specific locations within the environment based

[3]See `http://youtu.be/IpoReu_PV3M`

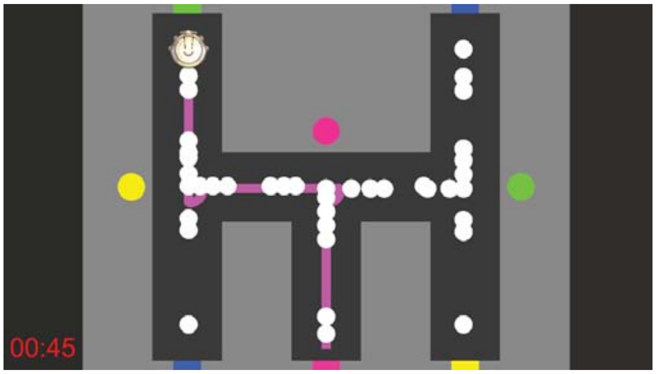


Fig. 4: The path taken by the robot to reach one of four target locations during one of the testing phases of the experiment.

on local perceptual information, and countenance behavioral responses based on a combination of local sensory cues, spatial knowledge and navigation-related goals. The ACT-R model developed to support these capabilities relied on a combination of visual, tactile and kinesthetic information in order to create memorial representations encoding the topological structure of the spatial environment. This approach resembles that seen in the case of real-world robotics research (e.g., [12]), and it is also consistent with the idea of visual and kinesthetic information being used to construct cognitive maps that subsequently guide the navigational behavior of a variety of animal species [9].

One extension of the current work could aim to improve our understanding of the cognitive mechanisms that are sufficient to yield adaptive navigational responses in other kinds of spatial environment. An important focus of attention, here, concerns the ability of virtual robots to exhibit navigational competence in the kinds of mazes that are typically encountered in bio-behavioral research (e.g., the radial-arm maze [13] and the Morris water maze [14]). This could establish the basis for cognitive models that attempt to emulate the spatial behavior of human and non-human subjects under specific test conditions.

Another potential target for future work concerns the enrichment of the cognitive representations used by the ACT-R model to support more sophisticated forms of spatial reasoning and behavioral control. One example here concerns the integration of metric information (e.g., information about angles and distances) into the topological map representation. Such information is deemed to be an important element of the spatial behavior of animals, and it is typically the focus of perceptual processing in the case of biologically-inspired robotic models of spatial navigation ability [8].

Future work could also aim to address some of the sensory and motor limitations of the robot used in the current study (recall the steps taken to simplify the visual processing of `RenderTexture` assets in Section 2.2). This includes work to improve the sophistication of visual processing capabilities, perhaps using techniques derived from computer vision research.

Finally, the availability of the current ACT-R/Unity integration solution opens up a range of relatively new research opportunities. One of these concerns the use of the integration solution to perform computational simulation studies that are relevant to current theoretical and empirical work in embodied, situated and extended cognitive science. Crucially, these simulations could serve as an important adjunct to studies that attempt to evaluate the role that environmentally-extended processing loops (and issues of material embodiment) play in the realization of human-level cognitive capabilities (see [3]).

References

[1] P. Thagard, "Cognitive architectures," in *The Cambridge Handbook of Cognitive Science*, K. Frankish and W. M. Ramsey, Eds. Cambridge, UK: Cambridge University Press, 2012, pp. 50–70.

[2] J. Rickel and W. Lewis Johnson, "Task-oriented collaboration with embodied agents in virtual worlds," in *Embodied Conversational Agents*, J. Cassell, J. Sullivan, and S. Prevost, Eds. Cambridge, Massachusetts, USA: MIT Press, 2000.

[3] A. Clark, *Supersizing the Mind: Embodiment, Action, and Cognitive Extension*. New York, New York, USA: Oxford University Press, 2008.

[4] J. R. Anderson, D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, and Y. Qin, "An integrated theory of the mind," *Psychological Review*, vol. 111, no. 4, pp. 1036–1060, 2004.

[5] U. Kurup and C. Lebiere, "What can cognitive architectures do for robotics?" *Biologically Inspired Cognitive Architectures*, vol. 2, pp. 88–99, 2012.

[6] B. J. Best and C. Lebiere, "Cognitive agents interacting in real and virtual worlds," in *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Interaction*, R. Sun, Ed. New York, New York, USA: Cambridge University Press, 2006.

[7] A. Barrera and A. Weitzenfeld, "Bio-inspired model of robot spatial cognition: Topological place recognition and target learning," in *International Symposium on Computational Intelligence in Robotics and Automation*, Jacksonville, Florida, USA, 2007.

[8] N. Burgess, J. G. Donnett, K. J. Jeffery, and J. O'Keefe, "Robotic and neuronal simulation of the hippocampus and rat navigation," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 352, no. 1360, pp. 1535–1543, 1997.

[9] B. Poucet, "Spatial cognitive maps in animals: New hypotheses on their structure and neural mechanisms," *Psychological Review*, vol. 100, no. 2, pp. 163–192, 1993.

[10] P. R. Smart, T. Scutt, K. Sycara, and N. R. Shadbolt, "Integrating ACT-R cognitive models with the Unity game engine," in *Integrating Cognitive Architectures into Virtual Character Design*, J. O. Turner, M. Nixon, U. Bernardet, and S. DiPaola, Eds. Hershey, Pennsylvania, USA: IGI Global, in press.

[11] R. M. Hope, M. J. Schoelles, and W. D. Gray, "Simplifying the interaction between cognitive models and task environments with the JSON network interface," *Behavior Research Methods*, vol. 46, no. 4, pp. 1007–1012, 2014.

[12] M. J. Mataric, "Navigating with a rat brain: A neurobioiogicaliy-inspired model," in *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behaviour*, J.-A. Meyer and S. W. Wilson, Eds. Boston, Massachusetts, USA: MIT Press, 1991.

[13] D. S. Olton and R. J. Samuelson, "Remembrance of places passed: Spatial memory in rats," *Journal of Experimental Psychology: Animal Behavior Processes*, vol. 2, no. 2, pp. 97–116, 1976.

[14] R. G. M. Morris, "Spatial localization does not require the presence of local cues," *Learning and Motivation*, vol. 12, no. 2, pp. 239–260, 1981.

A Novel Offline Path Planning Method

Pejman Kamkarian
Department of Electrical and Computer Engineering,
Southern Illinois University, Carbondale, IL 62901, USA

Henry Hexmoor
Department of Computer Science,
Southern Illinois University, Carbondale, IL 62901, USA

***Abstract*— This paper investigates a novel offline path planner for single point robots in cluttered environments. In order to achieve the best results in building shortest collision-free trajectory lengths from initial to the goal configuration, we considered a multi-layer solution in the form of a unique algorithm that works as a unit on workspace elements such as obstacles and determines proper trajectories. In addition, we have employed multiple parameters in our path planner to increase its level of flexibility to be able to maneuver on different scenarios related to the workspace components layouts. This versatility has increased our planner capabilities to override constraints related to diversity of environmental specifications as well as adjustment to the robot equipment constraints.**

***Keywords*— Navigation, Path planning, Rapidly Optimizing Mapper, Robot trajectory builder**

I. Introduction

An artificial intelligent point robot by default refers to a single mechanical device simplified to be represented by a point in space, which is able to maneuver through using proper equipment such as wheels in an environment called a workspace. In addition, a point robot is usually supplied with a predefined discipline, which enables it to operate in the workspace along with its related components independently. A workspace is usually bounded in a limited boundary area that consists of an initial and goal configurations as well as obstacles with various shapes and sizes located on different regions of the environment. One of the most important concerns for employing mobile robots is the ability to achieve the assigned tasks successfully without collisions. Unlike an online planner, an offline path planner outputs trajectories in an environment that is semi-static. The environment is composed of obstacles that remain stationary for long durations (e.g., as in a ship yard) and the robot path is expected to be traversed frequently. Therefore, path length and robot safety are of higher priority than impromptu path adjustments. In order to fulfill performing tasks flawlessly, a robot, as a key feature of its capability, has to possess a viable strategy to enable it to move among workspace objects while avoiding collisions. In other terms, the mobile robot has to be able to build an optimal collision-less trajectory from the initial to a prescribed goal configuration. A path planner for a point robot, hence, contemplates a viable trajectory, which is constructed in a form of a procedure that is responsible to control collisions while the robot is pursuing its goal. The more vigorous path planner that a single point robot has in terms of accuracy and planning, the higher rate of performance to reach the goal successfully in terms of collision avoidance and safety. In recent decades, the vital need of the planner construction for a robot has attracted several groups of researchers to work on developing and optimizing planners for purposes of achieving better and more accurate results to build trajectories for moving robots from a start point to the goal configuration successfully. Comparing the early planners with the later ones manifests drastic upgrades on both the accuracy and safety of path planners. Researchers have used a variety of different criteria and rules to construct robots path planners. At the early stages of developing path planners for offline robots [2], and [15] independently proposed planners inspired by electromagnetic fields. The Potential Field planner is constructed based on considerations of virtual attractive and repulsive forces among start and goal configurations along with the repulsive forces of obstacles that are present in the workspace. The Probabilistic Roadmap methods as another example of path planners was mainly introduced in [14]. It is categorized as a motion planning method that is built based on constructing a network of vertices located in the available free spaces in the workspace. They have adopted Dijkstra's algorithm [8] as a tool to analyze and refine the shortest trajectory from initial to the goal configuration. Some path planners are more focused on building a safe trajectory. For example, the Voronoi diagrams as a solution for planning have been studied and developed by several researchers [12], [16], [17], [19], [20], [23]. It basically works based on a set of vertices in the environment and builds the ideal trajectory considering the middle distances between workspace and obstacles as well as the distances between obstacles themselves. Employing this strategy allows robots equipped with lower accuracy rate of sensor detections to move between obstacles and reach their goals with the highest safety possibilities. While the earlier papers proposing the principle solutions of novel path planners, there is usually a departed change with later related approaches in terms of addressing issues and covering them. Therefore, later approaches achieved better results for building optimized trajectories. For instance, the local minima as one of the major disadvantages for using Potential Field planning method was addressed in [5], and [11]. Another issue that was resolved is the problem of using a single attraction point in workspace which leads to having difficulties with producing the resultant forces in environments including several closely located obstacles is reported in [18]. In order to fix the mentioned problems [3], and [9] have proposed modifications and upgrades on the Potential Field's originally constructed formulation by using different functions such as harmonic functions [7], [10], [24].

In order to reach higher optimization levels for path planners in terms of reliability, security, and trajectory length, several research articles from a variety of different groups of researchers have focused on incorporating many different approaches in form of hybrid solutions, with the sole intention of constructing more powerful planners with higher performances. Hybrid planning approaches are generally built based on a mixture of the key features that are adopting from different path planners in form of unique solutions with the purpose of treating problems that are involved with planners in calculating and constructing the ideal trajectories [1], [4], [6], [13], [21], [22], [25].

This research article introduces an offline path planner for single point robots that incorporate multi-layer strategies with the purpose of minimizing the trajectory length between initial to the goal configuration with respect to maintaining trajectory safety and saving the needed hardware resources to analyze the environment and achieve the ultimate trajectory in a reasonable time. Since the method rapidly optimizes for the path we coined it Rapidly Optimizing Mapper (ROM). In the next section we briefly explain the main parameters and components that are needed to construct our planner along with the key feature roles for each followed by illustrations of our novel offline path planner in the subsequent section.

II. Algorihmic formulation for ROM

In order to build an efficient path planner which is able to build collision-less trajectories for a mobile robot in a variety of different workspaces that are furnished to the planner with various obstacle arrangements, successfully, considering salient elements that play vital roles in robot navigation toward goal is essential. We categorized the most important elements that are necessary to be taken into account, while fabricating our adequate path planner in terms of building a collision-free and optimal trajectory, into two parts. Safety concerns and optimality consideration are our two major deliberations in terms of the shortest possible length between initial to the goal configurations. We will express these concepts in more details later in this section. We considered a three phase algorithm for our path planner in general, including: Processing the workspace, Graph transformation of the workspace, and Optimal trajectory determination. Figure 1 capitulates our planner phases along with steps that belong to each phase.

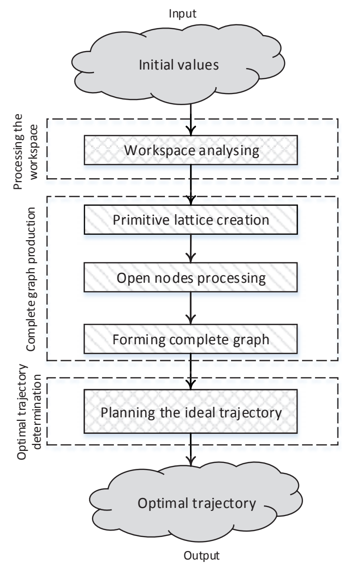


Figure 1. The general phases of our path planner

The planner algorithm computes the optimal trajectory based on the primitive adjustments for the related values that will be given at the beginning of the process of planning. These values empower the planner algorithm to consider the optimal trajectory based on the workspace and robot specifications. In other terms, equipping robots with different types of sensors having various detection capacities lead us to consider proper measurements in determining the trajectory to allow the robot to pursue a collision avoidance trajectory toward goal, successfully. The main objective for the first phase of our planner is to analyze the workspace elements such as obstacles with the purpose of determining and scanning perimeters of obstacles to locate their boundary edges and to provide necessary data for the next phase of planner. As it was discussed earlier in this section, in order for the planner to accomplish its task to build a flawless trajectory, we considered defining attributes that are directly incorporated in defining a safe and reliable path toward the goal. The safety property relies on considering proper boundaries around obstacles in regards to the robot's sensor equipment strength in terms of sensitivity to detecting and recognizing surrounding objects and also accuracy of detecting obstacle edges along with the robot's ability to maneuver in workspace via considering a proper distance to allow the mobile robot to adjust its direction toward following the computed trajectory. To fulfill the safety criterion, we defined Standoff Distance (SD) measurement. SD is an adjustable variable that is empirically determined based on the robot's sensor equipment specifications. In other words, the path planner algorithm computes the trajectory with consideration of the vertical length of SD value. The more mobile robot has poor obstacle edge detection as well as low degree of accuracy for detecting objects, the larger value for SD are applicable. An offline path planner by default analyzes the workspace and computes the proper trajectory prior to robot movement toward goal. Our planner builds the trajectory based on considering an abstract straight line from the initial to the goal configuration at each cycle of analyzing the workspace. In case where planner is confronted with crossing obstacles by the mentioned virtual straight line toward goal, the planner algorithm enters the phase of analyzing encountered obstacle accordingly. To fulfil obstacle inspection stage, the planner uses the Degree of Traverse (DT) concept. The DT value, quantifies the rate of obstacle surface scanning sensitivity. The DT value regulation depends on the workspace size along with obstacle primitives as well as the maneuvering skills rate of mobile robot. In some certain situations, the robot may need to move toward obstacle surfaces. When it comes to the skills of the robot in terms of safely maneuvering through obstacle primitive movements, considering a proper distance around obstacles as secure boundaries is essential. These boundaries have to be large enough to allow the robot to adjust its path toward the determined trajectory without collisions. The Degree of Surface Traversal (DST) is the measurement that we considered to achieve this objective. The value for the DST that is acceptable will vary depending on the distance that robot needs to adjust its direction successfully. The lower rate of accuracy for a robot to change its direction corresponds with considering higher value for DST. The algorithm steps of the workspace processing unit of the planner is illustrated with the following five step procedure:

1. The algorithm sets primitive variables related to the SD, DT, and DST, along with the start and goal configurations,
2. It then considers a virtual straight line from start point to the goal configuration,
3. The first obstacle that intersects with the virtual line in step 2 in at least one hit point will be categorized as a roadblock obstacle (RO),
4. For roadblock obstacles, the planner algorithm uses the analysis phase to scan the surface of the roadblock obstacles with the purpose of determining obstacle side edge nodes. In order to fulfil this aim, the planner adopts the value determined for the DT at the initialization phase. The initial angle that planner considers for obstacle surface examination is 0. At the beginning of the process, the algorithm inspects virtual paths in both sides of the obstacle hit point for the value resulting from accumulated of the latest angle and the DT value. If at least one of the virtual rays intersects the same obstacle in at least one hit point, the planner repeats the process of the roadblock obstacle surface scanning. In order to obtain the roadblock obstacle side edge nodes, the planner, at the beginning, considers the first virtual paths from both sides of the primitive hit point that shares no hit points with the same obstacle. It then reaches to the unsecure roadblock side edge nodes, which are calculated based on the nearest vertical distance from the surface of the obstacle. The planner appraises the roadblock side edge nodes by accumulating the unsecure roadblock side edge nodes distance and the SD value adjusted at the initiate phase of the planner. The last step of this phase of planner consists of considering the newly calculated roadblock side edge nodes as the new initial configurations that will be considered by the planner to route the trajectory from those points toward the goal, and
5. The process of analyzing workspace obstacles ends with reaching to the goal configuration. This unit of the planner supply the needed data for the next unit in order to revise and to form the complete graph in one connected component. The information produced at this unit includes the calculated roadblock side edge node points along with the initial start and goal configurations.

As it was outlined in the planner first unit procedure, the planner analyzes the workspace and recognizes and hence, selects those obstacles that are blocking the abstract straight paths from initial point toward goal configuration to be processed at the second unit of the planner algorithm. The major task of the second phase of our planner, as recently expressed, is to use the data produced in the first phase with the sole purpose of constructing a completed graph based on forming a lattice of nodes located on the edges of roadblock obstacles starting from primitive initial point and ending to the goal configuration. To fulfil the second stage of our planner, we conceived of the following three general attributes that the planner benefits to route an accurate and reliable complete graph toward goal.

Node visibility: Node visibility is the primitive condition that is met where pairs of nodes located on either side edges of roadblock obstacles or on their surfaces respectively, when a straight ray crossing from both nodes does not intersect any obstacles in the workspace. In other words, two nodes are considered to be visible, if there is a possibility to connect them through a straight line without intersecting any obstacles.

Visible pathways: A visible pathway consists of a group of visible nodes located consecutively on the same path. The following equation illustrates the expressed condition:

$$\forall \left[\left((\mathrm{n_i}, \mathrm{n_j}, \mathrm{n_k}) \in \mathrm{VN_s} \right) \&\& \left(\mathrm{VN_s} \in \mathrm{W_{O_s}} \right) \&\& \left(\mathrm{n_i n_j} \in \mathrm{C_x} \right) \&\& \left(\mathrm{n_i n_k} \in \mathrm{C_y} \right) \&\& \left(\mathrm{n_j n_k} \in \mathrm{C_z} \right) \right], \mathrm{if} \left[\left(\left(C_x \cap \bigcup_{f=1}^{p} W_{O_f} \right) = \emptyset \right) \&\& \left(\left(C_y \cap \bigcup_{f=1}^{p} W_{O_f} \right) = \emptyset \right) \&\& \left(\left(C_z \cap \bigcup_{f=1}^{p} W_{O_f} \right) = \emptyset \right) \right], then\, VP_t = \{\mathrm{n_i}, \mathrm{n_j}, \mathrm{n_k}\} \,\&\&\, (\mathrm{n_i}, \mathrm{n_k}) \in SVP_t \qquad (1)$$

In equation 1, $\mathrm{VN_s}$ indicates the visible nodes that belong to the obstacle $\mathrm{W_{O_s}}$. $\mathrm{C_x}$, $\mathrm{C_y}$, and $\mathrm{C_z}$ are illustrating connections between $\mathrm{n_i}$ and $\mathrm{n_j}$, $\mathrm{n_j}$ and $\mathrm{n_k}$, *and* $\mathrm{n_i}$ *and* $\mathrm{n_k}$, roadblock obstacle side edge nodes of $\mathrm{W_{O_s}}$, respectively. VP_t indicates the t^{th} path segment, which consists of $\mathrm{n_i}$ and $\mathrm{n_j}$ as its side points, SVP_t.

Isolated side nodes: The first phase of the planner task is to determine the roadblock obstacles side edge nodes. One of the major tasks of the second phase of the planner is to recognize roadblock obstacles side edge nodes belonging to the same obstacle that are not connected to one another through at least one correct direction toward the surface of the roadblock obstacle respectively. These nodes are considered to be isolated nodes.

The steps of the second phase of our planner is illustrated in the following four step procedure:

1. Identifying roadblock obstacles side edge nodes connection,
2. Eliminating uncompleted nodes generated by the analysis phase of the planner by connecting them together toward both directions of the surface of the obstacle with the vertical length of SD,
3. Simplification of nodes forming visible pathways by disposing the nodes located in between both sides of visible paths, and
4. Adjusting distances between pairs of nodes for the resultant graph which will be constructed from exploiting the previous steps of this phase of planner algorithm.

The second phase of our planner constructs a complete graph based on considering available paths from initial to goal configuration. This graph will be used in the subsequent phase of the planner algorithm with the purpose of recognizing optimal trajectory. In order to increase the performance of our planner in terms of building proper trajectory, we have benefitted from the application of the Dijkstra algorithm for the third phase of our planner. The main function of the last phase of our planner is to analyze the complete graph and hence, determining the shortest path toward goal configuration using the Dijsktra algorithm that we assume is common knowledge.

The next section is dedicated to validation of the functionality of our planner performance to determine optimal trajectories in workspaces consisting of various arrangements for initial and goal configurations along with obstacles specifications such as sizes, shapes, and locations.

III. VERIFICATION AND VALIDATION OF ROM

In order to validate the performance of our planner, we considered four different exemplar scenarios. These scenarios in diverse forms of various workspaces along with the results obtained from applying our planner to each are indicated in the following three case studies:

Case study 1:

For our first workspace map, we considered locating a variety of different obstacles in terms of shapes, connectivity, and locations in the form of crowded obstacle arrangements scenario to assess our planner performance on building trajectory in complicated situations. As shown in figure 2, some obstacles are connected to each other and hence, form larger obstacles while other obstacles are isolated from one another. The size of the workspace is 500 by 500 units. In order to maximize the number of roadblock obstacles from start to the goal configurations, we considered the start point at the upper left corner of the map at (20, 20) coordinates and the goal configuration at the lower right corner of the workspace at (378, 430) coordinates. The first workspace map along with the start and goal configurations are illustrated in the following figure 2.

Figure 2. The sample map of the workspace for the case study 1

Figure 3 reflects results from applying our path planner to the related workspace map.

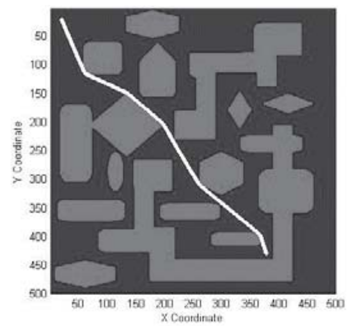


Figure 3. The first sample map of the workspace for the case study 1

The optimal pathway in the form of a collision-less trajectory is shown in bright path starting from start location and ending at the goal configuration. Figure 3 reveals that our path planner is able to build a collision-free trajectory in crowded workspaces in terms of obstacle numbers, sizes, shapes and arrangements, successfully.

Case study 2:

The second workspace consists of obstacles connected to each other to form a maze pattern shown in Figure 4.

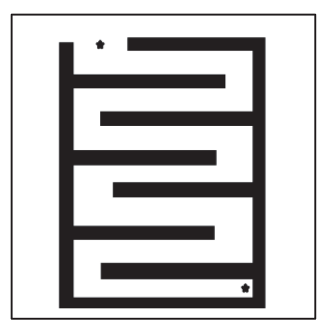


Figure 4. The second pattern of the workspace map for the case study 2

In this scenario, we considered the start point at (143, 50) coordinates and the goal configuration at (380, 452) coordinates. The following figure 5 shows the optimal path resulting from our planner operation on the related workspace illustrated in figure 4.

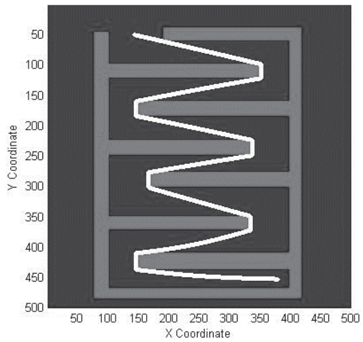


Figure 5. The first sample map of the workspace for the case study 2

As it shown in the figure 5, our planner is able to route the optimal trajectory benefiting different techniques in the form of a unique algorithm to calculate and refine the shortest path possible. This is done by adopting the simplifying skills to keep the side visible nodes and eliminate other nodes that are located in between. As a result, the final constructed graph consists of paths that are either in the form of straight lines that have the minimum distance between pairs of nodes or located on the surface of roadblock obstacles.

Case study 3:

The final considered case study is fabricated based on a workspace including spiral shapes obstacles. We used this scenario with the sole purpose of evaluating our planner skills to route shortest collision-less trajectory toward the surface of the roadblock obstacles. The following figure 6 is the showcase of the third environment including obstacles, start, and goal points.

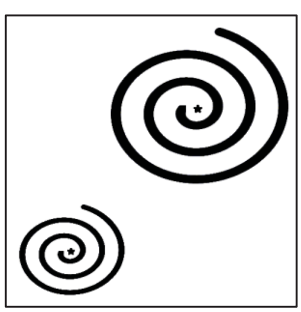


Figure 6. The third pattern of the workspace for the case study 3

The initial point is located at (110, 405) coordinates and the goal configuration is at (321, 160) coordinates. Our path planner generates the optimal trajectory in the form of the shortest collision-free path from start into goal configurations as indicated in the following figure 7.

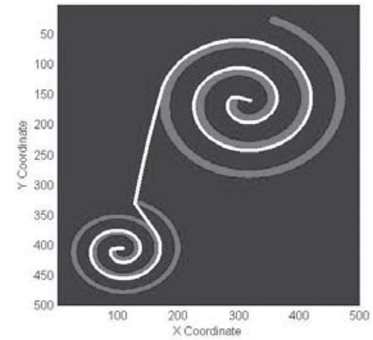


Figure 7. The first sample map of the workspace for the case study 3

The optimal trajectory is marked as a bright pathway crossing from the surface of the spiral obstacle. Our planner considered a portion of the optimal trajectory to be on the surface of the roadblock obstacle as indicated in figure 7. This is because the virtual ray crossing from roadblock obstacle side edge nodes were continuously intersecting the same obstacle and hence, a roadblock obstacle surface scanning enforced by the planner with the purpose of obtaining new side edge nodes toward the goal configuration.

The results obtained from applying our planner to all considered three cases revealed that our path planner is able to route optimal trajectory for all cases flawlessly. In addition, our planner is able to build the path without restriction to specific workspace arrangements. The way we fabricated our planner enables it to consider all possible workarounds to eliminate constraints, which elevate its performance to perform in any scenarios to analyze and refine trajectories from initial to the goal configurations.

IV. CONCLUSIONS

A novel offline path planner for single point robots has been presented that we dubbed ROM. ROM is able to plan a collision-free route from start point to the goal configuration. The planner benefits a series of different methods and parameters to analyze the workspace and compute the shortest collision-free trajectory toward goal. We increased the performance of the planner algorithm dramatically by considering techniques that overlook irrelevant obstacles from the workspace. The strategy of focusing on certain obstacles instead of all available obstacles in the environment leads to operation on fewer numbers of obstacles, and hence, reduces the required number of operations to analyze the workspace and to regulate the trajectory from start into goal configuration, more efficiently.

V. FUTURE WORK

Thus far, we focused on building ROM for moving offline robots that is able to analyze the workspace and construct an optimal trajectory in terms of computing the shortest length from initial points to the goal configuration, successfully. In addition, in regards to the safety of the processed path, our proposed planner is able to plan a collision-free route from start to the goal configuration. In order to accomplish the security aspect of determining the optimal trajectory, our path planner benefits from use of several parameters that we developed for our planner, as expressed in the previous sections. Throughout this research, we appraised our novel path planner performance by applying it on many prototypically complex situations with different workspace objects arrangements. Our planner revealed that it is able to route optimal collision avoidance trajectories in all cases, flawlessly. As the next phase of our planner performance evaluation in future work, we plan to assess our planner performance by comparing it with other offline path planners such as Potential Field and Rapidly-exploring Random Tree path planners. Our intent is to apply our planner along with other offline planners in different scenarios with the sole purpose of acknowledging the abilities of our path planner on operating workspace objects such as obstacles in terms of analyzing the environment as well as constructing the ideal collision-less trajectories.

REFERENCES

[1] Abinaya, S., Hemanth Kumar, V., Srinivasa Karthic, P., Tamilselvi, D., and Mercy Shalinie, S., “Hybrid Genetic Algorithm Approach for Mobile Robot Path Planning”, In Journal of Advances in Natural and Applied Sciences, Vol. 8, No. 17, Special 2014, pp. 41-47, 2014.

[2] Andrews, J.R. and Hogan, N., "Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator." Control of Manufacturing Processes and Robotic Systems, Eds. Hardt, D. E. and Book, W., ASME, Boston, pp. 243-251, 1983.

[3] Arámbula Cosío F. and Padilla Castañeda M.A., “Autonomous Robot Navigation using Adaptive Potential Fields”, In Mathematical and Computer Modelling, Vol. 40, Issue 9-10, pp. 1141-1156, 2004.

[4] Bashra K., and Oleiwi and Hubert Roth, “Modified Genetic Algorithm Based on A* Algorithm of Multi Objective Optimization for Path Planning”, In Journal of Automation and Control Engineering, Vol. 2, No. 4, pp. 357-362, 2014.

[5] Borenstein J. and Koren Y., “The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots”, In IEEE Transactions on Robotics and Automation 7, pp. 278-288, 1991.

[6] Chaari, I., Koubaa, A., Trigui, S., Bennaceur, H., Ammar, A., and Al-Shalfan, K., “SmartPATH: An Efficient Hybrid ACO-GA Algorithm for Solving the Global Path Planning Problem of Mobile Robots”, In International Jounal of Advanced Robotics Systems, pp. 1-15, 2014.

[7] Connolly C.I., Burns J.B., and Weiss R., “Path Planning using Laplace's Equation”, In Proceedings of the lEEE Conference On Robotics and Automation, Cincinnati, OH, pp. 2102-2106, 1990.

[8] Dijkstra, E.W., "A Note on Two Problems in Connexion with Graphs”, In Numerische Mathematik 1, pp. 269–271, 1959.

[9] Ge S.S. and Cui Y.J., “New Potential Functions for Mobile Robot Path Planning”, In IEEE Transactions on Robotics and Automation, Vol. 16, No. 5, pp. 615-620, 2000.

[10] Guldner J., Utkin V., and Hashimoto H., “Robot Obstacle Avoidance in N-Dimensional Space using Planar Harmonic Artificial Potential Fields”, In Journal of Dynamic Systems, Measurement, and Control, Vol. 119, pp. 160-166, 1997.

[11] Grefenstette J. and Schultz A.C., “An Evolutionary Approach to Learning in Robots”, In Proceedings of the Machine Learning Workshop on Robot Learning, International Conference on Robot Learning, New Brunswick, N.J., pp. 65-72, 1994.

[12] Hwang, F.K., “An O(nlogn) Algorithm for Rectilinear Minimal Spanning Trees”, In the journal of association of computer, pp. 177-182, 1979.

[13] Ju, M., Wang, S., and Guo, J., “Path Planning using a Hybrid Evolutionary Algorithm Based on Tree Structure Encoding”, In The Scientific World Journal, Vol. 2014, Article ID 746260, 8 pages, 2014.

[14] Kavraki, L.E., Svestka, P., Latombe, J.C., and Overmars, M.H., "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces", IEEE Transactions on Robotics and Automation Vol. 12, No. 4, pp. 566–580, 1996.

[15] Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots”, In IEEE International Conference on Robotics and Automation, St. Louis, Missouri, pp. 500-505, 1985.

[16] Kirkpatrick, D.G., “Efficient Computation on Continuous Skeletons”, In Proceedings of the 20th IEEE Symposium on Foundations of Computer Science, San Juan, pp. 18-27, 1979.

[17] Klein, R., “Abstract Voronoi Diagrams and Their Applications (Extended Abstract)”, In H. Noltemeier (ed.), Proceedings Computational Geometry and its Applications (CG "88), Wfirzburg, Lecture Notes in Computer Science, Vol. 333, Springer-erlag, Berlin, pp. 148-157, 1988.

[18] Koren Y., and Borenstein J., “Potential Field Methods and Their Inherent Limitations for mobile robot navigation”, In Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1398-1404, 1991.

[19] Lee, D.T. and Drysdale, R.L. III, “Generalization of Voronoi diagrams in the plane”, In Socity for Industrial and Applied Mathematics journal of computing, Vol. 10, Issue. 1, pp. 73-87, 1981.

[20] Lee, D.T. and Wong, C.K., “Voronoi Diagrams in $L_1(L_\infty)$ Metrics with 2-Dimensional Storage Applications”, In Socity for Industrial and Applied Mathematics Journal of Computing, Vol. 9, Issue. 1, pp. 200-211, 1980.

[21] Loo, C.K., Rajeswari, M., Wong, E.K., and Rao, M.V.C., “Mobile Robot Path Planning using Hybrid Genetic Algorithm and Traversability Vectors Method”, In Intelligent Automation and Soft Computing, Vol. 10, No. 1, pp. 51-64, 2004.

[22] McFetridge, L. and Ibrahim, M.Y., “New Technique of Mobile Robot Navigation using a Hybrid Adaptive Fuzzy-Potential Field Approach”, In Computers and Industrial Engineering, Vol. 35, Issues (3–4), pp. 471–474, 1998.

[23] Shamos, M.I. and Holy, D., “Closest-Point Problems”, In Proceedings of the 16th IEEE Symposium on Foundations of Computer Science, Berkeley, CA, pp. 151-162, 1975.

[24] Utkin V.I., Drakunov S., Hashimoto H., and Harashima F., “Robot Path Obstacle Avoidance Control via Sliding Mode Approach”, In Proceedings of the lEEE/RSJ International Workshop on Intelligent Robots and Systems, Osaka, Japan, pp. 1287-1290, 1991.

[25] Yao., Z. and Ren, Z., “Path Planning for Coalmine Rescue Robot based on Hybrid Adaptive Artificial Fish Swarm Algorithm”, In International Journal of Control and Automation, Vol. 7, No. 8, pp. 1-12, 2014.

Improve Robots Calibration Accuracy Using A Dynamic Online Interval Type-2 Fuzzy Interpolation System

Ying Bai
Dept of Computer Science and Engineering
Johnson C. Smith University
Charlotte, USA
ybai@jcsu.edu

Dali Wang
Dept of Physics and Computer Science
Christopher Newport University
Newport News, USA
dwang@pcs.cnu.edu

***Abstract*—This paper describes a novel technique for the position error compensations of the robot and manipulator calibration process based on an Interval Type-2 Fuzzy error interpolation (IT2FEI) method. Traditional robots calibration implements either model or modeless method. The compensation of position error in modeless method is to move the robot's end-effector to a target position in the robot workspace, and to find the target position error based on the measured neighboring 4-point errors around the target position. A camera or other measurement device is attached on the robot's end-effector to find and measure the neighboring position errors, and compensate the target position with the error interpolation results. By using the IT2FEI technique provided in this paper, the accuracy of the position error compensation can be greatly improved, which has been confirmed by the simulation results given in this paper. Compared with some other popular traditional interpolation methods, this IT2FEI technique is a better choice. The simulation results show that more accurate compensation result can be achieved using this technique compared with the type-1 fuzzy interpolation method.**

Keywords*—*Modeless robots calibrations; position error compensations; interval type-2 fuzzy interpolations; dynamic online fuzzy interpolation algorithm.

I. INTRODUCTION

The prerequisite requirement of the robotic modeless calibration is the successful self-calibration of the camera [1, 2]. Both internal and external parameters of the camera need to be calibrated accurately [3, 4]. Then modeless robot calibration is divided into two steps [5]. The first step is to identify the position errors for all grid points on a standard calibration board, which is installed on the robot's workspace. A calibrated camera or other measurement device is attached on the robot's end-effector to find the neighboring position errors. This process can be considered as an identification process, which is shown in Fig. 1.

At each grid point, a calibrated camera is used to check the position errors of the end-effector of the robot. In Fig 1, the desired position of the grid point 0 is (x_0, y_0), and the actual position of the robot end-effector is (x'_0, y'_0). The position errors for this grid point are $e_x = x_0 - x'_0$, and $e_y = y_0 - y'_0$.

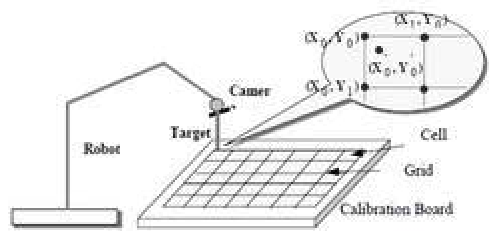


Fig.1 Setup of the robotic modeless calibration.

The robot will be moved to all grid points on the standard calibration board, and all position errors on these grid points will be measured and stored in the memory for future use.

In the second step, the robot's end-effector is moved to a target position that is located in the range of the workspace. The target position error could be found by an interpolation technique using the stored 4-neighboring grid position errors around the target position, which were obtained from the first step. Finally, the target position could be compensated with the interpolation results to obtain more accurate positions.

J. Triantafilis and D. Suzana et al reported approaches of using fuzzy interpolation methods to estimate the soil layer and geographical distributions for GIS database [6, 7]. F. Song et al described a fuzzy logic methodology for 4-dimensional (4D) systems with optimal global performance using enhanced cell state space [8]. The most popular interpolation techniques applied in the position compensations of the modeless robotic calibration include the bilinear interpolation and cubic spline interpolation methods; both methods can achieve satisfactory interpolation results for general calibration process [2]. Since the actual position errors are randomly distributed, and it is impossible to pinpoint a position on the error surface at any given moment, the traditional interpolation techniques are unable to provide an accurate estimation of the position errors. The IT2FEI technique utilizes an interval type-2 fuzzy inference system to estimate the position errors, which is consistent with the random distributed nature of position errors. The position errors can be considered as a fuzzy set at any given moment of the time. The fuzzification process takes into account of a range of error rather only a crisp error value. Therefore, the fuzzy error interpolation technique has the fundamentals to improve error estimation results.

The Interval Type-2 Fuzzy inference is the process of formulating the mapping from a given input to an output using interval type-2 fuzzy logic [9-10]. The mapping then provides a basis from which decisions can be made, or patterns discerned. Interval Type-2 Fuzzy inference systems have been successfully applied in fields such as automatic controls, data classification, decision analysis, expert systems, and computer vision [9]. S. Aminifar and A. Marzuki reported an analysis about the uncertainty between the type-1 and Interval Type-2 Fuzzy Logic System (IT2 FLS) [11]. Yu-Chuan Chang presented a new method for handling fuzzy rule interpolation in sparse fuzzy rule-based systems based on interval type-2 fuzzy sets [12]. Kashyap and Sudesh Kumar reported a research of using an i9nterval type-2 fuzzy logic system to perform image fusion [13]. Qing Lu et al. reported to control a nonlinear system with the interval type-2 FLC [14]. Dongrui Wu and Mendel, J.M reported a new method to simplify the design process for the interval type-2 fuzzy system [15]. Schrieber, M.D. and Biglarbegian, M reported to use an interval type-2 FLC to control the FPGA production process [16]. Ching-Chih Tsai et al. reported to use an interval type-2 FLC to control an intelligent bike [17]. Nurmaini, S. and Tutuko, B. described a motion coordination of swarm robots using Interval Type-2 Fuzzy Logic Controller (IT2FLC) to control swarm robots coordination to produce smooth trajectory without collision [18]. Mendel, J. et al. discussed and analyzed different structures of type-1 and type-2 fuzzy controllers [19]. Kumbasar, T examined the robust stability of a PD type Single input Interval Type-2 Fuzzy Logic Controller (SIT2-FLC) structure [20]. All of these updated applications provide a prospective future for interval type-2 fuzzy inference system.

This paper is organized in 5 sections. After this introduction section, the operational principles of the type-1 fuzzy interpolation technique are provided in section 2. Section 3 discusses the interval type-2 fuzzy error interpolation method. A simulation is given in section 4 to illustrate the effectiveness of the fuzzy error interpolation technique. Section 5 presents the conclusion.

II. TYPE-1 FUZZY ERROR INTERPOLATION SYSTEM

A. *The on-line versus off-line fuzzy system*

In order to improve the compensation accuracy, a dynamic on-line fuzzy error interpolation method is introduced. The traditional fuzzy inference system uses pre-defined membership functions and control rules to a construct lookup tables; then a control output is selected from the lookup table. This type of system is called off-line fuzzy inference system because all inputs and outputs have been defined prior to the application process. The off-line fuzzy system cannot meet our requirement for the several reasons. First, the position error of the target point is estimated based on 4 errors of the neighboring grid points, and these 4 neighboring errors are randomly distributed. The off-line fuzzy output membership functions are defined based on the range of errors, which is the neighboring errors' range. However, this range estimation is

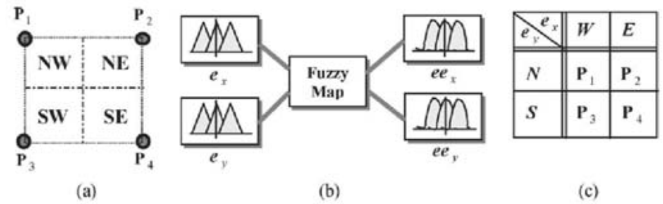


Fig. 2. Definition of the Fuzzy Error Interpolation System.

not as accurate as the real errors obtained on the grid points. Second, since each cell needs one lookup table for the off-line fuzzy system, it would require a huge memory space to save a large number of lookup tables. This results in demanding requirement in both space and time, and as a result, becomes not practical for real time processing. For example, in our study, 20×20 cells are utilized on the calibration board (each cell is 20×20 *mm*); this would require 400 lookup tables. By using an on-line dynamic fuzzy inference system, the target position error can be estimated by combining the output membership functions, which are defined based on the real errors on neighboring grid points and the control rules in real time. The output membership functions are not predetermined, and their definitions are based on the real errors on the grid points, not a range.

Fig. 2 shows the definition of the fuzzy error interpolation inference system.

Each square that is defined by 4 grid points is called a cell; and each cell is divided into 4 equal smaller cells, which are NW, NE, SW and SE, respectively (Fig. 2 (a)). The position error at each grid point is defined as P_1, P_2 , P_3 and P_4 .

For the fuzzy inference system, we apply the fuzzy error interpolation method in two dimensions separately, so the inputs to the fuzzy inference system are e_x and e_y and the outputs are ee_x and ee_y (Fig. 2 (b)). The control rules are shown in Fig. 2 (c), and are discussed following the discussion of membership functions.

B. *Membership functions*

In this work, the distance between two neighboring grid points on the standard calibration board is 20 *mm* in both *x* and *y* directions, which is a standard value for a mid-size calibration workspace. The calibration board includes a total of 20 by 20 cells, which is equivalent to a 400 by 400 *mm* space.

The input membership functions for both *x* and *y* directions and the predefined output membership functions are shown in Fig. 3. The predefined output membership functions are used as default functions, and the final output membership function will be obtained by shifting the default one by the actual error values on the grid points.

The gaussian-bell waveforms are selected as the shape of the membership functions for both inputs (Fig. 3 (a)) in *x* and *y* directions. The ranges of inputs are between –10 mm and 10 *mm* (20 *mm* intervals). H. Zhuang and X. Wu reported a special

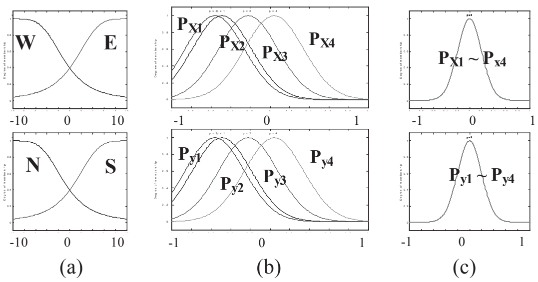


Fig. 3. Input and output membership functions.

histogram method to estimate the optimal membership function distribution [21]. However in our case, a gaussian-bell shape is selected due to the fact that most errors in real world match this distribution. We use W and E to represent the location of inputs in *x* direction, N and S to represent the location of inputs in *y* direction. Unlike the traditional fuzzy inference system, in which all membership functions should be determined in advance to produce the lookup table, the output membership functions will be determined during the application of the fuzzy inference system in real time. This is called dynamic fuzzy system.

Fig. 3 (b) shows an example of the output membership functions, which are related to the simulated random errors at neighboring grid points. Each P_{xi} and P_{yi} correspond to the position error at the *i*th grid point in *x* and *y* directions, respectively. During the design stage, all output membership functions are initialized to a gaussian waveform with a mean of 0 and a range between –0.5 and 0.5 *mm*, which is a typical error range for this workspace in robotic calibration (Fig. 3 (c)) [5]. These output membership functions will be determined based on the errors of the neighboring grid points around the target in the workspace as mentioned above. For example, during the compensation process if the input position in the *x* direction is in the NW area of a cell, the associated output membership function should be modified based on the position error in the NW grid point $\mathbf{P_1}$. This modification is equivalent to shift the P_{x1} Gaussian waveform (Fig. 3 (b)) and allow the center of that waveform to be located at x_0 = the position error value of the $\mathbf{P_1}$ in the *x* direction. Similar modification should be performed for the position error in the y direction. It can be seen from Fig. 3 (b) that the performance loss would be significant if the default output membership function is utilized, which is shown in Fig. 3 (c), for the position compensation process.

C. *Membership functions*

The control rules shown in Fig. 2 (c) can be interpreted as follows after the output membership functions are determined:

- If e_x is W and e_y is N, ee_x is P_{x1} and ee_y is P_{y1} (NW).
- If e_x is W and e_y is S, ee_x is P_{x3} and ee_y is P_{y3} (SW).
- If e_x is E and e_y is N, ee_x is P_{x2} and ee_y is P_{y2} (NE).
- If e_x is E and e_y is S, ee_x is P_{x4} and ee_y is P_{y4} (SE).

Each P_i should be considered as a combination of two error components, P_{xi} and P_{yi}, which are corresponding to errors in both *x* and *y* directions. The error on NW grid point should take more weight if the target position (input) is located inside the NW area on a cell. Similar conclusion can be derived for errors on SW, NE and SE grid points.

III. Interval Type 2 Fuzzy Interpolation System

A. *Overview of the interval type 2 fuzzy interpolation system*

Similar to type 1 fuzzy inference system, the type 2 fuzzy inference system still uses the input and output membership functions, combined with the control rules, to derive the outputs [10, 15]. However, the fuzzy sets used in the type 2 fuzzy logic or the membership grades involved in each membership function are not crisp values, but another fuzzy sets. This means that the membership degrees for all membership functions used in the type 1 fuzzy system are fixed values and can be determined uniquely before the fuzzy inference system works. But the membership degrees for all membership functions used in the type 2 fuzzy system are fuzzy sets. The difference between the standard type-2 fuzzy system and the so-called interval type-2 fuzzy system is that in the former system, the membership degrees are pure fuzzy sets, but the membership degrees are a set of crisp values with a range of 0 ~ 1 or an interval for the latter.

Fig. 4 shows the functional block diagram of an Interval Type-2 FLS [22]. It is similar to Typr-1 FLS, but the major difference is that at least one of the fuzzy sets in the rule base is an IT2 fuzzy set. The outputs of the inference engine are IT2 fuzzy sets, and a type-reducer is needed to convert them into a Typr-1 fuzzy set before defuzzification can be started.

Some fundamental operations in the type-2 fuzzy system are union (3.1), intersection (3.2) and complement (3.3) [23].

The union for interval type-2 fuzzy sets $\tilde{A}$ and $\tilde{\boldsymbol{B}}$ is:

$$\tilde{\tilde{A}} \sqcup \tilde{\tilde{B}} = \{ \int_{x\in X} \mu_{\tilde{\tilde{A}}}(x) \sqcup \mu_{\tilde{\tilde{B}}}(x) / x \} = \left\{ \int_{x\in X} \left[\int_{\alpha\in[\underline{\mu}_{\tilde{A}(x)}\vee\underline{\mu}_{\tilde{B}(x)},\, \overline{\mu}_{\tilde{A}(x)}\vee\overline{\mu}_{\tilde{B}(x)}]} 1/\alpha \right] / x \right\} \quad (3.1)$$

The intersection for interval type-2 fuzzy sets $\tilde{A}$ and $\tilde{\boldsymbol{B}}$ is:

$$\tilde{\tilde{A}} \sqcap \tilde{\tilde{B}} = \{ \int_{x\in X} \mu_{\tilde{\tilde{A}}}(x) \sqcap \mu_{\tilde{\tilde{B}}}(x) / x \} = \left\{ \int_{x\in X} \left[\int_{\alpha\in[\underline{\mu}_{\tilde{A}(x)}\wedge\underline{\mu}_{\tilde{B}(x)},\, \overline{\mu}_{\tilde{A}(x)}\wedge\overline{\mu}_{\tilde{B}(x)}]} 1/\alpha \right] / x \right\} \quad (3.2)$$

The complement for interval type-2 fuzzy sets $\tilde{A}$ and $\tilde{\boldsymbol{B}}$ is:

$$\neg\tilde{\tilde{A}} = \left\{ \int_{x\in X} \mu_{\neg\tilde{\tilde{A}}}(x) / x \right\} = \left\{ \int_{x\in X} \left[\int_{\alpha\in[1-\overline{\mu}_{\tilde{A}}(x),\, 1-\underline{\mu}_{\tilde{A}}(x)]} 1/\alpha \right] / x \right\} \quad (3.3)$$

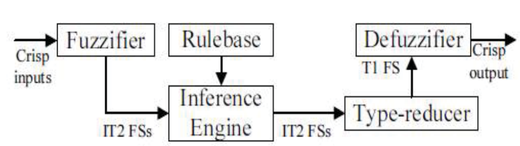


Fig. 4. A functional block diagram of the Interval Type-2 Fuzzy system.

In practice the computations in an IT2 FLS can be significantly simplified. Consider the rule base of an IT2 FLS consisting of N rules assuming the following form [22]:

R^n: if x_1 is X_1^n and … and x_i is X_i^n, then y is Y^n; n=1,2…N

where X_i^n (i = 1 ~ I) are IT2 Fuzzy sets, and $Y^n = [y_1^n, y_2^n]$ is an interval, which can be understood as the centroid [9, 24] of a consequent Interval Type-2 fuzzy set, or the simplest TSK model, for its simplicity. In many applications we use $y_1^n = y_2^n$, i.e., each rule consequent is a crisp number.

Assume the input vector is $x' = (x'_1, x'_2, ..., x'_I)$. Typical computations in an IT2 FLS involve the following steps:

1) Compute the membership of x'_i on each X_i^n
2) Compute the firing interval of the n^{th} rule, $F^n(\mathbf{x}')$
3) Perform type-reduction to combine $F^n(\mathbf{x}')$ and the corresponding rule consequents with the center-of-sets type-reducer [9]:

$$Y_{cos}(\mathbf{x}') = \bigcup_{\substack{f^n \in F^n(\mathbf{x}') \\ y^n \in Y^n}} \frac{\sum_{n=1}^{N} f^n y^n}{\sum_{n=1}^{N} f^n} = [y_l, y_r] \tag{3.4}$$

4) Compute the defuzzified output as:

$$y = \frac{y_l + y_r}{2} \tag{3.5}$$

B. Membership Functions

Similar to type-1 fuzzy interpolation system, the input membership functions for both *x* and *y* directions and the predefined output membership functions for IT2 FLS are shown in Fig. 5. The predefined output membership functions are used as default functions, and the final output membership function will be obtained by shifting the default those by the actual error values on the grid points.

We use W and E to represent the location of inputs in *x* direction, N and S to the location of inputs in *y* direction. For real outputs, 4 membership functions, $p_{x1} \sim p_{x4}$, should be designed for the *x* direction, and another 4 membership functions, $p_{y1} \sim p_{y4}$, are to be built for the *y* direction. These output functions should be located at the center position, which are defined as the default location, as the beginning and changed to the real location based on the actual position errors on each grid point. In Fig. 5, these functions are all displayed but not in the default locations.

As for the control rules, the identical control rules are used for this IT2 FLS, but the fuzzy sets are used as the degrees to replace those crisp values used in the type-1 FLS.

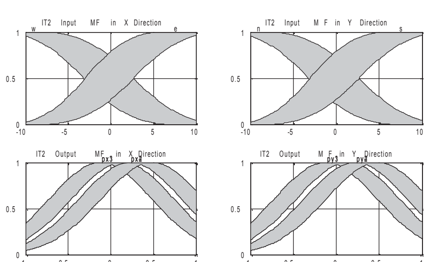


Fig. 5. The input and output membership functions for IT2 FLS.

IV. Simulation Result

Extensive simulation has been performed in order to illustrate the effectiveness of the proposed dynamic online IT2 fuzzy error interpolation technique in comparison to the type-1 FLS. Due to the random nature of the position errors, three different types of error are simulated in this study. These are:

- Normal distributed random error
- Uniform distributed random error
- Sinusoidal waveform error

Figs 6, 7 and 8 show the simulation results of the type-1 and the IT2 fuzzy error interpolation techniques for these three types of error [23].

In these figures, the simulated target (testing) positions on the standard calibration board are spaced from 1 *mm* to 20 *mm* within each cell being with a size of 1 *mm*.

Figs 9 to 11 show comparisons in mean error, maximum error and STD values between type-1 and IT2 fuzzy error interpolation techniques in the histograms.

It can be seen that both mean errors and maximum errors of the IT2 fuzzy error interpolation technique are smaller than

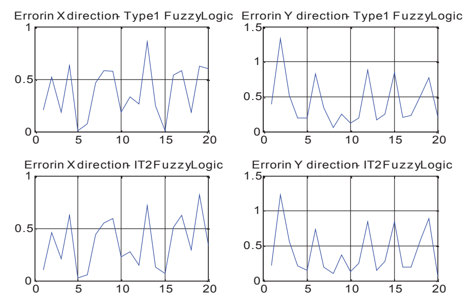


Fig 6. Normal Distributed Random Noise.

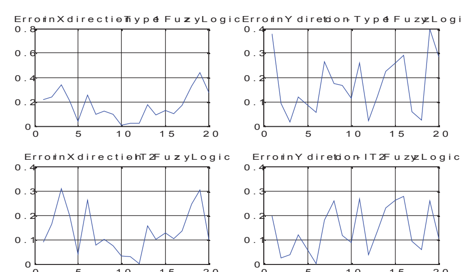


Fig 7. Uniform Distributed Noise.

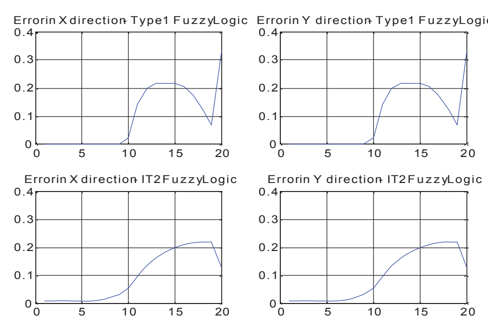


Fig 8. Sinusoidal Waveform Noise.

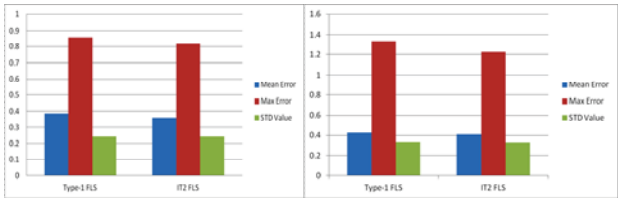


Fig 9. Normal distributed random errors in X and Y directions.

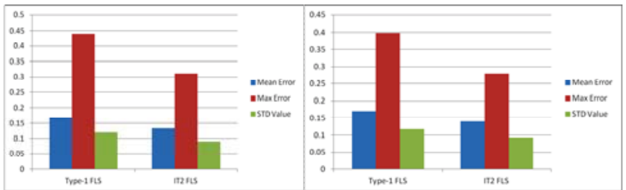


Fig 10. Uniform distributed random errors in X and Y directions.

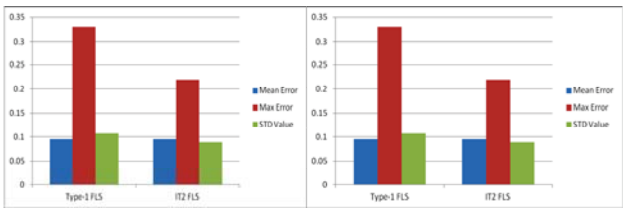


Fig 11. Sinusoid distributed random errors in X and Y directions.

those of type-1 FLS methods. For all three error distribution, the mean errors of the IT2 fuzzy error interpolation method are approximately 10% to 20% smaller compared with those of type-1 FLS method.

The maximum errors of the IT2 fuzzy error interpolation technique are about 10% to 30% smaller than those of the type-1 FLS method. In one case (normal distribution error in x and y direction), the maximum errors of the IT2 fuzzy interpolation method are about 4% smaller than those of the type-1 FLS method.

Figure 12 shows the IT2 fuzzy interpolation error surface.

The simulated results show the effectiveness of the dynamic on-line interval type-2 fuzzy error interpolation technique in reducing the position errors in the modeless robot compensation process.

To implement this interval type-2 fuzzy error interpolation technique as a real time application, an interface between the MATLAB® and high level programming languages C/C++ has been developed [25]. Although the most popular real time control programming language is C/C++, the fuzzy error interpolation method is developed and implemented in MATLAB [23]. Using this interface, the measured position errors on the grid points can be passed from C/C++ to MATLAB functions that implements the interval type-2 fuzzy error interpolation functions; and the fuzzy error interpolation results can be sent back to C/C++ for the real time controller to operate on the next target position.

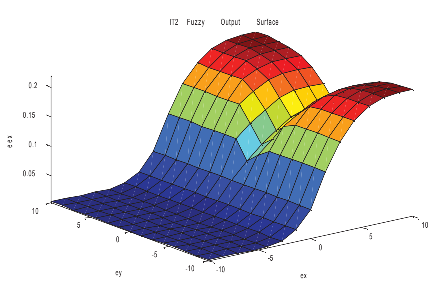


Fig 12. The IT2 fuzzy interpolation error surface.

V. Conclusion and Summary

A dynamic on-line interval type-2 fuzzy error interpolation technique is presented in this paper. The compensated position errors in a modeless robot calibration can be greatly reduced by the proposed technique. Simulation results demonstrate the effectiveness of the proposed fuzzy error interpolation technique. Three typical error models are utilized for comparison and simulation; these include sinusoidal waveform, normally distributed and uniformly distributed errors. This fuzzy error interpolation technique is ideal for the modeless robot position compensation, especially the high accuracy robot calibration process.

Acknowledgment

Special thanks to Dr. Oscar Castillo and his group to provide me the Interval Type-2 Fuzzy Logic Toolbox to enable us to perform our simulations for this paper [26-27]. We could not develop and share this paper with all of you without his sincerely help and support.

References

[1] B. W. Mooring, Z. S. Roth and M. R. Driels, Fundamentals of Manipulator Calibration, John Wiley & Sons, Inc. 1991.

[2] M. A. Penna, "Camera Calibration: A Quick and Easy Way to Determine the Scale Factor", IEEE Trans. Pattern Anal. Machine Intell. 13, 1991, pp. 1240-1245.

[3] J. Weng et al, "Camera Calibration with Distortion Models and Accuracy Evaluation", IEEE Trans, Pattern Anal. Machine Intell. 14, 1992, pp. 965-980.

[4] F. Du and M. Brady, "Self Calibration of the Intrinsic Parameters of Cameras for Active Vision Systems", In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, New York, NY June, 1993, pp. 477-482..

[5] Hanqi Zhuang and Z. S. Roth, Camera-Aided Robot Calibration, CRC Press, Inc. 1996.

[6] J. Triantafilis, W. T. Ward, I.O.A Odeh and A.B. McBratney, "Creation and Interpolation of Continuous Soil Layer Classes in the Lower Namoi Valley", Soil Science Society of America Journal, Mar/Apr 2001, Vol. 65, Issue 2, pp. 403-414.

[7] D. Suzana, J.D. Marceau and M. Claude, "Space, time, and dynamics modeling in historical GIS databases: A Fuzzy Logic Approch", Environment & Planning B: Planning & Design, Jul 2001, Vol. 28 Issue 4, pp. 545-563.

[8] F. Song, S. M. Smith, and C. G. Rizk, "A Fuzzy Logic Controller Design Methodology for 4D System with Optimal Global Performance Using Enhanced Cell State Space Based Best Estimate Directed Search Method", IEEE Int. Conf. On Systems, Man and Cybernetics, 1999.

[9] Jerry M. Mendel. Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions, Prentice Hall PTR, 2001.

[10] Oscar Castillo and Patricia Melin, Type-2 Fuzzy Logic: Theory and Applications, Studies in Fuzziness and Software Computing, Vol. 223, Springer, 2008.

[11] S. Aminifar and A. Marzuki, "Uncertainty in Interval Type-2 Fuzzy Systems, Mathematical Problems in Engineering", Volume 2013 (2013), Article ID 452780, 16 pages.

[12] Yu-Chuan Chang, "A new fuzzy interpolative reasoning method based on interval type-2 fuzzy sets", IEEE International Conference on Systems, Man and Cybernetics, 2008. SMC 2008, Singapore, Oct. 12-15, 2008, pp.82-87.

[13] Kashyap and Sudesh Kumar, "IR and color image fusion using interval type 2 fuzzy logic system", 2015 International Conference on Cognitive Computing and Information Processing (CCIP), Noida, India, 3-4 March 2015, pp.1-4.

[14] Qing Lu et al, "Interval Type-2 Fuzzy Model Predictive Control of Nonlinear Networked Control Systems", IEEE Transactions on Fuzzy Systems, Vol. PP, Issue 99, March 2015, pp. 1-8.

[15] Dongrui Wu and Mendel, J.M, "Designing practical interval type-2 fuzzy logic systems made simple", 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Beijing, 6-11 July 2014, pp. 800-807.

[16] Schrieber, M.D. and Biglarbegian, "Hardware implementation of a novel inference engine for interval type-2 fuzzy control on FPGA", 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 6-11 July 2014, Beijing, 6-11 July 2014, pp. 640-646.

[17] Ching-Chih Tsai et al., "Interval type-2 fuzzy gear-changing control for intelligent bikes", 2014 International Conference on Machine Learning and Cybernetics (ICMLC), 13-16 July 2014, Lanzhou, pp. 741-747.

[18] Nurmaini, S. and Tutuko, B., "Motion coordination for swarm robots", 2014 International Conference on ICT for Smart Society (ICISS), 24-25 Sept. 2014, Bandung, pp. 312-315.

[19] Mendel, J. et al., Book Chapter, Analytical Structure of Various Interval Type-2 Fuzzy PI and PD Controllers, Wiley-IEEE Press, 2014.

[20] Kumbasar, T., "Robust stability analysis of PD type single input interval type-2 fuzzy control systems", 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 6-11 July 2014, Beijing, pp. 634-639.

[21] H. Zhuang and Xiaomin Wu, "Membership Function Modification of Fuzzy Logic Controllers with Histogram Equalization", IEEE Trans. On System, Man, and Cybernetics – Part B: Cybernetics, Vol. 31, No. 1, Feb. 2001.

[22] Dongrui Wu, "A Tutorial on Interval Type-2 Fuzzy Sets and Systems", University of Southern California, USA2012.

[23] ITT/UABC, Interval Type-2 Fuzzy Logic Toolbox For Use with MATLAB®, Tijuana Institute of Technology and Baja California Autonomous, University, Tijuana Campus, Mexico, 2008.

[24] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," Information Sciences, vol. 132, pp. 195–220, 2001.

[25] Ying Bai, Applications Interface Programming Using Multiple Languages, Prentice Hall, March 2003.

[26] J.R. Castro, O. Castillo, P. Melin, L.G. Martínez, S. Escobar, I. Camacho. "Building Fuzzy Inference Systems with the Interval Type-2 Fuzzy Logic Toolbox", IFSA, 2007.

[27] Juan R. Castro, Oscar Castillo, Patricia Melin. "An Interval Type-2 Fuzzy Logic Toolbox for Control Applications", FUZZ-IEEE, 2007.

A Dynamic Hierarchical Task Transfer in Multiple Robot Explorations

Mehran Asadi
Information Technology
The Lincoln University
Lincoln University, PA 19352, U.S.A
masadi@lincoln.edu

Manfred Huber
Computer Science and Engineering
University of Texas at Arlington
Arlington, TX 76019, U.S.A
huber@cse.uta.edu

***Abstract*—To operate effectively in complex environments, learning agents have to selectively ignore irrelevant details by forming useful abstractions. These abstractions can be constructed using subtasks that are defined prior to the learning process. In this paper we extend our previous discoveries to a new multi-robot environment and we combine two recent methods in hierarchical reinforcement learning in order to introduce a novel mechanism that discovers the sub-policies in Markov Decision Process in a multi-agent system.**

I. Introduction

The work presented here focuses on the construction and transfer of control knowledge in the form of behavioral skill hierarchies and associated representational hierarchies in the context of a reinforcement learning agent. In particular, it facilitates the acquisition of increasingly complex behavioral skills and the construction of appropriate, increasingly abstract and compact state representations which accelerate learning performance while ensuring bounded optimality. Moreover, it forms a state hierarchy that encodes the functional properties of the skill hierarchy, providing a compact basis for learning that ensures bounded optimality.

II. Hierarchical Reinforcement Learning

To permit the construction of a hierarchical learning system, we model our learning problem as a Semi-Markov Decision Problem (SMDP) and use the options framework [1], [2] to define subgoals. An option is a temporally extended action which, when selected by the agent, executes until a termination condition is satisfied. While an option is executing, actions are chosen according to the option's own policy. An option is like a traditional macro except that instead of generating a fixed sequence of actions, it follows a closed-loop policy so that it can react to the environment. By augmenting the agent's set of primitive actions with a set of options, the agent's performance can be enhanced. More specifically, an option is a triple $o_i = (I_i, \pi_i, \beta_i)$, where I_i is the option's input set, i.e., the set of states in which the option can be initiated; π_i is the option's policy defined over all states in which the option can execute; and β_i is the termination condition, i.e., the option terminates with probability $\beta_i(s)$ for each state s. Each option that we use in this paper bases its policy on its own internal value function, which can be modified over time in response to the environment. The value of a state s under an SMDP policy π^o is defined as [3], [1], [4], [5]:

$$V^\pi(s) = E\left[R(s, o_i) + \sum_{s'} F(s'|s, o_i)V^\pi(s')\right]$$

where

$$F(s'|s, o_i) = \sum_{k=1}^{\infty} P(s_t = s'|s_t = s, o_i)\gamma^k$$

, where $\gamma \in [0, 1]$ is a discount-rate parameter.

III. Previous Work

In our previous work [6], [7] we constructed an appropriate BPMDP for a specific action set

$O_t = \{o_i\}$, and an initial model was constructed by concatenating all concepts associated with the options in O_t. Additional conditions are then derived to achieve the stability of partition and, once reward information is available, the partitions were further refined according to a defined criteria. This construction facilitates efficient adaptation to changing action repertoires.

To further utilize the power of abstract actions, a hierarchy of BPSMDP (Bounded Parameter SDMP) models was constructed where the decision-level model utilized the set of options considered necessary while the evaluation-level used all actions not considered redundant. In the our system, a simple heuristic was used where the decision-level set consisted only of the learned subgoal options while the evaluation-level set included all actions.

Let $P = \{B_1, \ldots, B_n\}$ be a partition for state space S derived by the action-dependent partitioning method, using subgoals $\{s_1, \ldots, s_k\}$ and options to these subgoals $\{o_1, \ldots, o_k\}$. If the goal state G belongs to the set of subgoals $\{s_1, \ldots, s_k\}$, then G is achievable by options $\{o_1, \ldots, o_k\}$ and the task is learnable. However, if $G \notin \{s_1, \ldots, s_k\}$ then the task may not be solvable using only the options that terminate at subgoals. The proposed approach solves this problem by maintaining a separate value function for the original state space while learning a new task on the partition space derived from only the subgoal options. During learning, the agent has access to the original actions as well as all options, but makes decisions only based on the abstract partition space information.
While the agent tries to solve the task on the abstract partition space, it computes the difference in Q-values between the best actions in the current state in the abstract state space and in the original state space. If the difference is larger than a constant value , then there is a significant difference between different states underlying the particular block that was not captured by the subgoal options.

IV. Autonomous Hierarchy Construction

In the multi-phase partitioning and hierarchical learning method discussed in the previous section, it has so far been assumed that either the correct set of actions for constructing an abstract state space is available or that, as a simple heuristic, all subgoal options are selected as the relevant action set. While the latter can lead to good results when used in conjunction with the learning method it might lead to an ever increasing action set if a large sequence of tasks is to be learned. In particular, this heuristic has the limitation that it can never remove an option from the action set used for multi-phase partitioning, even if it is not used for any of the tasks. To address this limitation, this section presents a method aimed at automatically constructing the abstract representation based on the information contained in the previously learned task policies.
In order to estimate the structure of the state space for learning future tasks, we construct the decision layer here based on an estimate of the expected time to learn a new task according to previously learned tasks. Let $\Pi = \{\pi_1, \ldots, \pi_n\}$ be the set of previously learned polices and $P_i = \{B_{i,1}, \ldots, B_{i,n}\}$ be the corresponding partitions. Also let the triple $T_i = (\pi_i, P_i, Q_i)$ be a task on partition $P_i = \{B_{i,1}, \ldots, B_{i,n}\}$ with policy π_i and the Q-function Q_i. The expected number of experiences required to learn a task, with high probability, on partition P with action set O using a DP-based version of Q-learning is [8]:

$$T_{conv}(P, O) = c|P|^2|O|$$

where c is a constant and it is assumed that the task is learnable on P with action set O.
The expected time required to learn task T_i on state representation P (including the refinement process) can be obtained by calculating the number of experiences that are needed for learning T_i on partition P plus the amount of time that is needed to refine a block of partition P, that is:

$$E[t_{T_i}|P] = t_{conv}(P, O) + \sum_{B_j \in P} P_{refine}(B_j|T_i)\, t_{conv}(\{B_{i,k}|B_{i,k} \cap B_j \neq \emptyset\}, O)$$

We compute the likelihood that a block B_j has to be refined during the exploration and learning of task T_i with the following equation:

$$P_{refine}(B_j|T_i) = \sum_{B_{i,k}: B_{i,k} \cap B_j \neq \emptyset} P_{refine}(B_j|B_{i,k}, T_i) P(B_{i,k}|T_i)$$

where

$$P_{refine}(B_j|B_{i,k},T_i) = \begin{cases} 1 & \text{if } A \\ 0 & \text{otherwise} \end{cases}$$

where $A = max_a(Q_i(B_{i,k},a) > max_{a \in O_{B_{i,k}}}(Q_i(B_{i,k},a))) + L$ and $L = 2(1 + (\frac{\gamma}{1-\gamma}))\max\{\epsilon,\delta\}$ and $P_{refine}(B_j|B_k^i,T_i)$ is the probability that block B_j has to be refined during the exploration and learning of T_i due to encountering block $B_{i,k}$ which is at least partially contained in B_j and for which an action a which is not contained in the currently considered action set $O_{B_{i,k}}$, with significantly higher value should then be included using the hierarchical learning scheme.

We compute the expected time required to learn a task randomly chosen from the distribution of previously learned tasks according to an importance distribution $U(T_i)$ which indicates the weight that should be put on each tasks by:

$$E[t_{learn}|P] = \sum_i \frac{U(T_i)}{\sum_i U(T_i)} E[t_{T_i}|P]$$

Algorithm 1 illustrates the process of autonomous hierarchy construction, in particular this is a greedy algorithm that finds action-dependent partitions that have the smallest expected learning time given previously learned tasks. The reason for the greedy approach is to reduce the complexity sufficiently to make it tractable. This approach is very similar to McCullum's U-tree algorithm [9], [10] except that splits are driven not by reward but by the expected learning time metric derived before. This procedure can be done either by splitting the blocks separately or by limiting the inclusion of actions across the state space. While the latter saves us more computational time, the former will give us more nuanced splits.

Algorithm 1 Autonomous Hierarchy Construction

Require: $O_0 = \emptyset$, $P_0 = \{s\}$
  $n = 0$
  repeat
    for all B_j in P_n and $o_i \in O - O_{n,B_j}$ **do**
      $P_{n+1,(i,j)} = P_n$ where B_j is refined with o_i
    end for
    $(k,l) = argmin_{(b,c)} E[t_{learn}|P_{n+1,(b,c)}]$
    $P_{n+1} = P_{n+1,(k,l)}$
    $B = B_l$
    for all $B_i \in P_n$ **do**
      for all $B_j \in P_{n+1}, B_j \subseteq B_i$ **do**
        if $B_i = B$ **then**
          $O_{n+1,B_j} = O_{n,B} \cup \{o_k\}$
        else
          $O_{n+1,B_j} = O_{n,B_i}$
        end if
      end for
    end for
    $n = n + 1$
  until $E[t_{learn}|P_n] \geq E[t_{learn}|P_{n-1}]$
  return P_{n-1}
  END

V. Empirical Results

The experiment shows the result of the presented approach in a game domain that is more complex and more similar to real environments. While all these experiments use the heuristic of using all subgoals action to construct the abstract decision layer, the experiment in the same game domain investigates the autonomous hierarchy construction approach in order to illustrate the construction of an approximate partition using the information of the previously learned polices.

The actions are GoUp, GoDown, TurnLeft, TurnRight, PickUp and DropOff. The cost for each single step action is -1 and each action for navigation succeeds with probability 1. The reward in the goal state where the agent can pickup and drop off the object is 100. The state is here characterized by the agent's pose as well as by a set of local object percept, resulting in an effective state space with $20,000$ states. The agent is first presented with a reward function to learn to move to a specific location. Once this task is learned, subgoals are extracted by generating random sample trajectories.

In order to show the construction of the decision layer, a sequence of five different tasks is learned in the game environment. The first task is to navigate the environment, i.e., the agent learns how to move from one location to another location. The second task is to navigate the environment and pick up an object. The goal of third task is to navigate a different region of the environment, and in the fourth task the agent learns how to navigate, pick up an object and drop it off in another location. The fifth task is the combination of the first four tasks

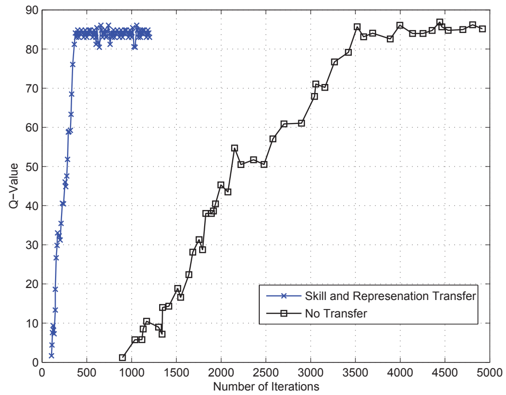


Fig. 1. Learning curves for the first navigation task. The agent learns to navigate the environment and the information acquired by learning this task will be used for constructing a partition for the next task ,i.e, the navigation and pickup tasks

by using the information acquired while learning the first four tasks, i.e., the agent learns to navigate the environment and to pick up an object and drop it off in another location.
Figures 1, 2, 3 and 4 show the learning curves for the first four tasks.

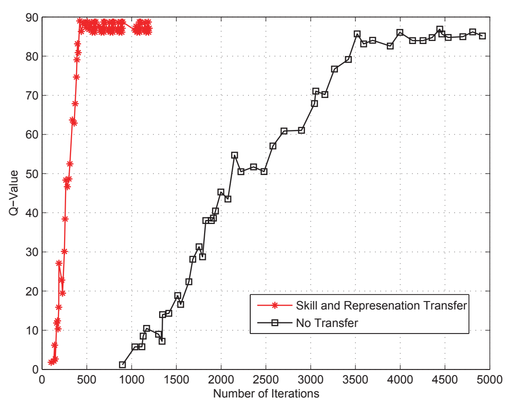


Fig. 2. Learning Curves for the second task, i.e., the navigation and pickup tasks. The information acquired by learning this task and the first task will be used for constructing a partition for the third task

At each step, a previously learned policy is added to the action set in order to construct a partition

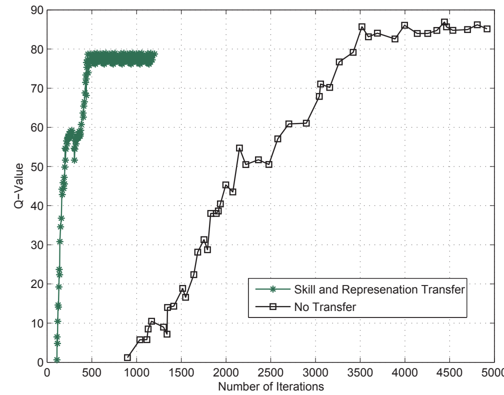


Fig. 3. Learning Curves for the third task, i.e., the second navigation task. The information acquired by learning this task and the previous two tasks will be used for constructing a partition for the fifth task

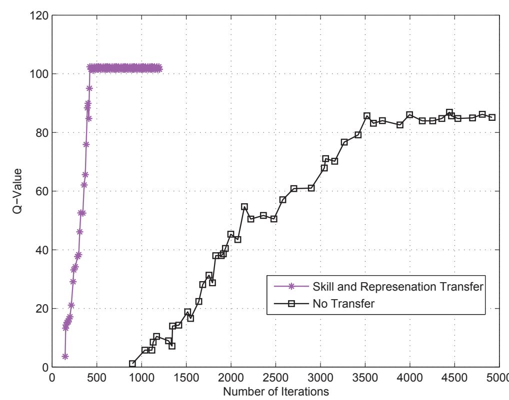


Fig. 4. Learning Curves for the fourth task, i.e., the navigation and dropoff tasks. A new partition will be constructed by using a history of previously learned tasks for future subsequent tasks

that is more relevant to the learning of a new task using Algorithm 1. The number of blocks for task 1 through 5 is illustrated in Figure 5. The number of blocks of this partition is illustrated in Figure 6. This experiment shows how a new partition can be constructed by using a history of previously learned task while it ensures that the new policy is within a fixed bound from the optimal policy. Figure 7 illustrates the learning curves on the compact state space, constructed by

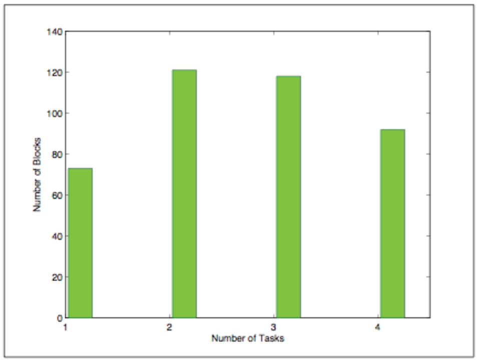


Fig. 5. Number of blocks constructed for learning task 1 through task 5

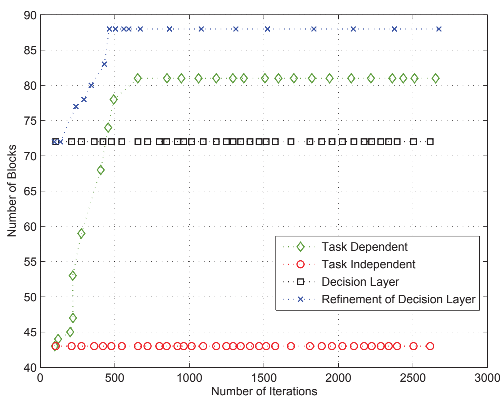


Fig. 6. Number of blocks for decision layer after refinement of task dependent partition. As a result of further refinement of the original blocks of partition the number of blocks increases, however this number becomes stables after finite and relatively small number of iterations.

using previously learned polices.

VI. Conclusion and Future Work

The results presented in this paper show a significant reduction in the number of states in the abstract state space, resulting in faster convergence of the value function. Furthermore, these experiments show a procedure to estimate the structure of the state space for learning future tasks and to construct the decision layer based on the expected time to learn a new task according to previously learned tasks.

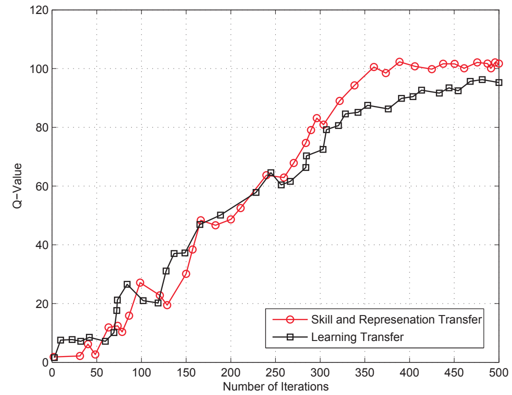


Fig. 7. Learning on a partition space obtained by Autonomous Hierarchy construction method by using the first four tasks. This experiment shows how a new partition can be constructed by using a history of previously learned tasks while it ensures that the new policy is within a fixed bound from the optimal policy

One of the future goals is to find even more efficient machine learning methods for control tasks. Algorithms can be developed for statistical generalization and reasoning about the algorithms that learn to incrementally scale up to analyze even more complex tasks. Discovering hierarchy in task structure and world structure is an important means in achieving this end. Algorithms need to be developed that learn to reason about their environment in a combinatorial way and learn to develop more cognitive internal representations that mimic relational structures. Integration of more powerful representations such as factorial HMMs and POMDPs are a potential follow-up to this work. Smarter hierarchical algorithms must be found to deal with larger tasks, and research must be directed at more intelligent representational design not only for incorporating hierarchy but also for sharing substructures.

References

[1] R. Sutton, D. Precup, and S. Singh, "Between MDPs and Semi-MDPs: Learning, Planning, and Representing Knowledge at Multiple Temporal Scales," *Artificial Intelligence*, vol. 112, pp. 181–211, 1999.

[2] R. Parr, "Hierarchical Control and Learning for Markov Decision Processes," Ph.D. dissertation, University of California, Berkeley, CA, 1998.

[3] C. Boutilier, T. Dean, and S. Hanks, "Decision-Theoretic Planning: Structural Assumptions and Computational Leverage," *Journal of Artificial Intelligence Research*, vol. 11, pp. 1–94, 1999.

[4] K. Kim and T. Dean, "Solving Factored MDPs using Non-Homogeneous Partitions," *Artificial Intelligence*, vol. 147, pp. 225–251, 2003.

[5] T. G. Dietterich, "An Overview of MAXQ Hierarchical Reinforcement Learning," *Lecture Notes in Computer Science*, vol. 1864, 2000.

[6] M. Asadi and M. Huber, "State Space Reduction For Hierarchical Reinforcement Learning," in *Proceedings of the 17th International FLAIRS Conference*. AAAI, 2004, pp. 509–514.

[7] ——, "Effective control knowledge transfer through learning skill and representation hierarchies," in *IJCAI, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India*, 2007, pp. 2054–2059.

[8] A. Barto and S. Mahadevan, "Recent Advances in Hierarchical Reinforcement Learning," *Discrete Event Dynamic Systems*, vol. 13, pp. 341–379, 2003.

[9] A. McGovern and A. Barto, "Automatic Discovery of Subgoals in Reinforcement Learning using Diverse Density," in *Proceedings of the 18th International Conference on Machine Learning*, 2001, pp. 361–368.

[10] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.

[11] H. Janzadeh and M. Huber, "Learning policies in partially observable mdps with abstract actions using value iteration," in *Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference.*, 2013.

[12] H. Rahmanian and M. Huber, "Data modeling using channel-remapped generalized features," in *IEEE International Conference on Systems, Man, and Cybernetics*, 2013, pp. 864–869.

[13] M. Aurangzeb, F. L. Lewis, and M. Huber, "Efficient, swarm-based path finding in unknown graphs using reinforcement learning," *Control and Intelligent Systems*, vol. 42, no. 3, 2014.

[14] D. M. Clement and M. Huber, "Using multi-agent options to reduce learning time in reinforcement learning," in *Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference*, 2015, pp. 26–31.

[15] C. Boutilier, R. Dearden, and M. Goldszmidt, "Exploiting Structure in Policy Construction," in *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1995, pp. 1104–1111.

[16] A. McCallum, "Overcoming Incomplete Perception with Utile Distinction Memory," in *Proceedings of the Tenth International Machine Learning Conference*, 1993.

[17] T. Dean, R. Givan, and M. Greig, "Equivalence Notions and Model Minimization in Markov Decision Processes," in *Special issue on planning with uncertainty and incomplete information*, 2003, pp. 163–223.

[18] T. Dean, L. Kaelbling, J. Kirman, and A. Nicholson, "Planning Under Time Constraints in Stochastic Domains," *Artificial Intelligence*, vol. 76, no. 1-2, pp. 35–74, 1995.

[19] B. Digney, "Emergent hierarchical control structures: Learning reactive / hierarchical relationships in reinforcement environments," in *Proceedings of the Fourth Conference on the Simulation of Adaptive Behavior*, 1996.

[20] C. Drummond, "Using a Case Base of Surfaces to Speed-Up Reinforcement Learning," in *Proceedings of the Second International Conference on International Conference on Case-Based Reasoning*, 1997, pp. 435–444.

[21] T. Dean, R. Givan, and S. Leach, "Model Reduction Techniques for Computing Approximately Optimal Solutions for Markov Decision Processes," in *Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence (UAI-97)*. San Francisco, CA: Morgan Kaufmann Publishers, 1997, pp. 124–131.

[22] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[23] R. H. Crites and A. G. Barto, "Elevator group control using multiple reinforcement learning agents," *Machine Learning*, vol. 33, pp. 235–262, 1998.

[24] R. Parr and S. Russell, "Reinforcement learning with hierarchies of machines. in," in *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 1998.

[25] T. G. Dietterich, "Hierarchical reinforcement learning with the maxq value function decomposition," *Artificial Intelligence Research*, vol. 13, pp. 227–303, 2000.

[26] M. L. Puterman, *Markov Decision Problems*. New York: Wiley, 1994.

[27] G. J. Tesauro, "Practical issues in temporal difference learning," *Machine Learning*, vol. 8, pp. 257–277, 1992.

[28] J. N. Tsitsiklis and B. V. Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Transactions on Automatic Control*, vol. 42, pp. 674–690, 1997.

[29] S. Mahadevan, N. Marchalleck, T. Das, and A. Gosavi, "Self-improving factory simulation using continuous-time average-reward reinforcement learning," in *Proceedings of the 14th International Conference on Machine Learning*, Nashville, TN, 1997.

[30] S. Singh and D. Bertsekas, "Reinforcement learning for dynamic channel allocation in cellular telephone systems," in *Proceedings of the 1996 Conference on Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 1997.

[31] C. J. C. H. Watkins, "Learning from delayed rewards," Ph.D. dissertation, Cambridge University, 1989.

[32] R. S. Sutton, "Generalization in reinforcement learning: Successful examples using sparse coarse coding," in *Advances in Neural Information Processing Systems*. Cambridge, MA: The MIT Press, 1996, pp. 1038–1044.

Inferring Robot Actions from Verbal Commands Using Shallow Semantic Parsing

A. Sutherland, S. Bensch, and T. Hellström
Department of Computing Science, Umeå University, Umeå, Sweden

Abstract— *Efficient and effective speech understanding systems are highly interesting for development of robots working together with humans. In this paper we focus on interpretation of commands given to a robot by a human. The robot is assumed to be equipped with a number of pre-defined action primitives, and an uttered command is mapped to one of these actions and to suitable values for its associated parameters. The approach taken is to use data from shallow semantic parsing to infer both the action and the parameters. We use labeled training data comprising sentences paired with expected robot action. Our approach builds on the hypothesis that the expected action can be inferred from semantic frames and semantic roles, information that we retrieve from the Semafor system. The generated frame names and semantic roles are used to learn mappings to expected robot actions and their associated parameters. The results show an accuracy of 88% for inference of action alone, and 68% for combined inference of an action and its associated parameters. Given the large linguistic variety of the input sentences, and the limited size of data used in for learning, these results are surprisingly positive and promising.*

Keywords: HRI, NLP, Robotics, Semantics, Arbitration

1. Introduction

Speech is one of the most efficient means of communication for humans, and has also been extensively addressed in human-robot interaction research [1], [2]. While robust speech recognition is a major unsolved problem in natural language processing (NLP), challenges also remain in other areas of NLP, such as syntactic and semantic analysis. Even if these problems would be solved, a general method to generate correct robot responses to speech requires a level of intelligence that is out of reach for current research in both cognitive science and artificial intelligence. The challenges in finding a general solution has of course not prevented researchers from proposing solutions to specific domains and sentence structures. Most implementations of NLP in robotics is concerned with imperative commands and this is also the target for the work presented in this paper. We propose a method to create mappings from sentences to expected robot actions. Humans, not least young children, are able to the perform such learning in a non supervised manner, i.e. without being explicitly told what to do when a certain sentence is uttered. For an excellent analysis of this process see [3]. While still awaiting human level intelligence in robots, we make the task somewhat simpler by providing the expected robot action for each sentence. Thus, we provide labeled training data comprising sentences paired with the expected robot action. A robot action comprises the name of the pre-defined action, and values for one or several parameters specifying the action. Our approach build on a hypothesis that the expected action can be inferred from shallow semantic data. In the learning phase, the labeled sentences are semantically parsed using the commonly available Semafor system [4]. The generated frame names and semantic roles are used to create mappings to expected robot actions including associated parameters. The evaluation shows very good results for cross-validated data.

This paper is organized as follows: Section 2 gives a brief overview of earlier related work. The theory behind semantic roles is briefly described in Section 3, followed by a description of our approach for generation of mappings from sentences to frames and semantic roles. The mechanism for inference of actions and parameters is described in Section 4. Results are presented in Section 5, followed by a discussion of results and limitations in Section 6, and plans for future work in Section 7.

2. Related earlier work

Substantial research has been focused on speech-based systems for giving commands to robots. In [5], a system for programming a robot for complex tasks through verbal commands is presented. The system filters out unknown words, maps predefined keywords to actions and parameters, and generates graphs representing the required task. The authors in [6] propose a speech-based robot system for controlling a robotic forklift. The proposed system requires commands to be given according to a given syntax. In [7], teaching soccer skills via spoken language is addressed. The vocabulary is predefined and focus is rather on constructing advanced control expressions than on language understanding. The authors in [8] use labeled sentences to learn a combinatory categorical grammar (CCG) for the specific task of interpretation of route instructions. In [9], specific techniques for incremental language processing coupled to inference of expected robot actions are described. The approach is to construct a grammar that facilitates incremental analysis such that robots can act pro-actively already during a verbal command is given.

The present work is similar to the once mentioned above in the aim to interpret natural language sentences by mapping sentences to expected robot actions. However, it differs by the method of using semantic frames and roles from an existing parser as inputs in the mapping.

Other attempts to human-robot interaction through natural language build on more traditional grammatical analysis combined with reasoning mechanisms to generate suitable robot actions. Low recognition rates and ungrammatical, incomplete or fragmentary utterances have been addressed in several ways. The authors in [10] constrain the task and use incremental language analysis based on CCG, regular expression-based filter and a trigram statistical model to process fragmentary or otherwise ungrammatical utterances.

3. Shallow semantic parsing

Shallow semantic parsing, also called semantic-role-labeling, is the task of finding semantic roles for a given sentence. Semantic roles describe general relations between predicates and its arguments in a sentence. For example, in a sentence like "Mary gave the ball to Peter", "gave" is the predicate, "Mary" represents the semantic role *donor*, "the ball" represents the semantic role *theme*, and "Peter" represents the semantic role *recipient*.

FrameNet [11] is a system with large amounts of such analyzes for English sentences. Whereas other attempts, like *PropBank* [12], assign roles to individual verbs, FrameNet assign roles to *frames*. A semantic frame includes a list of associated words and phrases that can potentially evoke the frame. Each frame also defines several semantic roles corresponding to aspects of the scenario represented by the frame. The Semafor system (Semantic Analyzer of Frame Representations) is built on FrameNet and provides both an on-line service (http://demo.ark.cs.cmu.edu/parse) and downloadable code for off-line use of the system. Semafor is reported [13] to have achieved the best published results up to 2012 on the *SemEval 2007 frame-semantic structure extraction task* [14]. In the present work we use the Semafor on-line system for extraction of frames and semantic roles from all sentences used in the experiments.

4. Description of method

We propose a method by which the expected actions for a verbally uttered commands to a robot can be learned, such that the robot automatically can determine what to do when hearing a new sentence. The robot learns how to infer action and parameters from a set of labeled example sentences. Each sentence is parsed by a shallow semantic parser that produces frames and associated semantic roles. If multiple frames occur, the frame related to the predicate is selected, and denoted as the *primary* frame. Conditional probabilities for how these entities relate to expected actions and associated parameters are estimated and used to construct the necessary inference mechanisms.

The example sentences used in this paper were manually generated. Each sentence was labeled with one of n_A robot actions $a_1, ..., a_{n_A}$ and m_a associated parameters $p_1, ..., p_{m_a}$ (see Table 1). A total of 94 sentences representing plausible commands that a human might utter to a robot were generated. Some examples are given in Table 4.

Table 1: Pre-programmed robot actions a_i with associated parameters p_1, p_2.

i	a_i	p_1	p_2	Expected function
1	BRING	object	recipient	Fetches object
2	TELL	message	recipient	Relays a message
3	COLLECT	object	source	Gathers objects
4	MOVE	location		Moves self to location
5	PUT	object	location	Places an object

For the purpose of this paper, the actions did not have to be physically implemented but would in a complete system be pre-programmed in the robot.

The proposed method comprises a learning part and an inference part, as described in the following two sub-sections.

4.1 Learning

In the learning phase, each sentence in a training data set comprising N sentences was presented to the Semafor system, which output frames and associated semantic roles. If several frames were generated, the primary frame is selected. For our entire data set, $n_F = 21$ distinct primary frames $f_1, ..., f_{n_F}$, were generated, and are listed in Table 2 together with some of their most common semantic roles.

The proposed method builds on the hypothesis that the expected action for a command can be inferred from the primary frame of the command. To initially test this hypothesis, statistics for combinations of primary frames and labeled actions for all sentences were generated, and is presented in Table 3. The number of occurrences for each frame/action combination is shown, followed by the relative frequency after the / symbol. Most rows contain only one non-zero entry, thus supporting the hypothesis that the expected action can be inferred from the frame. However, some frames occur for more than one action, and many actions occur for several frames.

In order to infer expected action from the primary frame of a sentence, the conditional probability

$$P(Action = a_i | Frame = f_j), \tag{1}$$

i.e. for the expected action to be a_i, given a primary frame f_j, are estimated. With simplified notation and by using the

Table 2: Frame names generated by the Semafor system for the sentences used in the experiments.

i	Frame f_i	Common semantic roles
1	BRINGING	Theme Goal Source Path
2	GETTING	Event Experiencer Focal participant
3	GIVING	Donor Recipient Theme
4	NEEDING	Cognizer Dependant Requirement
5	DESIRING	Event Experiencer
6	TELLING	Addressee Message Speaker
7	STATEMENT	Message Speaker Medium
8	POLITICAL LOCALES	Locale
9	BEING NAMED	Entity Name
10	TEXT	Text Author
11	COME TOGETHER	Individuals
12	AMASSING	Mass Theme Recipient
13	GATHERING UP	Agent Individuals
14	PLACING	Agent Goal Theme
15	MOTION	Path Goal Theme
16	GRANT PERMISSION	Grantee Grantor Action
17	DEPARTING	Source Theme
18	STIMULUS FOCUS	Stimulus
19	HAVE AS REQ.	Dependant Required entity Requirement
20	LOCALE BY USE	Locale Use
21	COMPLIANCE	Act Norm Protagonist

definition of conditional probability, (1) can be written as

$$P(a_i|f_j) = P(a_i, f_j)/P(f_j), \tag{2}$$

which can be estimated from data by

$$\widehat{P}(a_i, f_j) = \#(a_i, f_j)/N \tag{3}$$

and

$$\widehat{P}(f_j) = \#(f_j)/N, \tag{4}$$

where $\#(a_i, f_j)$ denotes the total number of sentences in the training data that were labeled with action a_i and for which Semafor determines f_j as primary frame[1]. Hence, $P(a_i|f_j)$ can be estimated by

$$\widehat{P}(a_i|f_j) = \#(a_i, f_j)/\#(f_j). \tag{5}$$

The n_F different frames that appear in our scenario have in total n_R distinct associated semantic roles with the following names: *Goal, Theme, Source, Recipient, Requirement, Cognizer, Event, Experiencer, Addressee, Message, Name, Text, Donor, Individuals, Mass theme, Path, Grantee, Action, Direction,* and *Dependent*. These semantic roles are in the following denoted $r_1, ..., r_{n_R}$.

Normally, each frame only has a few semantic roles defined. When parsing an input sentence s, Semafor assigns substrings of s as values to these semantic roles.

According to the suggested approach, parameters for each robot action are related to specific semantic roles. Since the manual identification of parameters in the labeling of sentences not necessarily works by the same principles as the identification of semantic roles in Semafor, a parameter p_i is regarded as *matching* (denoted by the symbol $\sim$) a semantic role r_j if p_i is a nonempty substring of the value of r_j:

$$p_i \sim r_j \equiv p_i \text{ is a nonempty substring of the value of } r_j. \tag{6}$$

Example: Assume that the sentence "Give me the glass" is labeled with action a_1 (i.e. BRING) and parameter $p_1 =$ "glass". Semafor generates a primary frame f_3 (i.e. GIVING), and semantic role r_2 (i.e. Theme) is assigned the value "the glass" for the sentence. Hence, $p_1 \sim r_2$.

In the next section we will construct a classifier to infer expected action a_E for a sentence with a primary frame name f_E. To infer parameters for a_E, we need to estimate the probability that a parameter p_i for a_E matches a semantic role r_j, given that the primary frame is f_E (separate estimates for each $p_i, i = 1, ..., m_a$). With the introduced notation, and by using the definition of conditional probability, this can be written as:

$$P(p_i \sim r_j|f_E) = P(p_i \sim r_j, f_E)/P(f_E). \tag{7}$$

The probabilities on the right-hand-side of (7) can be estimated as follows.

$$\widehat{P}(p_i \sim r_j, f_E) = \#(f_E, p_i \sim r_j)/N \tag{8}$$

and

$$\widehat{P}(f) = \#(f_E)/N \tag{9}$$

where $\#(f_E, p_i \sim r_j)$ denotes the total number of sentences in the training data for which Semafor determines a primary frame f_E and a semantic role r_j, and the sentence was labeled with parameter p_i, satisfying $p_i \sim r_j$. The entity $\#(f_E)$ is the total number of sentences in the training data for which Semafor determines a primary frame f_E. Combining (7−9), yields the following estimation:

$$\widehat{P}(p_i \sim r_j|f_E) = \#(f_E, p_i \sim r_j)/\#(f_E). \tag{10}$$

As described in the next section, the estimated conditional probabilities are used to infer expected action and associated parameters for a given sentence.

4.2 Inference of expected action and parameters

A Bayes classifier is used to infer the expected action a_E for a sentence with a primary frame name f_E and semantic roles $r_i, i = 1, ..., n_R$. It works by inferring the action with highest conditional probability, as given by (1−5):

$$\begin{aligned} a_E &= \arg\max_{1\le i\le n_A} \widehat{P}(Action = a_i|Frame = f_E) \\ &= \arg\max_{1\le i\le n_A} \#(a_i, f_E)/\#(f_E) \\ &= \arg\max_{1\le i\le n_A} \#(a_i, f_E). \end{aligned} \tag{11}$$

[1]In general, the function # denotes the number of observations for with the conjunction of all arguments are true. We simplify the notation as when we denote probabilities, and write for instance a_i instead of *Action*$= a_i$.

Frame \ Labeled Action	BRING	TELL	COLLECT	MOVE	PUT
1 Bringing	7/100%	0/0%	0/0%	0/0%	0/0%
2 Getting	4/100%	0/0%	0/0%	0/0%	0/0%
3 Giving	3/60%	2/40%	0/0%	0/0%	0/0%
4 Needing	2/100%	0/0%	0/0%	0/0%	0/0%
5 Desiring	4/100%	0/0%	0/0%	0/0%	0/0%
6 Telling	0/0%	8/100%	0/0%	0/0%	0/0%
7 Statement	0/0%	8/100%	0/0%	0/0%	0/0%
8 Political locales	0/0%	0/0%	0/0%	0/0%	0/0%
9 Being named	0/0%	0/0%	0/0%	0/0%	0/0%
10 Text	0/0%	1/100%	0/0%	0/0%	0/0%
11 Come together	0/0%	0/0%	4/100%	0/0%	0/0%
12 Amassing	0/0%	0/0%	4/100%	0/0%	0/0%
13 Gathering up	0/0%	0/0%	6/100%	0/0%	0/0%
14 Placing	0/0%	0/0%	2/11%	0/0%	17/89%
15 Motion	0/0%	0/0%	0/0%	14/82%	3/18%
16 Grant permission	0/0%	0/0%	0/0%	0/0%	0/0%
17 Departing	0/0%	0/0%	0/0%	1/100%	0/0%
18 Stimulus focus	0/0%	0/0%	0/0%	0/0%	0/0%
19 Have as requirement	0/0%	0/0%	0/0%	0/0%	2/100%
20 Locale by use	0/0%	0/0%	0/0%	0/0%	1/100%
21 Compliance	0/0%	0/0%	0/0%	0/0%	1/100%

Table 3: Occurrences/frequencies for combinations of primary frames and labeled actions, for the input data used in the experiments. Most rows contains only one non-zero entry, thus supporting the hypothesis that the expected action can be inferred from the frame.

Each one of the parameters $p_i^E, i = 1, ..., m_{a_E}$ required by action a_E is assigned the value of one of the semantic roles $r_i, i = 1, ..., n_R$ for the sentence. The procedure for inference of parameters follows the same principles as for inference of action in (7−10), and parameter values are assigned as follows:

$$p_i^E = r_{opt}, \tag{12}$$

where

$$\begin{aligned} opt &= \arg \max_{1 \le j \le n_R} \widehat{P}(p_i \sim r_j | f_E) \\ &= \arg \max_{1 \le j \le n_R} \#(f_E, p_i \sim r_j) / \#(f_E) \\ &= \arg \max_{1 \le j \le n_R} \#(f_E, p_i \sim r_j). \end{aligned} \tag{13}$$

The inference of expected action and parameters, as described above, is expressed as pseudo-code in Algorithm 1. In steps 5-6, Semafor is used to compute primary frame and semantic role values for the input sentence s. The subset of training sentences with the same primary frame is selected in step 7, such that the computation of the expected action in step 8 corresponds to (11). Values for the parameters p_i^E are computed in steps 11-12, corresponding to (12−13). The algorithm was implemented and evaluated with cross-validation, as described in the next section.

Algorithm 1 Infer expected action a_E and associated parameters p_i^E for an input sentence s.

1: **return** a_E and $p_i^E, ..., p_{m_{a_E}}^E$
2: **inputs:**
3: s : **sentence to be analyzed**
4: A : **set of training sentences labeled with action** a **and parameters** $p_1, ...p_{m_a}$
5: $f_E \leftarrow$ **the primary frame of** s
6: $r_1^E, ..., r_{n_R}^E \leftarrow$ **semantic roles for** s
7: $B \leftarrow$ **the subset of** A **with** f_E **as primary frame**
8: $a_E \leftarrow$ **the most common action** a **in** B
9:
10: **for** $i = 1$ **to** m_{a_E} **do**
11:     **find the index** opt **for which** $p_i \sim r_{opt}$ **in most sentences in** B
12:     $p_i^E \leftarrow r_{opt}^E$
13: **end for**

4.3 Evaluation

The developed system was evaluated using the full data set of 94 sentences. Evaluation was done by leave-one-out cross-validation, i.e. one sentence was left out of the training data set, and a model was constructed as described in Section 4.1. The model was evaluated by inferring expected

Table 4: Examples of sentences used for training and evaluation. Each sentence is labeled with expected action and associated parameter(s).

	Sentence	Expected action	p_1	p_2
1	move the chairs to the kitchen	PUT	chairs	the kitchen
2	Move 2 meters to the left	MOVE	2 meters	to the left
3	I want a glass of water.	BRING	a glass of water	
4	Robot, tell Ola the name of the book.	TELL	the name of the book	Ola
5	stash the balls in the wardrobe.	PUT	the balls	in the wardrobe
6	package all glasses into nice parcels.	PUT	all glasses	into nice parcels
7	Gather all the green balls.	COLLECT	all the green balls	
8	Robot, tell Ola the color of the ball.	TELL	the color of the ball	Ola
9	Gather dust in the room.	COLLECT	dust	in the room
10	Go to the tire storage.	MOVE	the tire storage	
11	Robot, tell the direction of the exit to me.	TELL	the direction of the exit	me
12	Bring Ola's book to me.	BRING	Ola's book	me

Table 5: Semantic parses of the sentences in Table 4, as given by the Semafor system. The table shows primary frame name and some of the generated semantic roles for the frame.

	Primary frame	Semantic role/value	Semantic role/value	Semantic role/value
1	MOTION	Theme/the chairs	Goal/to the kitchen	
2	MOTION	Theme/2 meters	Goal/to the left	
3	DESIRING	Experiencer/I	Event/a glass of water	
4	TELLING	Speaker/Robot	Addresse/Ola	Message/the name of the book
5	PLACING	Theme/the balls	Goal/in the wardrobe	
6	PLACING	Theme/all glasses	Goal/into nice parcels	
7	COME TOGETHER	Individuals/all the green balls		
8	TELLING	Speaker/Robot	Addresses/Ola	Message/the color of the ball
9	COME TOGETHER			
10	MOTION	Goal/to the tire storage		
11	TELLING	Speaker/Robot	Addresses/to me	Message/the direction of the exit
12	BRINGING	Theme/Ola's book	Goal/to me	

action and parameters for the held out sentence, as described in Section 4.2, and the procedure was then repeated 93 times such that all sentences were left out once from the training. Performance figures were computed as the average performance for all 94 training/evaluation sessions.

5. Results

In order for a robot to be able act correctly on an uttered sentence, both action and parameters have to be correctly inferred. We present results for both these tasks in Table 6. Each row in the confusion matrix shows how sentences with a specific labeled action leads to inference of various actions, shown in separate columns. Cases where no inference of action was possible are shown in the column labeled "?". At the end of each row, the accuracy for combined action and parameter inference is shown. E.g., sentences labeled with the TELL action leads in 2 cases (11%) to an incorrectly inferred BRING action, and in 16 cases (84%) to a correctly inferred TELL action. For one sentence labeled with a TELL action, no action could be inferred. The reason was that the primary frame (the TEXT frame) for this sentence occurred only once in the whole data set (see Table 3), and hence not at all in the training set for that specific sentence. Hence, no inference was possible for that sentence. The combined inference of both action and parameters, for all sentences labeled with a TELL action, was correct in 14 cases (74%). As a whole, the non-zero entries are gathered on the diagonal, which means that the inferred actions equals the labeled actions. The average accuracy for all sentences for inference of action was 88%, and for combined inference of action and parameters 68%.

6. Discussion

The proposed method builds on the hypothesis that expected actions can be inferred from shallow semantic information. We conclude that the hypothesis was valid for more than 88% of the tested sentences. Expected actions and parameters were correctly inferred for 68% of the cases.

Table 6: Confusion matrix showing number of cases/percentages for inference of expected robot actions. Figures for inference of both actions and associated parameters is shown in the right-most column. Each row contains results for sentences with one specific labeled action.

Labeled \ Inferred	BRING	TELL	COLLECT	MOVE	PUT	?	Accuracy
BRING	20/100%	0/0%	0/0%	0/0%	0/0%	0/0%	14/70%
TELL	2/11%	16/84%	0/0%	0/0%	0/0%	1/5%	14/74%
COLLECT	0/0%	0/0%	14/88%	0/0%	2/13%	0/0%	10/62%
MOVE	0/0%	0/0%	0/0%	14/93%	0/0%	1/7%	9/60%
PUT	0/0%	0/0%	0/0%	3/13%	19/79%	2/8%	17/71%

Given the large variety of sentences, and the small data set being used, the result is considered both surprising and promising. Better results can be expected by adding more data. One specific problem with limited data was discussed in the previous section: if a frame occurs only once in the data set, it is not possible to infer the expected action for that sentence since it is removed as part of the cross-validation process. Extending the data such that there are at least two sentences for each frame name, would clearly improve performance. By removing the four sentences for which the situation occurs in our data, the accuracy for inference of action improves to 92%, and for combined inference of actions and parameters to 71%.

The proposed method relies, to a very large extent, on the quality of the semantic labeling, which in our case was performed by the Semafor system. While identified frames and semantic roles do not necessarily have to be linguistically "correct", they should be consistent in the sense that semantically similar sentences should give the same results. This is unfortunately not the case with the online version of Semafor that we have been using (the downloadable version behaves somewhat differently but not better in this respect). Not only does it fail in the sense described above, but also by producing vastly different results depending on capitalization of the first letter in the sentence, and on whether the sentence is ended by a period or not. As an example, adding a period to sentence 1 in Table 4 results in a replacement of the semantic role *Goal* with *Building subparts* (also see Table 5). Another example is the sentence "Bring Mary the cup.", with varying results depending on both punctuation and replacement of "Mary" by "me". Due to such experienced problems with the Semafor system, the sentences used in the reported experiments were manually selected to ensure that the automatic semantic analysis was reasonable and consistent. This is clearly a concern for practical usage and continued research on the proposed method, but was outside the scope of the present work.

7. Future work

Since the results of the proposed method depends heavily on the quality of the semantic parsing, alternative approaches in which syntax and semantics are treated simultaneously [15] will be investigated.

As part of the inference process, the parameters for an expected action are bound to the values of certain semantic roles (12). These values are substrings like "the green ball", and "all my books" and have in this work not been further analyzed, but rather assumed to be properly interpreted by the pre-programmed action routines. This is definitely not a trivial task and contains several hard problems. The parameters are typically noun phrases, that have to be semantically analyzed and grounded to objects that the robot can perceive. This task will be a major and important part of the continued work.

References

[1] G. Bugmann, J. Wolf, and P. Robinson, "The impact of spoken interfaces on the design of service robots," *Industrial Robot: An International Journal*, vol. 32, no. 6, pp. 499–504, 2005.

[2] M. Scheutz, P. Schermerhorn, J. Kramer, and D. Anderson, "First steps toward natural human-like hri," *Autonomous Robots*, vol. 22, no. 4, pp. 411–423, 2007.

[3] J. M. Siskind, "A computational study of cross-situational techniques for learning word-to-meaning mappings," *Cognition*, vol. 61, no. 1, pp. 39–91, 1996.

[4] D. Das, N. Schneider, D. Chen, and N. A. Smith, "Probabilistic frame-semantic parsing," in *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*. Association for Computational Linguistics, 2010, pp. 948–956.

[5] C. Meriçli, S. D. Klee, J. Paparian, and M. Veloso, "An interactive approach for situated task teaching through verbal instructions," in *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[6] E. Chuangsuwanich, S. Cyphers, J. Glass, and S. Teller, "Spoken command of large mobile robots in outdoor environments," in *Spoken Language Technology Workshop (SLT), 2010 IEEE*. IEEE, 2010, pp. 306–311.

[7] A. Weitzenfeld, A. Ejnioui, and P. Dominey, "Human robot interaction: Coaching to play soccer via spoken-language," *IEEE/RAS Humanoids*, vol. 10, 2010.

[8] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Experimental Robotics*. Springer, 2013, pp. 403–415.

[9] S. Bensch and T. Hellström, "Towards proactive robot behavior based on incremental language analysis," in *Proceedings of the 2014 Workshop on Multimodal, Multi-Party, Real-World Human-Robot Interaction*. ACM, 2014, pp. 21–22.

[10] R. Cantrell, M. Scheutz, P. Schermerhorn, and X. Wu, "Robust spoken instruction understanding for HRI," in *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*. IEEE, 2010, pp. 275–282.

[11] C. J. Fillmore, C. R. Johnson, and M. R. Petruck, "Background to FrameNet," *International Journal of Lexicography*, vol. 16, no. 3, pp. 235–250, 2003.

[12] P. Kingsbury and M. Palmer, "From TreeBank to PropBank," in *LREC*. Citeseer, 2002.

[13] D. Das, D. Chen, A. F. Martins, N. Schneider, and N. A. Smith, "Frame-semantic parsing," *Computational Linguistics*, vol. 40, no. 1, pp. 9–56, 2014.

[14] C. Baker, M. Ellsworth, and K. Erk, "SemEval'07 task 19: frame semantic structure extraction," in *Proceedings of the 4th International Workshop on Semantic Evaluations*. Association for Computational Linguistics, 2007, pp. 99–104.

[15] S. Bensch, F. Drewes, H. Jürgensen, and B. van der Merwe, "Graph transformation for incremental natural language analysis," *Theoretical Computer Science*, vol. 531, pp. 1–25, 2014.

Efficiency Considerations of an Offline Mobile Robot Path Planner

Pejman Kamkarian
Department of Electrical and Computer Engineering,
Southern Illinois University, Carbondale, IL 62901, USA

Henry Hexmoor
Department of Computer Science,
Southern Illinois University, Carbondale, IL 62901, USA

***Abstract*— The aim of disseminating this research article is to showcase a novel path planner method, which is shown to be an efficient offline path planner in terms of its capacity for analyzing workspace robot maneuvering skills and constructing collision-less free trajectories that yield the shortest possible path from initial point toward goal configuration. The determined route is considered to be adequately secure such that it enables the mobile robot to maneuver among obstacles in the workspace without dangers of encountering a near miss. In addition, this paper evaluates our novel path planner algorithm abilities and skills by examining it against different workspaces. We assess our novel path planner by comparing it to two of the most popular planners with the purpose of revealing its capability to route trajectories in regards to building optimal trajectory distances from initial to the goal configurations.**

***Keywords*— Path planning, Rapidly Optimizing Mapper, Robot path planning, Robot trajectory builder**

I. Introduction

A regular robot that is capable of performing an assigned task typically consists of many units that cooperate together for the sole purpose of enabling it to successfully achieve its missions. As an important functionality that plays a vital role for a robot to function appropriately, robust path planning is essential. The central task of a path planner is to analyze the robot's surrounding using equipped sensors and plan a secure and reasonable trajectory that guarantees a safe traversal for the robot from initial point to the goal configuration. The concept of security for the calculated path is commonly understood to be on a collision-less less trajectory that the path planner determines corresponding to the accuracy and reliability of detecting objects around the robot while moving toward its goal. The traversal distance of a path that is computed by the path planner component is directly related to the methodology employed by the planner when it composes the trajectory. Since the last few decades, we have witnessed several diverse methods proposed for path planning where each has its own advantages and disadvantages. The path planner is considered reliable in regards to planning a secure path when it is demonstrates that it maintains generous distances between the robot and every obstacle in the workspace. The objective for safety is often at odds with the objective for constructing an optimal path in terms of planning the shortest possible collision-less less trajectory from start point to the goal configuration.

Earliest reported work on robotic path planners has been the Potential Field planning method proposed in [2], and [13]. The Potential Field path planner employs the concept of virtual electromagnetic fields in the workspace modeled by a simulated attraction force towards desired points (i.e., goal/destination points) as well as repulsive forces from undesirable points (i.e., points occupied by obstacles). Each of these forces are simulated by a vector that captures the direction and magnitude of the force. Whereas the goal point vectors continuously exert pulling force for the robot, vectors corresponding to obstacles exert pushing away forces. At any point during path planning, the trajectory is adjusted to coincide with the result of reconciling the cumulative sum of applicable vector forces and directions. An impulse movement along the suggested path moves the robot to the next consecutive point along the trajectory. The magnitude of the impulse step is a parametric value corresponding to the path granularity. An overall trajectory for a pair of start and finish points is the accumulation of consecutive impulse moves. The process of path planning through Potential Field method guarantees a collision-less less trajectory from initial to goal configurations. However, due to the electromagnetic fields' constraints, this method performs poorly in certain scenarios. For example local minima is seen that causes when the robot becomes trapped in U-Shape obstacles (i.e., box canyons). There are other planner issues such as obstacles that leave narrow passage ways creating erratic trashing forces that can be either redundantly cyclic or contradictory forces. Latter problems often lead to impasse phenomenon [5], [11], [14]. Many papers have proposed different solutions by updating the original algorithm or combining different method with the Potential Field planner construction to remedy specific problems [3], [9], [7], [10], [19]. The Rapidly-exploring Random Tree is a sampling based mapping technique that solves non-holonomic constraints and it was introduced in [15]. Information-rich Rapidly-exploring Random Tree proposed in [16] is the extension of the Rapidly-exploring Random Tree, which is able to maneuver more efficiently to build the trajectory in workspaces with the presence of different constraint domains such as complex moving agent dynamics and moving robots sensor limitations in terms of resolution and narrow the detection view site. Several researchers have developed hybrid solutions, and hence, several studies and approaches related to the hybrid path planners are reported in [1], [4], [6], [12], [17], [18], and [20]. A hybrid path planner typically takes advantages from multiple path planning strategies, which are combined into a unique algorithm. Hybrid planners are promising to address more efficient path planning in order to elevate the quality and accuracy of the generated trajectories and also overcome various constraints and situations that can affect the traditional path planners in analyzing and determining optimal possible trajectories. In the next section, we illustrate the general concepts and methods that our planner employs to produce an optimal trajectory. In order to evaluate the performance of our planner, in subsequent section three, we will compare it with two other path planners by applying them on two sample workspaces. This paper further explores efficiency issues for our Rapidly Optimizing Mapper (ROM).

II. FOUNDATIONS AND FUNCTIONS OF OUR ROM FRAMEWORK

We have fabricated our planner on the premise of a multi-layer approach in the form of a unique algorithm such that each layer uses data provided from the prior layer and is responsible to generate the needed information for the following level. Each layer is also treated as a phase of a sequentially phased system that provides data for the next phase using information processes in the previous phase. Our ROM planner algorithm is constructed based on five general levels along with initial and final phases indicating as follows: Initializing phase, Workspace analyzer, Graph builder, and Shortest path calculation unit. Each of the mentioned phase along with their objectives is detailed in the remainder of this section.

Initialization phase: This phase is achieved by adjusting values for variables that are salient constituents for building a trajectory. We considered the key feature of path security, which has a direct bearing on the robot maneuvering skills. Consideration for this element has to be determined at the initial phase as the primitive value which helps the planner to construct proper trajectories. The Standoff Distance, (SD) is our main path security parametric variable that is defined to be the width of a virtual buffer zone around perimeters of obstacles in order to specify a safety area for pathways within which allows the robot to perform a near miss avoidance transition toward its assigned goal. The security channel width is determined based on robot sensors accuracy specifications. The more sensitive obstacle detection equipment the robot possesses, the lower security consideration required for the width of the safety channel. Each robot, based on the type of mission and the terrain specifications, is equipped with different capabilities such as actuators and sensors that equip it to move around and detect objects in the environment and thereby adjusts its path toward the determined trajectory instructed by the path planner.

Workspace analysis phase: This phase of ROM is responsible for analysis of the workspace obstacles to determine roadblock obstacles as well as roadblock obstacles side edge nodes generation. The roadblock obstacles will be recognized by considering virtual straight lines from valid accessible nodes (i.e., safe obstacle boundary points) toward goal configuration. The valid nodes are in the form of a group of certain nodes starting from the initial configuration and ending with the goal location along with the group of roadblock side edge nodes. As it is illustrated in figure 1, the workspace analyzer phase obtains the first group of roadblock side edge nodes by considering straight rays from candid nodes toward goal configuration. Any obstacle that shares intersecting points with the start-goal line in at least one hit point is classified as a roadblock obstacle. The number of detected roadblock obstacles will vary based on the number of obstacles intersecting with the start-goal straight line. The best scenario occurs when there are no roadblock obstacles in the workspace. In such a situation, the optimal trajectory will be considered to be the straight line from start point to the goal configuration.

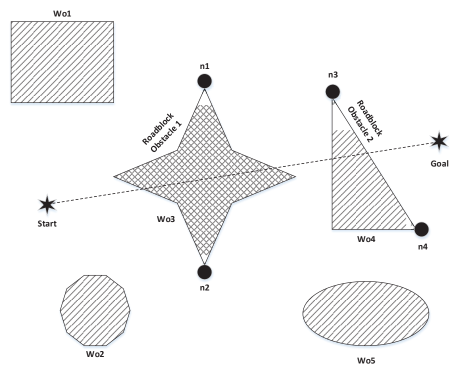


Figure 1. A sample *Roadblock obstacles* side edge nodes generated from the workspace analyzer unit

Graph builder: The main objective of this phase of ROM is to form a complete graph (i.e., a single connected component) consisting of roadblock side edge nodes combined with the start and goal configuration points. In order to achieve this goal, ROM enforces multiple steps. As the first step, the planner connects roadblock side edge nodes together to form a primitive graph of trajectories so as to enable maneuvering around the perimeter of the obstacle. The next step is to examine roadblock side edge nodes in order to recognize isolated nodes and to adjust them. Isolated nodes will be considered based on roadblock side edge nodes belonging to a single obstacle that are not yet connected to one another that do not go across the surface of the obstacle and remain entirely at one possible contiguous side of the obstacle. In other words, the path planner at this processing stage examines side edge nodes of each single roadblock obstacle to assure that they trail each other contiguously and steer clear of the surface of the obstacle. We use this technique to enable our planner to consider all possible paths crossing from roadblock side edge nodes with the purpose of elevating the planner ability to consider all possible paths toward goal and increasing the accuracy in determination of the shortest possible trajectory toward goal configuration. To obtain the best results in terms of refining the shortest possible trajectories, the path planner, as the next step, simplifies the paths via removing nodes that are located in between pairs of visible nodes. Visible nodes will be recognized if there is a possibility to connect two nodes through a straight path without intersecting any obstacles in the workspace. The last step of this phase of the planner consists of adjusting Euclidean distances for the remaining pairs of nodes that are already processed. The final result of this phase of ROM is a complete graph including start and goal points. Depending on the specifications of elements in the workspace, such as the number, size, shape and locations of obstacles, the pattern of the lattice and hence, the graph that forms through the graph builder phase will vary. Different scenarios result in having different numbers of paths from start to goal configurations.

Shortest path computation phase: This phase of our planner adopts the Dijkstra algorithm [8] in order to refine the shortest trajectory from start point toward goal configuration. Our planner uses the graph, which is generated at the previous phase of the planner as input data. The mentioned graph consists of all possible pathways that are optimized through the path planner optimization steps and eventually constructed as a form of graph. The predominant task of this phase is to examine all available paths in the graph and subsequently produce the best possible trajectory consisting of roadblock side edge nodes as the optimal path. The output of this phase is the final result of the planner in generating the trajectory, which is the optimal single collision-less less path from start point to the goal configuration.

III. Experiments and evaluation of ROM

In order to assess our planner performance, we compare it with the two other path planners including Potential Field and Rapidly-exploring Random Tree path planners. The process of evaluation is conducted by applying our planner as well as the mentioned path planners on two exemplar workspaces that are both illustrated in figure 2.

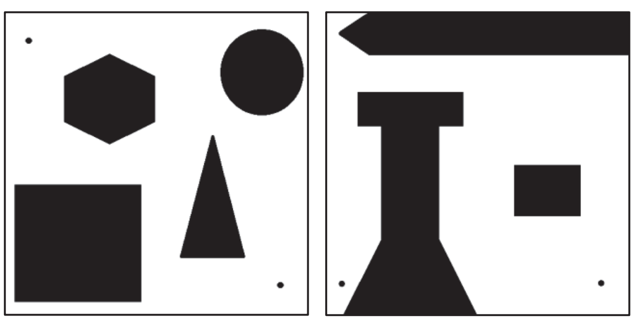


Figure 2. Left: The first candidate environment for applying path planners. Right: The second candidate workspace

The first map illustrated in figure 2, (left), consists of four obstacles with different polygonal geometric shapes and locations whereas figure 2, (right), consists of three obstacles. As seen in the figure 2, we considered a narrow distance between obstacles with the purpose of evaluating the skills of path planner algorithms to analyze and determine the optimal trajectories in terms of distance from start to goal configurations and the security of the constructed path. Both workspaces are considered to have a same dimensions of 500 by 500 units in Euclidean measurement system for X and Y coordinates. The vertical axis spans from up to down and the horizontal axis emanates from left to right. The initial point for the first workspace is considered at (50, 50) coordinates and the goal location is at (450, 450) coordinates. The start location for the second environment is at (25, 450). The goal for the second workspace is also located at (450, 450) coordinates. The following figures 3 illustrate the resultant trajectory from applying out path planner on the sample workspaces.

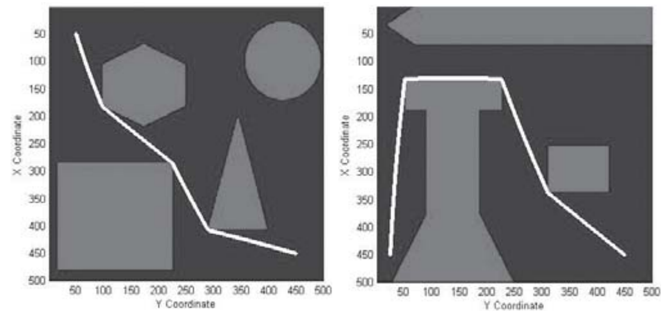


Figure 3. Left: The resultant trajectory from applying ROM on the first workspace. Right: The resultant path from applying ROM on the second environment

The optimal path, which is constructed using our planner algorithm is shown as a bright path trajectory starting from initial point to the goal configuration. The Euclidean distance from start location to the goal configuration is calculated to be 709.12 units for the first environment and 1018.43 units long for the second workspace. The trajectory resultant of applying our novel path planner shown in both workspaces in figure 3 indicates that our planner is able to route a collision-less less path, successfully. Moreover, neither obstacle complexity in terms of shapes nor the distances between obstacles could permit the planner to route the best possible trajectory in terms of the safety and the length toward goal. This is because our planner considers all possible valid directions around each roadblock obstacle to achieve the best results in determining the optimal trajectory. In addition, our planner benefits from using nontrivial strategies to reconstruct the generated graph in the early stages of its algorithm with the purpose of recognizing the best candidates among all possibilities of different routes and generating worthwhile trajectories, regardless of constraints posed by different scenarios in workspaces.

In order to compare the performance of ROM path planner skills with other planners, we employed two path planners in offline mode, specifically Potential Field path planner and Rapidly-exploring Random Tree algorithm. Each planner is applied on the sample workspaces in the form of a different case study and the results of each scenario are discussed in detail. Through case study I, the process of planning trajectories using Potential Filed algorithm will be analyzed.

Case study 1:

The Potential Field algorithm performs the trajectory based on considering start and goal points as well as obstacles as electromagnetic charges and fields. The goal point has the most attractive power (i.e., exerting attraction force) among other objects in the workspace, whereas obstacles repel (i.e., exert pushing way force) the planner path finder away from them.

The Potential Field planner method benefits this strategy to build a collision-less less trajectory. The following figure 4 shows the result of the path generated by applying the Potential Field algorithm on the sample workspaces.

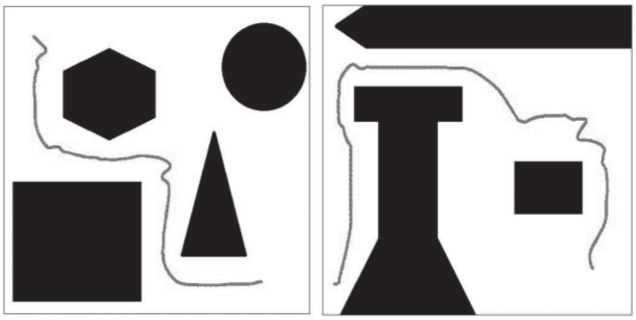


Figure 4. Left: The resultant trajectory from applying the Potential Field path planner on the first workspace. Right: The resultant path from applying the Potential Field on the second environment

The constructed trajectory resultant from applying the Potential Field method on sample workspaces are considered in a dark path starting from start into goal configuration. The trajectory length from start point to the goal configuration for the first map is equal 984.16 units based on Euclidean distance measurement. The Potential Field planner is also computed the path length for the second environment to be 1984.31 units long in Euclidean distance measurement. As it is evident in both maps in figure 4, the final generated trajectories consist of several curvatures. This event can be explained according to the similarity of simulated charges between the trajectory and obstacles. In addition, the rate of severe curves increases when the path is crossing from sharp obstacle edges nearby. This is because of forming electromagnetic fields with different intensities around sharp edges of obstacles that cause the Potential Field planner to continuously adjust the path based on different amounts of repulsion forces around the mentioned areas. Also, in the second workspace illustrated in the figure 4, (right), the planner was not able to consider the trajectory with shorter route. This is because there exists many adjacent sharp edges between two T-Shape and square obstacle that push the planner to stay out of the pathway crossing from the shorter side toward goal configuration. Comparing constructed trajectories in both workspaces through the Potential Field planner reveals that the planner is able to route a collision-less less trajectory in both environments. It is, however, suffering from the side effects of electromagnetic fields forming around nearby obstacles, especially around obstacles sharp edges surroundings to consider the optimal paths in terms of length and hence reduces the performance of the Potential Field planner.

The next case study is dedicated to examining the Rapidly-exploring Random Tree path planner on the sample workspaces to evaluate its performances on determining the optimal trajectories.

Case study 2:

The Rapidly-exploring Random Tree method works based on forming random trees consisting of a group of arbitrary sample points located outside of the obstacles in the workspace. The planner algorithm will then examines all possibilities of branches that form randomly around the main stem of the generated trees at each moment during the path generation process and gradually selects the best collision-less less matches in terms of the length and security, as sub-optimal trajectories among them. The planner constructs the final path from considering all optimal sub-trajectories obtained in the previous level. The following figure 5 demonstrates the resultant trajectory determined from the application of the Rapidly-exploring Random Tree on two sample workspaces.

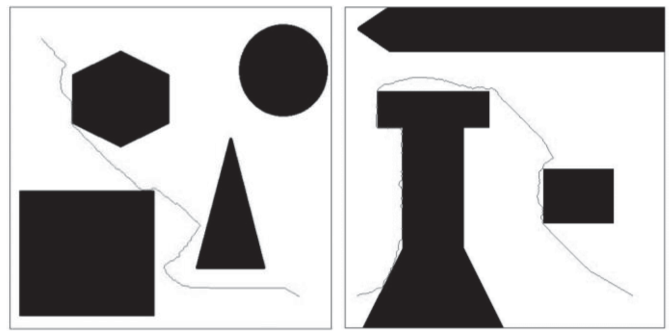


Figure 5. Left: The resultant trajectory from applying the Rapidly-exploring Random Tree planner on the first workspace. Right: The resultant path from applying the Rapidly-exploring Random Tree algorithm on the second environment

The Euclidean distance from the start configuration for the trajectory length resultant from applying the Rapidly-exploring Random Tree on the first workspace is calculated to be 865.52 units for the first scenario. The Rapidly-exploring Random Tree is also determined to be 1253.89 units long Euclidean distance from start point for the second workspace. Because the nature of the RRT planner algorithm in forming random nodes to explore the workspace surroundings and to refine the best matches, we observe that the planner constructs trajectories that are slightly different from each other at every run. Benefiting the techniques of using random nodes leads the planner to achieve collision-less less trajectories that are close to the optimal trajectories in most cases. It is, however, still is not able to reach the shortest possible collision-less less trajectories due to lack of existing proper methods to shortened the final generated path in the planner construction. As it can be examined in figure 4, (right), the issue that is addressed above exhibits a larger effect in accuracy reduction to determining the shortest possible path in the area between the T-Shaped and the square obstacles.

In order to clarify the best results in trajectory lengths, we collected all results that our ROM planner as well as other path planner candidates achieved in a single trajectory length chart as shown in the following figure 6:

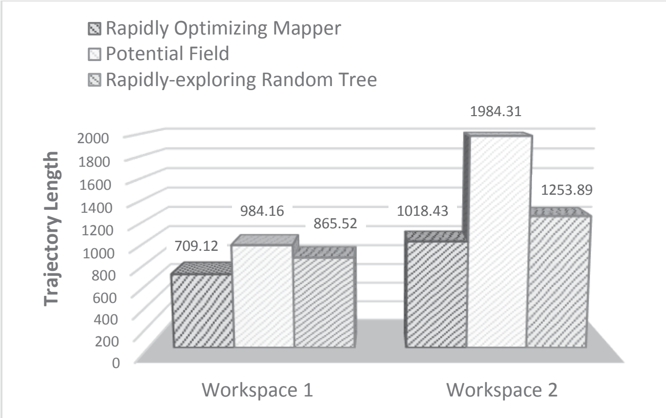


Figure 6. The trajectory distance chart of computer path for the Rapidly Optimizing Mapper as well as Potential Field and Rapidly-exploring Random Tree path planners

Comparing trajectory distances illustrated in the figure 6 reveals that our planner is able to route the most efficient trajectories in terms of shortest paths among other candidates. In other words, Rapidly Optimizing Mapper exhibits a higher performance for building collision-free trajectories in terms of distances from start points to the goal configurations. Employing the Potential Field algorithm to route trajectory in workspaces with close obstacles and sharp edges reduces the performance of the planner dramatically as it illustrated in the second workspace trajectory rate in figure 6. In order to overcome the mentioned environmental constraints, our planner benefits from specific techniques to optimize the calculated paths during the process of planning helps our planner to take all possible paths into account for the planning operation.

IV. CONCLUSION AND PERSPECTIVES

A novel path planner termed ROM has been elaborated within this research article. We demonstrated our novel path planner specifications and abilities by illustrating its constituent components. In order to validate the performance of our planner, we considered sample workspaces with complex constraints that contain a variety of different shapes and locations to apply and evaluate the planner performances for building trajectory skills on different scenarios. In order to clarify our novel path planner's strength for determining the ultimate trajectories, we compared it with two of the best known path planners through applying them on the same sample workspaces. Based on the obtained results, we conclude that ROM is able to compute the optimal trajectories in terms of path length more efficiently. This is because we adopted techniques to furnish our planner with the intention of elevating its abilities to operate overcoming a variety of different constraints on workspaces elements specifications. Our future target is to examine our novel path planner on workspaces with more constraints along with upgrading its structure to heighten its efficiency to act limitlessly in any types of environment, flawlessly.

REFERENCES

[1] Abinaya, S., Hemanth Kumar, V., Srinivasa Karthic, P., Tamilselvi, D., and Mercy Shalinie, S., "Hybrid Genetic Algorithm Approach for Mobile Robot Path Planning", In Journal of Advances in Natural and Applied Sciences, Vol. 8, No. 17, Special 2014, pp. 41-47, 2014.

[2] Andrews, J.R. and Hogan, N., "Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator." Control of Manufacturing Processes and Robotic Systems, Eds. Hardt, D. E. and Book, W., ASME, Boston, pp. 243-251, 1983.

[3] Arámbula Cosío F. and Padilla Castañeda M.A., "Autonomous Robot Navigation using Adaptive Potential Fields", In Mathematical and Computer Modelling, Vol. 40, Issue 9-10, pp. 1141-1156, 2004.

[4] Bashra K., and Oleiwi and Hubert Roth, "Modified Genetic Algorithm Based on A* Algorithm of Multi Objective Optimization for Path Planning", In Journal of Automation and Control Engineering, Vol. 2, No. 4, pp. 357-362, 2014.

[5] Borenstein J. and Koren Y., "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots", In IEEE Transactions on Robotics and Automation 7, pp. 278-288, 1991.

[6] Chaari, I., Koubaa, A., Trigui, S., Bennaceur, H., Ammar, A., and Al-Shalfan, K., "SmartPATH: An Efficient Hybrid ACO-GA Algorithm for Solving the Global Path Planning Problem of Mobile Robots", In International Jounal of Advanced Robotics Systems, pp. 1-15, 2014.

[7] Connolly C.I., Burns J.B., and Weiss R., "Path Planning using Laplace's Equation", In Proceedings of the lEEE Conference On Robotics and Automation, Cincinnati, OH, pp. 2102-2106, 1990.

[8] Dijkstra, E.W., "A Note on Two Problems in Connexion with Graphs", In Numerische Mathematik 1, pp. 269–271, 1959.

[9] Ge S.S. and Cui Y.J., "New Potential Functions for Mobile Robot Path Planning", In IEEE Transactions on Robotics and Automation, Vol. 16, No. 5, pp. 615-620, 2000.

[10] Guldner J., Utkin V., and Hashimoto H., "Robot Obstacle Avoidance in N-Dimensional Space using Planar Harmonic Artificial Potential Fields", In Journal of Dynamic Systems, Measurement, and Control, Vol. 119, pp. 160-166, 1997.

[11] Grefenstette J. and Schultz A.C., "An Evolutionary Approach to Learning in Robots", In Proceedings of the Machine Learning Workshop on Robot Learning, International Conference on Robot Learning, New Brunswick, N.J., pp. 65-72, 1994.

[12] Ju, M., Wang, S., and Guo, J., "Path Planning using a Hybrid Evolutionary Algorithm Based on Tree Structure Encoding", In The Scientific World Journal, Vol. 2014, Article ID 746260, 8 pages, 2014.

[13] Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", In IEEE International Conference on Robotics and Automation, St. Louis, Missouri, pp. 500-505, 1985.

[14] Koren Y., and Borenstein J., "Potential Field Methods and Their Inherent Limitations for mobile robot navigation", In Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1398-1404, 1991.

[15] LaValle, S.M., "*Rapidly-exploring Random Tree*: A New Tool for Path Planning", Computer Science Department, Iowa State University, TR 98-11, 1998.

[16] Levine, D., Luders, and B. How, J.P., ""Information-rich Path Planning with General Constraints using *Rapidly-exploring Random Tree*.", In American Institute of Aeronautics and Astronautics (AIAA), Atlanta, Georgia, AIAA 2010-3360, 2010.

[17] Loo, C.K., Rajeswari, M., Wong, E.K., and Rao, M.V.C., "Mobile Robot Path Planning using Hybrid Genetic Algorithm and Traversability Vectors Method", In Intelligent Automation and Soft Computing, Vol. 10, No. 1, pp. 51-64, 2004.

[18] McFetridge, L. and Ibrahim, M.Y., "New Technique of Mobile Robot Navigation using a Hybrid Adaptive Fuzzy-Potential Field Approach", In Computers and Industrial Engineering, Vol. 35, Issues (3–4), pp. 471–474, 1998.

[19] Utkin V.I., Drakunov S., Hashimoto H., and Harashima F., "Robot Path Obstacle Avoidance Control via Sliding Mode Approach", In Proceedings

of the lEEE/RSJ International Workshop on Intelligent Robots and Systems, Osaka, Japan, pp. 1287-1290, 1991.

[20] Yao., Z. and Ren, Z., “Path Planning for Coalmine Rescue Robot based on Hybrid Adaptive Artificial Fish Swarm Algorithm”, In International Journal of Control and Automation, Vol. 7, No. 8, pp. 1-12, 2014.

Architecture for Multi-robot Systems with Emergent Behavior

Angel Gil[1], Jose Aguilar[2, 3], Rafael Rivas[2], Eladio Dapena[2], Kevin Hernandez[1]
[1]Universidad del Táchira, Laboratorio de Prototipos. Tachira, Venezuela
{agil,kevin.hernandez}@unet.edu.ve
[2]Universidad de Los Andes, Facultad de Ingeniería, CEMISID, LASDAI. Mérida, Venezuela
{aguilar,rafael,eladio}@ula.ve
[3]Investigador Prometeo, Universidad Técnica Particular de Loja, Ecuador

Abstract - *This paper describes the design of an architecture for multi-robots systems with emergent behavior. The platform must manage the dynamics in the system, so to enable the emergence therein. The architecture is divided into three levels. The first level provides local support to the robot, manages its processes of action, perception and communication, as well as its behavioral component. The behavioral component considers the reactive, cognitive, social and affective aspects of a robot, which influence its behavior and how it interacts with the environment and with the other robots in the system. The second level provides support to the collective processes of the system, as are the mechanisms of cooperation, collaboration, planning, and/or negotiation, which may be needed at any given time. This level of the architecture is based on the emerging coordination concept. The third level is responsible by the knowledge management and learning processes, both individually and collectively, that are occurring in the system.*

Keywords: multi-robots architecture, emergent behavior, multi-robots

1 Introduction

Multi-robots systems are used today in different tasks, particularly when using a single robot is not enough, or the task is not efficiently executed. In such systems several phenomena of interest to investigate are presented, such as studying the interactions between robots, and how behaviors can emerge in the system in an unstructured environment, with tasks, functions or objectives unallocated explicitly. The architecture proposed in this paper aims to support these systems, in which are emerging tasks, functions or objectives to be met by robots. It consists of three levels, a level for each robot, which consists of three modules: acting, perception and behavior, another level for the entire collective part, formed by a layer of emerging coordination; and a final level of knowledge management and learning processes. In particular, the collective level manages the processes of cooperation and/or collaboration in the system as well as the planning of agents.

In previous research, such as [1, 2, 3], proposals involving multi-robots systems with emergent behavior are presented, applied to problems such as robotic soccer, cooperative hunting and food search. However, generally homogeneous robots are used both in hardware and software, and for very specific tasks.

Our architecture doesn't arise from such assumptions, it presents a new structure for the characterization of the processes involved in the system, where they are managed by specific modules of collective and individual level to allow emergence; also, it implements a module of basic emotions in robots, which directly affects their behaviors in the environment and their interactions with the system.This article is divided into four sections, the first summarizes some initial considerations that contextualize the research, the second one presents the design of the proposed architecture. In the third one, a study case that allows looking at the operation of the architecture in an emergent process is presented, and finally, some final considerations regarding the research are presented.

2 Initial considerations

This section describes some concepts that help to contextualize the research presented in this paper.

2.1 Emergent Behavior

[4] is based on the premise that if a robot wants to have some autonomy in their operation, it should be able to display some basic behaviors, defining behavior from the biological point of view, as the processes that are carried out by the individual from the information taken from the environment and its own internal state, to eventually respond to the perceived changes. Furthermore, to do this it must recognize and interpret observed patterns of the physical phenomena occurring in its environment.

They describe from a functional point of view, some primitive behaviors for mobile robots, for example:

Evasion: behavior that allows a robot to avoid a collision, and thus preserve its integrity.

Attraction: towards an object, another robot, an environment.

From activation, concurrent or not, of these behaviors in different forms, can emerge more complex behaviors, such as: avoidance of an object, simple navigation, perimeter monitoring, among others.

On the other hand, in [5] emergent behavior is defined as that one which is not attributed to any individual in the group, but arises from the dynamics of the members of the system of which is part the individual.

In nature, there are multiple examples of living beings societies, which individually are simple organisms, and when they unite and work together a complex behavior can be observed, which arises from the interaction of individuals of the System. Ant colonies represent a clear example of occurrence of this type of behavior. In general, an emerging system is a complex system, adaptable, where behaviors that cannot be explained by explicit rules of operation appear [Aguilar, 2015].

2.2 Multi-robots Systems

[6] defines multi-robot systems as homogeneous or heterogeneous teams of robots, which are used in tasks where a single robot is not enough, or when the team becomes more efficient execution thereof. It generally has the following characteristics: cooperation, communication and coordination.

The emergent behavior of insect societies has inspired multi-robot systems, creating a whole area of research called swarm of robots. Swarms of robots usually are formed by simple individuals, and are designed to mimic the behavior of swarms of insects, thereby seeking to have robust and adaptable low-cost multi-robots systems.

3 Description of the proposed architecture

The architecture consists of three levels, where each one has specific tasks, but their integration is what allows giving support to emerging processes. Emergence is possible, since the three levels provide through their components the required elements: deployment of interactions, management of distributed processes, management of local decision rules, mechanisms of local and collective learning, and shared memory spaces. The levels of the architecture are (see Figure 1).

- **Collective level**

Supports the processes of interaction of the robots with others individuals within the system and with the environment. It has mechanisms that enable emerging processes of coordination among robots.

- **Individual level**

Provides support to the individual processes of the robot; manages the perception abilities of each robot, control of its actuators and their local decision-making, as well as aspects related to the robot's conduct and *emotions.*

- **Level of knowledge management and learning processes**

Manages the Knowledge, both individual and collective, as well as learning processes that occur in the system (individual and collective).

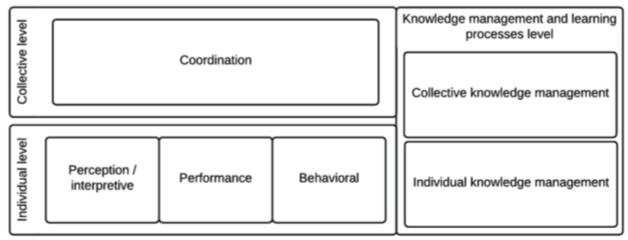


Figure1.Proposed Architecture

The following section explains in detail each one.

3.1 Collective Level

In [7, 8] is described the conceptual design of an adaptive architecture for cooperative multi-robot systems, which inhibits or activates its levels according to the expected behavior of the system, with planning and coordinating tasks as the core of its structure. In this paper (inspired in that work), is described an architecture that supports an emerging self-organized system composed of general purpose robots.

The essential level for this task is the Collective. This level comprises a coordination module. This module manages the dynamics of the interactions between the robots and of the robots with the environment, and the processes that arise from them. Next, we give an overview of how is the operation of the coordination module.

In this module, processes involved in the management of the interactions among individuals working in the system occur, such as inform their tasks, look for information they are interested, etc. Coordination in this module is manage under the emerging approach, such as a priori actions to be performed aren't established, but when decisions are made.

Under this approach, the division and allocation of tasks and negotiation emerge naturally in the system [9, 10]. Depending on the need of the community of agents in a given moment (collective goal), can be carried out information processes (direct or indirect), recruitment, search, among others. This module facilitates the occurrence of these processes.

3.2 Level of knowledge management and

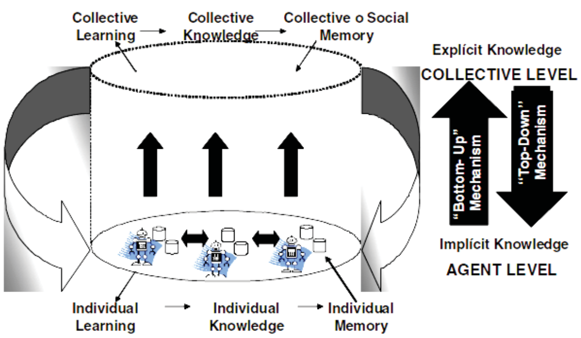
learning

The behavioral component of the robot governs the behavior of itself during its interactions with the environment and with other individuals; for which use their local knowledge as well as collective knowledge, which are acquired (learned) based on the performances of the robots in the system.

This level is responsible for the management of this knowledge, and facilitates the learning process. In Figure 2 is described the process of learning and knowledge management, based on the work presented in [11, 12], as is required by our self-organized emergent system.

Figure 2. Types of knowledge and learning in MASOES. [12]

According to this model, the robot increases their knowledge through a process of individual learning, and its interaction with other individuals and the environment. To do this, this level has a collective knowledge base and several locals, one for each robot. It also enables the processes of individual and collective learning, which feed off each other, through three phases:



- Socialization: mechanisms to share individual knowledge with the group.
- Aggregation: mechanisms to sort, filter and merge that knowledge.
- Appropriation: mechanisms for converting the collective knowledge in individual knowledge.

In particular, this level is structured in two layers:

- *Management Module of individual knowledge*: in this layer individual learning mechanisms are implemented, and the management of knowledge bases of each robot is performed. The mechanisms of socialization and appropriation are in this layer.

- *Management Module of collective knowledge*: This layer manages the collective knowledge base of the system, so that it is available to all individuals. Aggregation mechanism is in this layer.

3.3 Individual Level

In particular, the emergent behavior of insect societies and the appearance of swarms of robots have inspired a new class of robots, called swarm of robots [13].

A robot swarm has two parts, one being related to the hardware that shapes it and another related to the software, which controls its operation and behavior.

On the hardware side, for this research was designed a prototype robot swarm, inexpensive, with hardware architecture of four layers (see Figure 3):

Control: This component is responsible for the control process of the devices of the robot. It features an Arduino board, based on a micro-controller ATmega 328.

Locomotion / Performance: This component is responsible for managing the robot locomotion system, and the management of actuators that could be incorporated into its design. The robot has a system of differential locomotion idler, implemented through a gearbox, which transmits the power of two DC motors to the wheels.

Perception: It's responsible for managing the perception of the robot. The robot has infrared sensors to capture information from the ground, and a sound that allows it to capture information on distances to objects that are in front of it.It also has a virtual sensor that is provided by an artificial vision system, which allows it to capture information from the world, which isn't possible to grasp with their physical sensors.

Communication: This component does all the management communication process of the robot. The robot has a Xbee® device based on the IEEE 802.15.4 protocol, which allows communication point to point or multipoint with devices of the same class.

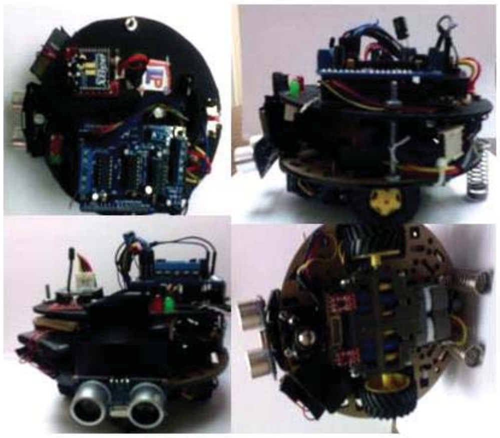


Figure 3.Robot of general purpose

The robot also has a software architecture that allows it to manage its tasks. This local architecture consists of a set of libraries that allow it to control the sensors and actuators of the robot, as well as communication between the robot and other devices. It is organized into modules:

3.3.1 Module of perception/interpretation

This module is responsible for obtaining the information from the sensors of the robot, both physical and virtual. These data are pre-processed and can cause activation of reactive or deliberate behaviors, as appropriate, the latter are managed by a sub-module of interpretation, which receives sensory information from the robot, it interprets it, and activates deliberative behaviors of the robot. These can lead to planning and/or cooperation needs, managed by the collective level of architecture. In other words, external or internal stimuli are processed at a t time which is received by the robot, and lead to a change in its internal state or its environment at a t+1 time. This module basically gives the robot capabilities of perceiving information from the environment, and interpreting it to generate appropriate responses to stimulus received.

3.3.2 Executing Module

It manages the operation of the actuators of the robot. Particularly, manages the locomotion system of the robot, in order to allow its movement in the environment as well as it controls any other actuator that could be implemented in it. This level is implemented locally on the robot.

3.3.3 Behavioral Module

The behavioral part of the robot is more complex. In [11] is proposed an architecture for non-formal modeling of emergent self-organized systems (MASOES), which includes individual and collective aspects involved in this type of system.

For individual aspects, it describes a series of internal components of each agent, which through their interactions generate the behavior itself, which are activated by other modules of the architecture. These are:

Behavioral component: it is responsible for regulating the conscious and emotional behavior of the agent.

Reactive component: it is responsible for generating the reactive behavior of the agent.

Cognitive component: it generates cognitive behavior through reasoning mechanisms.

Social component: it promotes aware in the agents about what made the other agents. Here collective knowledge is used.

Based on this model is proposed the behavioral module for the robot, which takes the individual components proposed in MASOES, and are modified in order to suit them to the architecture proposed in this paper. Figure 4 shows the proposed design for behavior on the robot module:

Reactive Layer: This component interacts directly with the components of perception and performance of the hardware architecture of the robot, and it is responsible for generating reactive behaviors in the robot. It is understood by reactive behaviors, actions that execute the robot directly from a stimulus received, without first going through a deliberative process, are actions resulting from a process of stimulus - reaction. This layer manages survival behaviors of the individual, and has priority over any other behavior that can be generated.

Cognitive Layer: This layer generates deliberative behaviors in the robot, based on its local knowledge. We define deliberative behaviors those actions that occur after the stimulus received by the robot has been processed and interpreted; in order to generate an analyzed action (there are implicit reasoning processes). Here it generates complex behaviors, built from primitive behaviors, which allows running specific and more complex actions.

Social layer: It explodes the collective knowledge in decision-making processes of the robot. It interacts closely with the other two levels (collective and knowledge management), for the planning, coordination and/or cooperation processes in the system to generate behaviors that allow the robot to interact with others within the system, either directly or indirectly. Basically, manages how the robot interacts with other individuals of the group.

Affective layer: based on [12] is proposed an affective model, which considers a set of positive or negative emotions involved in generating behavior of robots, which affects the level of self-organization and emergence of the system. These emotions directly affect the individual and collective behavior of the robots. In the behavioral model proposed in this paper, the affective component is transverse to the cognitive and social components, and consists of a set of basic emotions that affect the behavior of the robot in the environment and its willingness to the execution of the tasks to be executed. Mainly, the affective component inhibits or activates the behaviors generated by the cognitive and social layers.

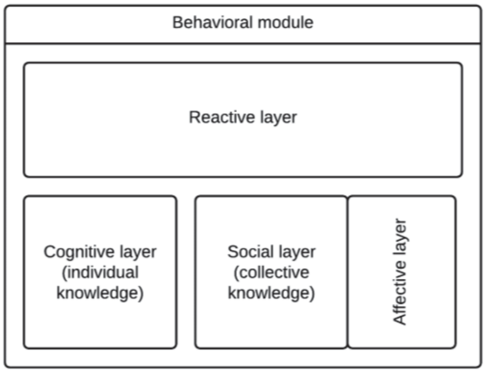


Figure 4.Behavioral model proposed

In general, the behavioral component provides a set of behaviors, which allow the operation of the robot and the emergence of collective behaviors in the system.

At the implementation level, almost everything is made in a shared computing cloud between robots, which allow them to perform highly complex processes. What is implemented in the robot are basically: libraries to control actuators, sensors, and the primitive behaviors, and management of the communication device.

4 Case of study

In this section is described the performance of our architecture for transporting objects.

4.1 Transport of an object

A classic case of study in multi-robot systems is the transport of objects. In [14, 15, 16] some related works are presented, where different solutions to the problem are shown. For example, a solution poses a leader in the system that centralizes planning task; another raised by the use of bio-inspired algorithms in solving the problem, as in the proposal presented in [17], the last taking as a reference the behavior of ant societies to establish a set of primitive behaviors for each agent, where a simple coordination mechanism between them is achieved.

However, a solution based in emerging systems consists of:

- The group of individuals explores the environment in pursuit of its goal,
- When one finds the object, it tries to move it and take it to the place prepared for deposit,
- If it can't do it, it recruits others,
- Then, a cooperative mechanism is activated in order to organize the group of individuals to move the object collectively.

This solution is inspired by the behavior of ant colonies, and the key is a pattern of formation around the object that allows its transport. The operation of our architecture is described in this last case of transport of an object; in particular, it shows how the architecture provides support to the emergent behavior displayed during the process.

4.2 Our architecture supporting the transport task

Following is described step by step, in general, the operation of the architecture, for the case of transporting an object seen as an emergent process:

1) System configuration: three robots, an object, and a point of deposit (where leave the object), are located at random positions.
2) Initialization of the emotional state of each robot: in the affective layer of the behavioral module of each robot, an *emotional* state and *an activation threshold* is randomly defined. This threshold and state will influence its disposition towards the execution of tasks in the environment, and its relationship with other individuals.
3) *Activation of primitive behaviors*: Once the robots are in the environment a primitive behavior, from the set of robot's behaviors, is activated in the reactive layer of behavior module. In this particular case is *search*, which focuses on moving through the environment without a predefined plan, searching for the object.
4) Detection of an element in the environment: When it detects an element in the front (it is based on the perception module, in its virtual sensor), it parses it through the cognitive layer, to determine one of the following situations:

- *Detects an obstacle:* it activates an evasion behavior thereof through the cognitive layer. The robot analyzes (reasons) a number of actions required to avoid the obstacle and move on. The Individual knowledge base is updated and the action is shared collectively, to be used by other individuals.
- *Detects a robot*: the reactive layer of the behavioral module activates a primitive behavior: collision avoidance. The robot immediately stops its movement to a safe distance.
- *Detects pheromone:* it activates the coordination layer to allow them to manage these interactions. The robot recognizes the type of stimulus, its intensity, etc., and on that basis it takes the decision to act (follows the stimulus) or not, that decision is also affected by its emotional state. The robot updates its individual knowledge base, and if action is carried out, the collective knowledge base (pheromone) is updated.
- *Detects the object to be transported (to push):* after the object is detected, the reactive layer is responsible for

generating a behavior through which the robot tries to push the object. In the first instance it will try to do it individually, depending on its emotional state. If the result is negative and fails, the cognitive layer is activated for the robot to try other alternatives to move it, for example, reposition. If the result remains negative, the robot will seek to recruit others, using the behavioral module (to know its environment), the layer of emergent coordination, and the knowledge management level (for example, to leave traces in the environment through the virtual sensor, or to send messages of help (broadcast)). In this way, the robot tries to attract other individuals to the object in order to generate a collective action. In all cases, individual knowledge is updated through knowledge management level, and in the case of seeking help, the collective knowledge.

5) Several robots around the object: in this case, more than one robot tries to transport the object, either because they got almost simultaneously or they were recruited. In this case the collective level is activated, as well as the social layer of behavioral module, in order to coordinate the actions of the robots around of the object. They solve the situations of conflict as the direction of the movement, positioning of the robots around of the object, and the number of individuals needed to execute the task. To solve it, it activates repositioning mechanisms around the object, negotiating to reach a common direction, and adjustment in the number of individuals. That is, the transport task is coordinated, emerging *an organizational pattern* that allows transport the objet.

6) End of task: the robot or the robots which are carrying out the transport detect the destination of the object, such that it is deposited. The level of coordination is inhibited if there aren't interactions at that moment, and the individual level activates its behavioral component, in order to activate new individual behaviors. All databases of knowledge are updated (local and collective) as the result of the task (the learning mechanism is activated).

5 Final Considerations

The proposed architecture provides a framework to facilitate the emergence in multi-robot systems, through its three layers. They manage the processes at individual and collective levels that occur in the system, as well as knowledge that is built during the performance of robots.

The architecture allows the individual performance of each robot, allows local decision-making of each, and supports the interactions of the robot with the environment and with other robots. Its modular structure facilitates the addition of new components, and the update of the mechanisms used. The behavioral component of the robots allows different behaviors and intentions towards the execution of a task, or the interaction with other robots, enabling the study of their behaviors at the collective level, and how this affects the emergence in the system. This component will be based on a range of *emotions*, represented by indices of satisfaction or rejection to carry out a task.

The architecture allows the management of key aspects of emerging systems, such as the execution of the processes of the swarm in a distributed way (for robots), the local decision-making for each robot, facilitates the interactions between robots (through the use of shared memory spaces or messages), and allows the construction of individual and collective knowledge.

To show the behavior of the architecture, we have used the case of study of transportation of an object in an emerging form. In this particular case, the architecture is capable of supporting the emergence of the organizational pattern of the swarm for transport. To do this, it enables collective typical tasks of this emerging process, such as determining the number of individuals needed to perform the task, recruit other robots when one can´t move the object, etc. It also allows the resolution of conflicts that arise at the time of the movement of the object, or the conflicts of the interaction of multiple individuals in the same space.

The architecture is scalable in the number of robots, this is because each individual acts independently in the environment, and collective processes appear by the own dynamics of the system and not a priori defined configurations. Whenever a robot is included in the system, it is instantiated in the architecture individually. The architecture also allows the inclusion of robots with different hardware (there are not assumptions about it). The system is robust and flexible, since the failure, replacement or addition of a robot, does not affect the overall structure of the system.

Future works will be devoted to the specific problem of implementing of each level, and analyze the processes of emergence when there are specialized individuals (robots). Also, we will study the emotions in the system, to include behavioral and emotional aspects of greater complexity.

Acknowledgments

Dr Aguilar has been partially supported by the Prometeo Project of the Ministry of Higher Education, Science, Technology and Innovation of the Republic of Ecuador.

6 References

[1] Roberto Polesel, Robert Rosati, Alberto Speranzon, Carlo Ferrari and Enrico Pagello. "Usign Collision Avoidance Algorithms for Designing Multi-robot Emergent Behaviors"; In Proceedings of International Conference on Intelligent Robot and Systems, pages 1403-1409, 2000.

[2] Mininath K. Nighot, Varsha H. Patil and G.S. Mani. "Multi-robot hunting based on swarm intelligence"; 12th

International Conference on Hybrid Intelligent Sytems, 2012, pp. 203-206.

[3] Jong-Hyun Lee, Chang Wook Ahn and Jinung An. “A Honey Bee Swarm-Inspired Cooperation Algorithm for Foraging Swarm Robots: An Empirical Analysis”; Proceedings of International Conference on Advance Intelligent Mechatronics, pages 489-493, July 2013.

[4] Tom Anderson and Max Donath. “Autonomous Robots and Emergente Behavior: A Set of Primitive Behaviors for Mobile Robot Control”; Proceedings of the International Workshop on Intelligent Robots and Systems IROS, pages 723-730 1990. IEEE.

[5] Zhengping Li, Cheng Sim and Malcolm Low. “A Survey of Emergent Behavior and Its Impacts in Agent-based Systems”; Proceedings of IEEE International Conference on Industrial Informatics, pages 1295-1300, August 2006.

[6] Sean Verret. “Current State of the Art in Multi-robot Systems. Defense Research and Development Canada, Technical Memorandum”; DRDC Suffield TM 2005-241, Canada. 2005.

[7] Angel Gil, Jose Aguilar, Rafael Rivas and Eladio Dapena. “Arquitectura Adaptativa de Cooperación para Múltiples Robots: Primera aproximación al diseño conceptual”; Proceedings of First National Conference of Computing, IT and Systems (CoNCISA), pages 154-161, October 2013.

[8] Angel Gil, Jose Aguilar, Rafael Rivas and Eladio Dapena. “Implementación de Mecanismos de Cooperación en una Arquitectura Adaptativa para Sistemas Multi-robots”; Proceedings of XII International Conference on Numerical Methods in Engineering and Applied Sciences, pages EC 7 - EC 12, April, 2014.

[9] Jose Aguilar, Addison Rios, Francisco Hidrobo and Mariela Cerrada. “Sistemas Multiagentes y sus Aplicaciones en Automatización Industrial (2da Ed.)”. Talleres Gráficos, Universidad de los Andes, 2012.

[10] Jose Aguilar. “Introducción a los Sistemas Emergentes”. Talleres Gráficos, Universidad de los Andes, 2014.

[11] Niriaska Perozo, Jose Aguilar Oswaldo Terán y Helen Molina. “A Verification Method for MASOES”; Transactions on Systems, Man, and Cybernetics, 43(1): 64-76, 2013.

[12] Niriaska Perozo, Jose Aguilar Oswaldo Terán y Helen Molina. “An affective model fot the multiagent architecture for self-organizing and emergent systems (MASOES)”; Revista Técnica Ingeniería Universidad del Zulia, 35(1):80-90, 2012.

[13] Vinay Pilania Shaunak Mishra, Subrat Panda, and Aurosish Mishra. “A Novel Approach to Swarm Bot Architecture”; Proceedings of IEEE International Asia Conference on Informatics in Control, Automation and Robotics, pages 418-422, February 2009.

[14] Yoshitaka Tohyama and Hiroshi Igarashi. “Cooperative Transportation by Multi-robots with Selecting Leader”; Proceedings of 35th Annual Conference of IEEE Industrial Electronics, pages 4179-4184, November 2009.

[15] Sungwon Moon, Dongjun Kwak and How Kim. “Cooperative control of differential wheeled mobile robots for box pushing problem”; Proceedings IEEE 12th International Conference on Control, Automation and Systems, Korea, pages 140-144, October 2012.

[16] Arnab Ghosh, Avishek Ghosh, Amit Konar and Ramadoss Janarthanan. “Multi-robot Cooperative Box-pushing Problem Using Multi-objetive Particle Swarm Optimization Technique”; Proceedings of World Congress on Information and Communication Technologies (WICT), 2012.

[17] C. Ronald Kube and Eric Bonabeau. “Cooperative transport by ants and robots”; Robotics and Autonomous Systems, Elsevier, 2000.

Singer Module with Singing Synthesis for Humanoid Robots Applications

A. Rojo-Hernandez[1], H.M. Perez-Meana[1], N. Takayuki[2], E. Hernandez[1], and J. C. Sanchez[1]
[1]SEPI, National Polytechnical Institute, Mexico D.F., Mexico
[2]DMEIS, The University of Electro-Communications, Tokyo, Japan

Abstract— *Currently the research on robotics is a field with the most demand in various parts of the world, as it seeks to ensure they are capable of developing many of the most common tasks humans perform, such as identifying objects and taking them to recognize people by voice or face, identifying the words a person is saying, moving in free space, talking (to interact with humans) and one of the last things to be achieved: Making a singing robot. For this reason, on this task we will talk about a method in which a robot can sing using Vocaloid Editor 3.0, a speech synthesizer software which is able to generate the songs a robot will sing.*

Keywords: Speech synthesis; Phoneme; TTS (Text-to-speech); Acapella; Vocaloid Editor 3.0; VocaListener.

1. Introduction

A synthesizer is a tool designed to produce electronic musical sounds generated artificially, using techniques such as additive synthesis, subtractive, frequency modulation, or physical modeling of phase modulation, to create sounds. The synthesizer sounds are created by direct manipulation of electrical signals, through the manipulation of a digital FM waves, handling of discrete values using computers (software based synthesizers), or combining any method. In the final phase of the synthesizer, electric currents are used to produce vibrations in speakers and headphones.

The steps followed in all synthesis process are: First, a set of modules analyzes the input text to determine the structure of the sentence and the phonetic composition of each word and a secondly, another set of modules transforms this abstract linguistic representation into speech [9].

Currently there are many techniques and algorithms for synthesizing text into speech. These algorithms are called Text-To-Speech (TTS). An example of this type of algorithm can be seen in the program developed by Yamaha Corporation: Vocaloid Editor 3.0.

Given the case that Vocaloid has been developed by the Japanese company Yamaha is expected that the program is based on the same language in others words in Japanese. Therefore this research will use the Japanese language as basis for the songs. It is important to emphasize that Vocaloid Editor 3.0 will be responsible for synthesizing the song, therefore it will be syncing the timing of the notes with the phonemes and taking care of the duration of them.

2. Objective

The main objective of this paper is to propose a way or method by which it is possible to create a new module which will make the robot sing. Since the songs are synthesized using Vocaloid Editor 3.0, the aim of this project is to find a way in which the robot can play the songs that have been previously synthesized and are stored in a database, by a new module programmed in C++ that is responsible for the reproduction of these songs and can be added to existing modules of the robot.

3. Robot Plataform

To understand what is planned, it is necessary to first understand how a robot works, and how hardware and software combine themselves to correct operations, providing a better control of it.

We can see a robot like a system which is in turn divided into multiple modules that control the different parts or components of the robot. Modular organization systems provide greater control of the robot and make the error corrections.

The robot has modules that control vision (camera), audio (speakers and microphone), hands and arms, wheels among other modules. Each module has been programmed in C++ on Visual Studio environment. This work focuses on creating a new module that will be charged of making the robot sing, or in our case, play songs that have been synthesized using Vocaloid Editor 3.0. The whole system is divided in four modules: Vision, Audio, Robot and Task Modules and a server.

All modules are connected through the "Server" with GigE and have subscription information that describes required information for the processing each module. All information is gathered in the server and then the server forwards information to each module according to its subscription information.

The "Task Module" works as a controller for each scenario. This modular network architecture makes it relatively easy to share the job in the development stage of the robot system [1]. The robot platform used in this paper is shown in Figure 1. The robot is based on the Segway RMP200 and consists of the following hardware components:

- The robot is equipped with a visual sensor.
- Two Arms capable of six degrees of freedom (6DOF robotic arm manufactured by Exact Dynamics), and 6-DOF hands.
- Omnidirectional wheels and a laser range finder (LRF) enable the robot to move freely within a room. Laser range finder (HOKYO UTM-30LX) is used for environmental mapping.
- Four on board PCs (Intel Core2Duo processor) are communicated each other through LAN.
- A sanken shotgun microphone CS-3e for audio input and Yamaha speaker NX-U10 for audio output.
- A stereo camera is used for obtaining depth information.
- The camera and microphone are mounted on Directed Perception pan-tilt unit PTU 46-47. [1] [2]

The abilities of the robot, other than the learning novel objects, are listed below [2]

1) Online SLAM(Simultaneous Localization and Mapping) and path planning.
2) Object manipulation (RRT-based path planning).
3) Simple speech interaction in English and Japanese.
4) Human detection and tracking using visual information.
5) Searching objects in the living room environments.
6) Face recognition using 2D-HMM.
7) Gesture Recognition. [1]

The new task to be performed is making the robot sing.

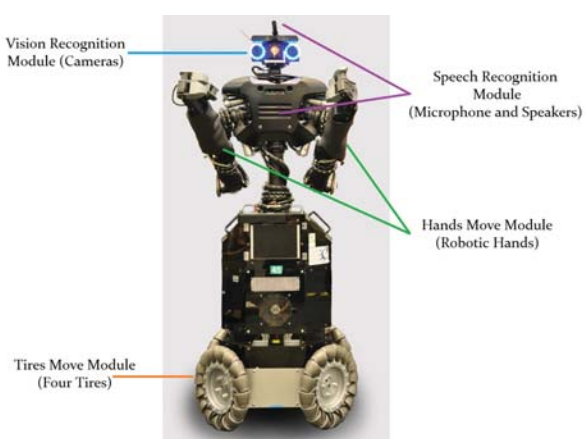


Fig. 1: Robot's Parts 大五郎 (DIGORO).

4. System Singer

The Singer Module is the new module that has been programmed and, as the name mentions, it will be in charge of the task of singing. The Singer Module needs to use the Audio Module; the Audio Module is divided into two parts: The Speech Recognition Nodule and the Jukebox Module.

The Speech Recognition Module works with everything related with speech process. The Jukebox Module is responsible for making the reproduction of the songs that will be performed by the robot. We will talk more about these modules later. The proposed path for making the robot to be able to sing is as follows.

The user asks the robot to sing any song using the Speech Recognition Module. The robot detects and sends the command to the Singer Module, this is the one that is responsible for processing and executing the command. Finally the robot starts to play the song that has been requested. In Figure 2 a block diagram of the system is presented. The process is performed as follows.

According to the Figure 2 this process was made in this way because the robot doesn't have ears, vocal cords, or other Senses and the way in which the robot is able to perform this task in a way is similar to a human is because the robot replaces ears with speakers and the vocal cords with the microphone.

The first time we tried to do a synthesis of the songs in real time, Vocaloid Editor 3.0 required a complex process for speech synthesis and it was not possible do it in real time, therefore the songs that the robot will reproduce have been previously synthesized using Vocaloid Editor 3.0 (For more references go to section 7, 7.1 and 7.2). In subsequent sections, each of the steps involved in all the modules will be explained in detail.

Fig. 2: General block diagram of proposed system.

5. Speech Recognition Module

The Speech Recognition Module used by the robot is a software system called Julius. It is responsible for speech processing and speech recognition. Within the recognition Julius makes use of the internet for search of vocabulary.

The Speech Recognition Module detects the words that are being said and with the help of the internet, it provides ten possible options of what was said.

5.1 Julius

Julius is a high-performance, two pass large vocabulary continuous speech recognition (LVCSR) decoder software for speech-related researchers and developers. It can perform almost real-time decoding on most current PCs in 60k word dictation tasks using word 3-gram and context-dependent on Hidden Markov Model (HMM). Major search techniques are fully incorporated. It is also modularized carefully to be independent from model structures. Various HMM types are supported, such as shared-state triphones and tied-mixture models, with any number of mixtures, states, or phones. Standard formats are adopted to cope with other free modeling toolkits.

Julius works with HTK which has the control of HMM. In order to execute the Julius recognizer, you need a language model and an acoustic model for your language. Julius adopts acoustic models in HTK ASCII format, pronunciation dictionary in HTK-like format, and word 3-gram language models in ARPA standard format (forward 2-gram and reverse 3-gram as trained from speech corpus with reversed word order). Its important to mention that Julius is only distributed with Japanese models. [15]

5.2 Hidden Markov Model Toolkit (HTK)

The Hidden Markov Model Toolkit (HTK) is a portable toolkit for building and manipulating hidden Markov models. HTK is primarily used for speech recognition research; although it has been used for numerous other applications include research on speech synthesis, character recognition and DNA sequencing.

HTK consists of a set of library modules and tools available in C language source form. The tools provide sophisticated facilities for speech analysis, HMM training, testing and results analysis. The software supports HMMs using both continuous density mixture Gaussians and discrete distributions and can be used to build complex HMM systems. [8]

5.3 Singer Module

The Singer Module is responsible for processing the command received from the Speech Recognition Module.

Figure 3 shows a block diagram of the edge module. We can see that the first step is to receive the command of the Speech Recognition Module for later processing.

The first thing that is done in the processing is to find the proper command because the Speech Recognition Module delivers ten possible options, so the first step is to analyze each of these options and identify if any of them corresponds to an command known.

Once the command has been identified the process continue to process it. In the processing section it is necessary to separate the command into two parts, by name of the song and command.

When the program has separated the name of the song and the command to run, the next step is to analyze the command. It is important to note that the commands will be delivered to the robot in Japanese as well as the names of the songs.

In order to identify the command it is necessary to search which command has been given. In this case there are two different commands with diferent Japanese representations, but in this case, we just to use two representations: hiragana and kanji (Katakana just on case of words in english).

1) Play the song (song's name をうたってください, を歌ってください song's name o uttate kudasai)
2) Stop the song (やめてください, 止めてください yamete kudasai)

After having identified the command, the module proceeds to search the name of the song in a database, if found, the command is executed: e.g. If the given command was "ああをうたってください" ("aaa o utatte kudasai) will be sought in the the song database "ああ" ("aaa") and as the command has already been previously identified "うたってください" ("utatte kudasai") in this case, it will play the song; then finally, it proceed to play the song that has already been found. The graphical process of this module is indicated in Figure 3.

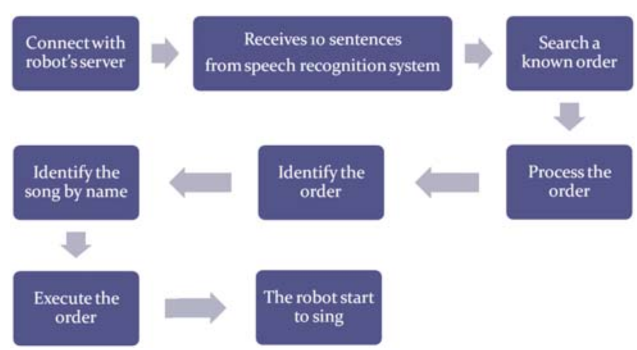


Fig. 3: The Singer Module process block diagram.

It's important to mention the Japanese's representation of the commands because there are differently presentations and therefore it's more difficult to identify a command both of this representations are given by the voice recognition within the ten options that it produces, ie within ten options it generates could be possible that two or more sentences have the same meaning but with different representation, however the singer module take into account just the first correct representation that you find.

The identification process commands and song's title. Because we only take into account two possible representations for commands, in total we have four possible correct commands, two for play and two for the stop.

- Command 1: をうたってください (uttate kudasai “play”).
- Command 2: を歌ってください (uttate kudasai “play”).
- Command 3: やめてください (yamete kudasai “stop”).
- Command 4: 止めてください (yamete kudasai “stop”).

In the flowchart of Figure 4 you can see the process undertaken to identify commands and songs, we can see that through a string comparison, it performs a search for known commands as well as the search of the database of song title. In other words, when the ten options are received by the Singer Module, it takes the first option and compares it one of the four possible commands. If the comparison is negative, then it takes the second option and a comparison process is repeated again. The process is repeated until the comparison is positive in any of the ten choices, or until all have been compared with the ten options and none is positive, meaning that the user is required to repeat the command and the process start again.

In the event that one of the comparisons is positive, the next step is to clean the sentence eliminating unnecessary information such as the name of the robot. When you already have the free judgment garbage information, then it proceeds to compare the title song with song titles available in the database. This procedure is performed on the basis of comparisons similar to the identification of the command. So if it was compared to all existing titles in the database and comparison have been negative, then it is necessary that the user repeats the command and to restart process. However, in the event that it finds the title of the song in the database, it proceeds to extract the song ID and the ID of the command, and these two numbers will be sent to Jukebox which is the module responsible for playing or pausing songs.

As the goal was to create a module that was able to control playback of songs acording the information received by speech recognition, we don’t implemented a more elaborate algorithm for identifying commands and songs. Another reason for not implementing a more efficient searching algorithm is because the database that we created is small and it would be a computational process that we don’t really need.

6. Jukebox Module

As mentioned previously Jukebox Module is an audio processing manager. In this case, it represents the songs which have been previously synthesized with Vocaloid Editor 3.0.

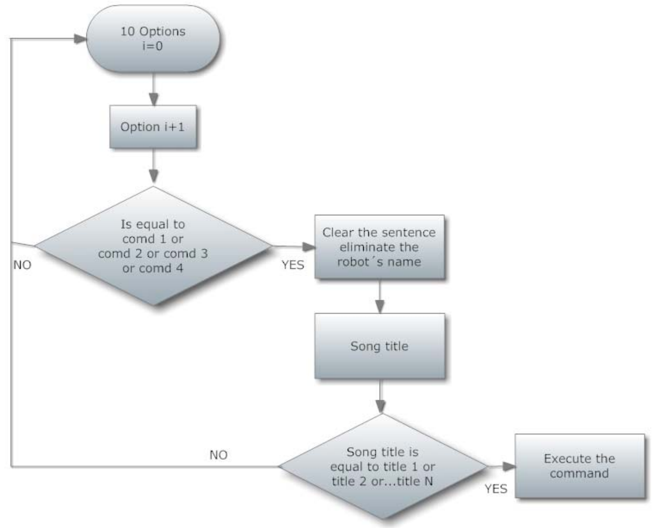


Fig. 4: Command and song title identify flowchart.

Jukebox is able to play files in wav, mp3, wmv, wma and mpg. Basically it has two functions: playing and stopping audio files in the formats mentioned above. These two functions or commands use one ID number to facilitate the execution of commands. In turn, each audio file has an ID number that is assigned in the command in which they were added to the database.

Jukebox receives the command’s ID and the song title ID after the song begins playing or is stopped according to what prompted. Virtually, Jukebox plays or stops a song when the command has been executed.

7. Vocaloid Editor 3.0

Vocaloid (ボーカロイド Bōkaroido) Editor 3.0 is an speech application software that is able to sing. It was developed by the Yamaha Corporation in collaboration with the Music Technology Group at the University Pompeu Fabra in Barcelona, Spain.

The software provides the user with the ability to synthesize songs simply by typing the lyrics and melody. It uses the voice synthesize technology specially recorded from dubbing actors or singers. To create a song, the user must enter the melody and lyrics. An interface of a piano roll is used to incorporate the melody and lyrics that can be put into each note [10].

Figure 5 shows the main screen of Vocaloid Editor 3.0. To the left, you can see the piano roll that helps create notes and some notes being made example.

7.1 VocaListener

VocaListener is a plug-in that can be integrated to the Vocaloid Editor 3.0 program. VocaListener focuses on the combination with the Vocaloid Editor 3.0 software

Fig. 5: The main screen of Vocaloid Editor 3.0

and facilitates the work and the burden of making a song with Vocaloid Editor 3.0, and at the same time, improves the quality of the song. VocaListener is an automated online-generated file system and is running on a specific server, allowing users quickly get a good quality of CSA (Vocaloid Sequence) in a quick way, to present it before loading the original file and the voice wave lyrics file. This technology can greatly reduce the work to make a good Vocaloid VSQ file [3].

In Figure 6 we can see at the top an example of how to introduce the song (Acapella Song) and VocaListener generate the voice signal that is showed in the image, therefore the notes are created for this input signal. We can also see an example of the notes as they are created and how you entered the letters of the same manner that the letters should be introduced in Japanese using hiragana.

7.2 Synthesizing a song using VocaListener

There are several ways to synthesizing a song using Vocaloid Editor 3.0 but the way you have best results is when using VocaListener.

To synthesize a song using VocaListener, it is necessary to have the song's acapella version in wav format. Then the file is exported to VocaListener and this generates the signal corresponding to the voice, this signal is what helps us generate the notes on the size and scale necessary to go manually entering only the lyric of the song, remembering that we have to respect the hiragana letters and we must introduced the lyric in Japanese.

Since it has been introduced the letter and notes were generated need to listen and edit manually synthesizing as VocaListener not a perfect program that can generate so silent moments that cause the sequence of notes lose

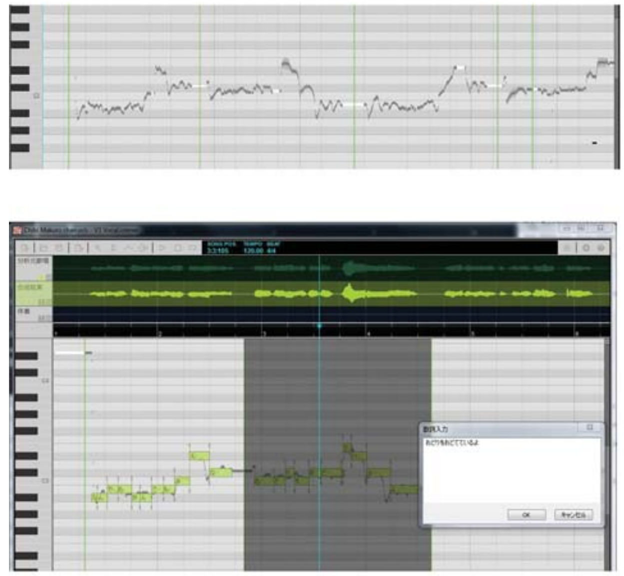


Fig. 6: VocaListener's Main Screen.

quality.

When you finish editing the synthesis according to our requirements is exported to Vocaloid Editor 3.0 the sequence of notes which is what allows us to manipulate the voice of the interpreter, i.e. the interpreter we can choose to be a male or female voice. Finally this synthesized song is exported database where Jukebox can take it to be reproduce.

8. Experimental Results

Being able to sing as part as a module adds an additional function to the robot, which in this case as the name says is a feature that allows the robot to sing. This function as seen above is activated by voice no matter who is the speaker, but the commands are specific, i.e. for the module is activated you need to ask the robot to sing a song, we have to remember that the robot base language is Japanese.

The module makes singing the prayers processing provided by speech recognition that facilitates the rapid identification of errors. We have different types of possible errors are detected. Then we'll talk a little about them.

- Error 1: The Singer Module first searches the command is known, therefore if within ten options provided by speech recognition known no command module discards this information and start the process again so the user will have to repeat the command.

Table 1: Commands executed.

<table>
<tr><th></th><th>Commands Numbers</th><th>Times Repeat the Command</th><th>Type Error</th><th>Type Command</th></tr>
<tr><td>Execute</td><td>91</td><td>1/1</td><td>0</td><td>0</td></tr>
<tr><td rowspan="4">Error</td><td rowspan="4">9</td><td>2 (3 times repeat, first and second time causes an error on third time the command is executed)</td><td>4 (song)</td><td>4 (play)</td></tr>
<tr><td rowspan="3">7 (2 times repeat, first time cause error second time the command is executed)</td><td rowspan="2">4(command)</td><td>2 (stop)</td></tr>
<tr><td>2(play)</td></tr>
<tr><td>3 (song)</td><td>3(play)</td></tr>
</table>

- Error 2: The following error has to do with the song and here are two possibilities:
  - Error 2.1:
    Taking into account that the command has been detected as known in this case we proceed to find the song but if the song is not in the database the Singer Module will detect this error and then this process will reboot and the user will have to repeat the by the robot.
  - Error 2.2 If the song is in the database but speech recognition is not optimal, i.e. has the wrong name, then reboot process and the user will have to repeat the command to the robot. These errors have to do with the command to sing (play), in case the command you ask the robot is the stopping (stop) then there can only be one type of error is that the command is not identified as known and the user will have to repeat it.

To test the operation of the Singer Module became the next test. He gave the commands system randomly hoping that properly executed which run ninety-one times correctly, i.e. that it was only necessary once for both Speech Recognition Module singing as operate properly. These results we can observe in the table 1.

The remaining nine times an error was generated from those already mentioned above. In this case two of those nine times that there was error was necessary to repeat the command three times, i.e. the first and the second time the command was given and error was detected until the third time that the command was properly recognized and enforced, for this case, the error occurred in the command “play” and due to a mistake in the name of the song because the name was not detected correctly.

On the other hand seven times was necessary to repeat the command twice, i.e. the first time you said the command was a mistake and was present until the second time the command was said that this was identified and executed properly. Four times the error is present because the command was not correctly identified by speech recognition. It is noteworthy that were twice as was the command “play” and twice the command “stop” respectively which were not identified correctly.

Finally three times was no error in the command “play” because the song is not found as the name of the song provided by the speech recognition did not match that of the database.

From this we conclude that on average 90.09% commands are executed correctly and also on average it can fail 9.90% of the time, that the best you only need to repeat the command a second time for this to run properly.

9. Conclusions

Using the Singer Module we can obtain a new function that the robot will be able to perfom perform in this case is to sing a song when prompted to do so.

The Singer Module as we saw earlier in the process needs the help of Speech Recognition Module and Jukebox Module, it is important to mention that for the Speech Recognition Module its really complex identify long sentences as is the command they need to activate the Singer Module, so one of the qualities of Singer Module is processing information that provides speech recognition and extract the important parts of the sentences that receives and analyze, identify errors if even present and otherwise execute the command you have received.

As mentioned in the results section the 90% of the commands that are given to the system run successfully, these are good results since we can say that the processing module performs Singer is optimal and efficient.

We note that speech recognition improves if the person giving the commands have a complete mastery of the language in this case Japanese because as mentioned in section Julius Speech Recognition Module is a program designed especially for the recognition of Japanese. Furthermore Vocaloid Editor 3.0 was chosen as base software for creating songs as this is a software dedicated to speech synthesis based on letters and notes here.

For the creation o the songs, those were recorder in ”acapella” versions in order to use the tool VocaListener and generate more quality songs. Vocaloid Editor 3.0 is one of the best programs for creating synthesized speech, artificial voice in other words, so far has not been possible for humans to create a perfect speech cast by a robot or a machine, Vocaloid Editor 3.0 is as close to a human voice as the software allows modification of some features of voice such as vibrato, pitch, stress of the pronunciations, changing dynamics and tone of voice.

As future work is to further expand the database of songs and create songs in other languages such as English or Spanish. In addition to adding a new process to Singer Module that is dance, i.e. the robot generates some movements either your head or hands, to be in line with the song that played. If the database can be expanded for future work would be to implement a more efficient search algorithm to improve the processing of commands and execute commands faster.

Acknowledgments

This work was completed in Nagai's Laboratory of The University of Electro-Communications. For this reason I want to thank all members of Nagai's Laboratory and specially my professor Nagai who kindly have help me during the last year.

Thanks so much to JUSST program, JUSST program Staff, and JASSO scholarship for supporting this research.

Also, special thanks to my advisor Enrique Escamilla Hernandez from National Polytechnic Institute of Mexico because he gave me the oportunity to come to Japan and do this work.

Finally thanks to Rizki Satya Utami and Fukui Miyuki by giving me permission to record their voices.

References

[1] M. Attamimi, A. Mizutani, T. Nakamura, K. Sugiura, T. Nagai, N. Iwahashi, H. Okada and T. Omori, "Learning Novel Objects Using Out-of-Vocabulary Word Segmentation and Object Extraction for Home Assistant Robots," *in Proc. ICRA*, 2010, p.745-750.

[2] H.Okada, T. Omori, N. Iwahashi, K. Sugiura, T. Nagai, N. Watanabe, A. Mizutani, T. Nakamura and M. Attamimi, "Team eR@sers 2009 in the @Home League Team Description Paper," *in Proc. RoboCup International Symposium*, 2009.

[3] T. Nakano, M. Goto, "VocaListener: A singing-to-singing synthesis system based on iterative parameter estimation,"*in Proc. of SMC*, 2009, p. 343-348.

[4] M. Goto, "Active Music Listening Interfaces Based on Signal Processing," *in Proc. of ICASSP*, 2007, p. 1441-1444.

[5] M. Goto, "A Chorus-Section Detection Method for Musical Audio Signals and Its Application to a Music Listening Station," *IEEE Transactions on Audio, Speech and Language Processing*,2006, Vol. 14, No.5, pp. 1783-1794.

[6] M. Goto, "Music Listening in the Future: Augmented Music-Understanding Interfaces and Crowd Music Listening," *in Proc. of AES 42nd International Conf. on Semantic Audio*,2011, p. 21-30.

[7] M. Goto, T. Saitou, T. Nakano and F. Fujihara, "Singing Information Processing Based on Singing Voice Modeling," *in Proc. of ICASSP*, 2010, p. 5506-5509.

[8] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev and P. C. Woodland, *The Fundamentals of HTK*, in The HTK Book version 3.4, December 2006, ch. 1, pp. 1-13.

[9] T. Saitou, M. Goto, M. Unoki and M. Akagi, "Speech-to-singing synthesis: Converting speaking voices to singing voices by controlling acoustic features unique to singing voices," *in Proc. of WASPAA*,2007, p. 215-218.

[10] H. Kenmochi and H. Ohshita, "VOCALOID - commercial singing synthesizer based on sample concatenation," *in Proc. of ICASSP*,2007, p. 4011-4010.

[11] H.Kenmochi, Yamaha Corp., "Singing synthesis as a new musical instrument,"*in Proc. Of ICASSP*, 2012, p. 5385-5388.

[12] H. Fujihara, "Singer identification based on accompaniment sound reduction and reliable frame selection," *in Proc. of ISMIR*, 2005, p. 329-336.

[13] H. Fujihara and M. Goto, "A music information retrieval system based on singing voice timbre," *in Proc. of ISMIR*, 2007, p. 467-470.

[14] T. Nakano, "Voice Drummer: A music notation interface of drum sounds using voice percussion input," *in Proc. of UIST*, 2005, p. 49-50.

[15] (On line), [22 de February de 2013], Available online:http:// julius.sourceforge.jp //en__index.php

PathFinder: An autonomous mobile robot guided by Computer Vision

Andre R. de Geus[1,2], Marcelo H. Stoppa[1], Sergio F. da Silva[1,2]
[1]Modeling and Optimization Program, Federal University of Goias, Catalao, Goias, Brazil
[2]Biotechnology Institute, Federal University of Goias, Catalao, Goias, Brazil
Email: geus.andre@ufg.br, mhstoppa@gmail.com, sergio@ufg.br

Abstract— *Localization is a key research topic in mobile robotics, responsible to assist robotic systems equipped with sensors, to navigate with certain autonomy. Unfortunately, the sensors shows frequently reading errors that disturb its location. In this paper, we describe the development of a computer vision system for autonomous navigation of a robot in a simulated environment. The system uses a unattached camera to detect the robot, concentrating the localization problem in the feature extraction of the images. Also, it uses Artificial Intelligence algorithms to determine the path in order to find the best solution. Our results show that the robot was able to follow the path and reach the goal, validating the proposed method.*

Keywords: Autonomous Navigation, Path Planning, Artificial Intelligence, Computer Vision

1. Introduction

According to [10], mobile robots are automatic transport devices, indeed, mechanical platforms equipped with a locomotion system able to navigate through a given environment, endowed with a certain level of autonomy to their locomotion. Autonomy is not just about energy sufficiency issues, but also the processing capability to plan and execute some tasks. Navigation in unknown environments is one of the areas with great interest to mobile systems, offering wide range of applications, from systems that assist in-house cleaning [5] to dangerous operations of search and rescue [6].

Navigation is an intrinsic feature of robots, allowing them to move freely into its environment until reach its goal. According to [7], dead-reckoning is a classic method of navigation, whose accuracy depends directly on the quality of the sensors used. Given that its location is based in previously locations, is inevitable the accumulation of position errors.

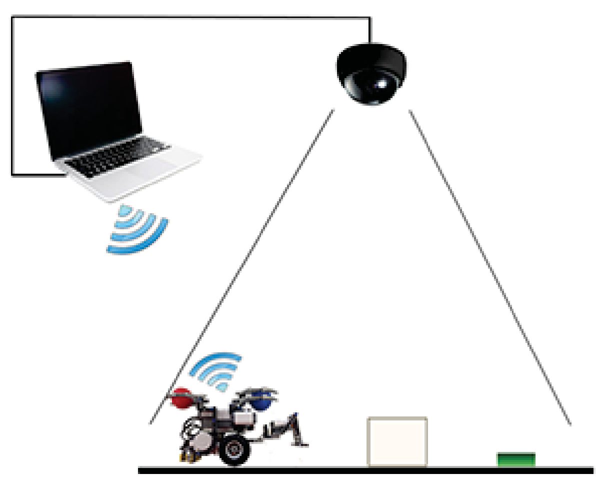
The compensation of position errors demands integration of multiple sensors. Combining their data, improves significantly the estimation of robots location in its environment. The technique proposed by [3] uses sensors that capture the direction, acceleration and engine revs. The authors demonstrate various improvements considering that heterogeneous sensors have different perceptions and can cooperate with each other.

Looking to increase the accuracy, adding information extracted from images provided by a webcam, shows to be an potential alternative to assist the path planning and execution. This is the motivation for the proposed work, that uses a webcam with panoramic view, to guide a robot to a goal, without human intervention and use of sensors.

2. Problem Description

The evaluated system consists of a robot with colourful parts that assist in determining its position, which is built under the Mindstorms NXT® platform. Big boxes are used to simulate obstacles that blocks the robots way. The goal is represented by a small sheet on the floor. A low-resolution webcam positioned on the environment's ceiling. The processing is performed by a laptop connected to the webcam, which sends the movement commands to the NXT®. Fig. 1 illustrates the environment layout and its components, where the target is a green rectangle, and the obstacle is a white rectangle.

Fig. 1: Side view PathFinder's environment layout



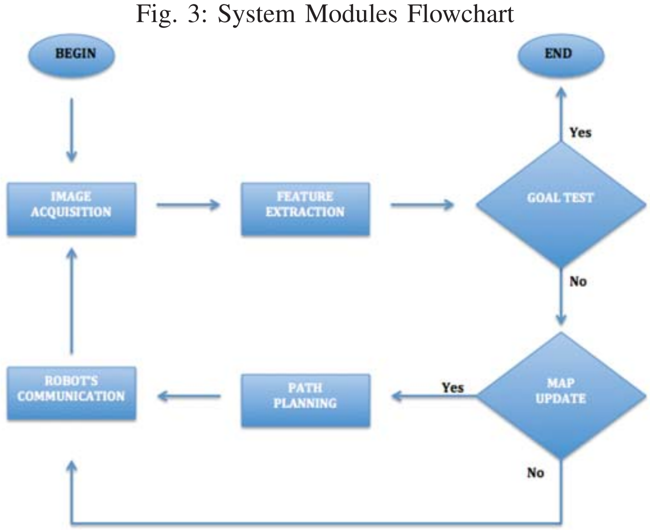
The environment layout has restrictions on the objects colours, due to the filters used to localize them. The robot has blue colour on its head and red in its tail (Fig. 2). The obstacles are white boxes of the NXT® kit and the goal is a small green sheet.

Fig. 2: PathFinder's design

The system uses Computer Vision [2], [8], [12] and Artificial Intelligence [13], [9] techniques. The system modules are illustrated in Fig. 3 and are responsible for:

1) Image acquisition using a webcam;
2) Feature extraction that determines the position of robot, obstacles and target;
3) Path planning to find the best solution avoiding obstacles;
4) Communication between the computer and robot;
5) Mapping update that also checks if the objects stay at the same place in the map;
6) Goal test to check if the robot reached the goal

Fig. 3: System Modules Flowchart



3. System Development

The autonomous control system is implemented in the programming language Python, which although slower than other languages has become very popular in a short time, due to its simplicity and clean syntax. Furthermore, it's easily integrated with C/C++ codes, allowing to involve them in Python wrappers. This provides two advantages: have a fast code as the original C/C++ code and easy syntax coding.

Two libraries are used: OpenCV (Open Source Computer Vision), coded in C and C++ with interface to Python, which has a wide range of Image Processing and Computer Vision techniques; and nxt-python, an interface designed to send commands interpreted by the NXT® brick, unlike NXC and NXT-G, compilable languages that runs on the brick.

The next subsections will describe the approaches for each module of the system shown in Fig. 3.

3.1 Image Aquisition

The system initializes sampling an image from the webcam through the OpenCV library, however this first image should be discarded because it's completely out of focus due to insufficient time to focus adjustment implemented in most webcams. To get around this problem, the system initializes the webcam and waits 30 seconds, enough time to adjust the focus.

The acquisition of image doesn't takes into consideration the quality of webcam resolution, so any low-cost webcam can be used in the system. It happens due to the reduction in the resolution of the image after the feature extraction module described below. The new resolution is calculated based on the minimum space that the robot can move in a safe way. Therefore, any higher resolution provide for a better hardware would be wasted. The next step will detail the techniques used to define the objects positions.

3.2 Feature Extraction

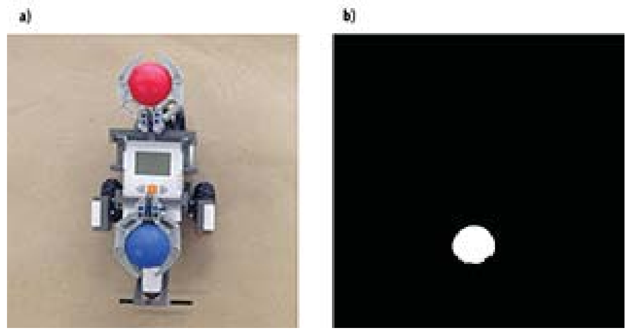
This step has a big importance to the system, since the navigation of the robot is based on the feature extraction of the image obtained in the previous step, without intervention of any other sensor. This step is subdivided in two groups of interest, one used for navigation and the other to construct the occupancy grid map:

1) Robot's Position and Direction;
2) Obstacles and Target.

3.2.1 Robot's Position and Direction

As described before, the robot has details in red and blue on its edges, then to determine the robot's position comes down to locate the region that contains the greatest intensity of one of these colours. Our approach uses the thresholding technique described in [4], to determine the region in the image that has a given colour. To detect the blue colour, for example, he technique on channel zero index of image, representing only the intensity of blue in each one pixel. It is normal that images with regions in blue colour show greater intensity on this channel as compared to regions of other colours, which are also composed of blue. The result is a binary image, indicating regions that present values higher than the threshold determined.

Fig. 4: a) Segmented image of the robot. b) Binary image applying threshold 70 on the blue channel.



Once the blue region of the robot is identified, the robot's position on the map is determined by the central pixel of this area. The Python function that returns the median pixel in shown in Fig. 5

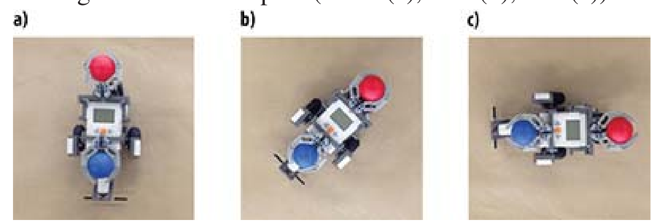
Fig. 5: Python function that returns the median pixel of a colour region

```
def getMedianPixel(self, color, threshold):
    image = npy.array(self.image,int)
    pic = npy.array(image[:,:,self.color])
    averagePic = npy.array(image,int)
    pic -= npy.average(averagePic,2)
    image = npy.array(255 * (pic > threshold))
    hFlat = npy.sum(image,0)
    vFlat = npy.sum(image,1)
    hIndex = 0
    vIndex = 0
    i = 0.
    for val in hFlat:
        hIndex += val * i
        i += 1.
    hIndex /= npy.sum(hFlat)
    i = 0.
    for val in vFlat:
        vIndex += val * i
        i += 1.
    vIndex /= npy.sum(vFlat)
    return int(hIndex),int(vIndex)
```

Likewise the previous step, the median pixel of the red region is found, and assists in determining the direction of the robot. In order to select the best technique to be used in the system, two approaches were implemented and analysed.

The first approach uses SVM (Support Vector Machine), a machine learning method presented by [14], to classify which class a given image belongs. To implement this procedure, is used the scikit-learn library, that is able to classify multiple classes. Five classes were criated: Up, Down, Left, Right, and Null, corresponding to valid directions (multiple of 90 degrees) for navigation and Null for any other invalid direction (see Fig. 6).

Fig. 6: Class examples (Down(a), Null(b), Left(c))



To train the SVM, a set of several images was provided, and the training parameter was calculated by the versor $[vx, vy]$ of the median pixel of the blue region $(x2, y2)$, with origin the median pixel of the blue region $(x1, y1)$. The versor is computed as follows:

$$\begin{aligned} vx &= \frac{(x2 - x1)}{\sqrt{(x2 - x1) + (y2 - y1)}} \\ vy &= \frac{(y2 - y1)}{\sqrt{(x2 - x1) + (y2 - y1)}} \end{aligned} \tag{1}$$

The second approach uses only geometric equations to determine the direction, using Python numpy library. The arctan2 function receives as argument two values that indicate the direction of the vector in a Cartesian plane and returns the corresponding value of the angle formed by these points in radians, in the range $[-\pi, \pi]$. With the vector direction with the coordinates to the edges of the robot, is used the *rad2deg* function to transform the result in degrees.

Fig. 7: Python function that returns the angle relative to the Y axis.

```
def findAngle(self):
    y = (self.blueY-self.redY)
    x = (self.blueX-self.redX)
    return (npy.rad2deg(npy.arctan2(x,y)))
```

With a preliminary simple comparison, it was observed that both produce equivalent results for this problem, being indifferent to to choose of one over the other. However, we opted for the second approach to avoid the SVM training step and storage of large data set required to the training step. In trivial cases like this, is not obvious improvements on the use of Machine Learning techniques, but in facial expressions recognition applications, as in [1], the use of SVM becomes indispensable.

3.2.2 Obstacles and Target

The obstacles placed in the environment block the robot's way, so it has to be identified and indicated as unavailable places in the path planning step. The first approach tested used edge detection algorithm to find the obstacles, but ended up being discarded because it left the inside of the obstacles as free points for navigation.

The obstacles, the Mindstorms NXT® kit white boxes, such as for the robot, are localized by thresholding technique in the blue channel of the image, using a blue threshold just below the value we use to find the robot. The resulting binary image contains the area of the boxes and also, undesirable parts of the robot. This is corrected after removing a rectangle that represents the total area occupied by the robot, around its blue dot calculated previously.

Use the same process of thresholding to determine the goal's central pixel, represented by the green sheet.

Once all the obstacles and goal are detected, the map represented as an occupancy grid is constructed, as described by [11]. All the available and unavailable indexes to navigate and the goal are represent by 0, 1 and 99, respectively.

In this step, with obstacles and targets detected, the map is constructed represented by occupation grid [11], assigning 1 to spaces occupied and 99 for the target in a two-dimensional array.

3.3 Map Update

This step is important for saving a considerable amount of processing time in the system, avoiding unnecessary path planning calculations every time a new image is acquired. This analysis process compares the current positions of the objects with their positions when the path is previously planned, determining if the path needs to be recalculated. Therefore, in case the objects move for some reason, or any intervention in the environment occurs, the robot is able to recalculate the path and reach the goal.

Fig. 8: Python function that indicates a map change and updates it

```python
def mapChange(self):
    newMap = self.readMap()
    mapDiff = npy.abs( newMap - self.map )
    if ( npy.sum(mapDiff) > self.mapThreshold ):
        self.map = newMap
        return True
    else:
        return False
```

3.4 Path Planning

Once the map is constructed, indicating the position of the robot, obstacles and target, we used the A-Star algorithm to find the best path to be executed by the robot. As demonstrated by [9], it is necessary to estimate a consistent heuristic function to have a optimal results. The cost function F is calculated by G + H, where G is the exact cost of the starting point to the current point and approximate heuristic function H is calculated by the number of movements required to reach the goal, excluding the obstacles, moving only horizontally and vertically.

Fig. 9: Python function to calculate the A* cost function F

```python
def G(pos):
    return Dg * (abs(pos[0] - startX) + abs(pos[1] - startY))

def H(pos):
    return Dh * (abs(pos[0] - targetX) + abs(pos[1] - targetY))

def F(pos):
    return G(pos) + H(pos)
```

In order to facilitate the execution of the path, the function returns a new two-dimensional array with same size of the constructed map. It contains only the path, indicating the goal by the number 99, decreasing 1 each movement toward the starting point. The Fig. 10 shows an example of the path returned.

Fig. 10: Path planned by A* algorithm, with starting point [2,3] e goal [15,15]

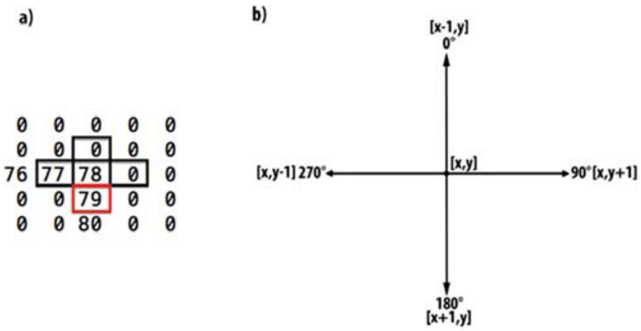
```
[[ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0 74 75 76 77 78  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 79  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 80  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 81  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 82  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 83  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 84  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 85  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 86 87  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0 88 89  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0 90 91 92 93 94  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 95  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 96 97  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0 98 99  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]]
```

3.5 Robot's communication

Using nxt-python interface, control commands to move the servo motor are given through USB or Bluetooth connections. The use of wires is not feasible in this context, since it influence the feature extraction step and obstruct the movement of the robot in its environment. Therefore, even with the delay in wireless connections, we opted to use Bluetooth.

As proposed by the model, the entire path execution is based on visual information obtained by a webcam. Therefore, every new image acquired, compares the neighbours values of the robots current, considering only horizontal and vertical movements. The highest value determines the direction that the robot must follow, Fig. 11 (a) illustrates an example where the robot is at position [x,y], 78 value, and determines that the highest neighbour value is in the position below, [x + 1, y] highlighted in red, readjusting his new direction to 180 degrees relative to the Y axis.

Fig. 11: a) Robot's direction adjustment, considering the robot is at the position with value 78. b) Angle correction based on the highest nighbour



Our developed approach is able to reach the goal even with uncertainty and variability of servo motors described by [13], which causes deviations even in simple commands like move forward.

3.6 Goal test

To determine if the goal is reached, we compare the coordinates of the robot and the goal, acquired in the feature extraction step. If they have the same value, the system finished. Otherwise, the system keeps sending commands to move the robot towards the goal. The function is shown below:

Fig. 12: Python goal test function

```
def checkFinish(self):
    if (self.targetX == self.blueX and self.targetY == self.blueY):
        cv2.VideoCapture(0).release()
        return True
    else:
        return False
```

4. Conclusions

The present work showed that, through a computer vision system, a mobile robot in a restricted environment layout can be guided from one point to another without needing human intervention. To perform this task, studies were carried out in search algorithms, machine learning, computer vision, image processing and robotic navigation to ensure the robot is able to plan and execute the path satisfactorily.

Using thresholding techniques to object detection and A* search algorithm to path planning, it was possible the system, validating the proposed autonomous navigation method. However, some aspects should be carefully analysed, such as the computational cost of image processing and the processing capacity of the hardware involved, since they are determining factors to the system execution.

During the experiments, where the feature was evaluated to determine if the robot had reached the target or not, undesirable characteristics were observed in the system, which can be improved in a future work. As an example, one can cite that, in some cases the system drew trajectories very close to obstacles. Even having enough space to keep distance, providing possible collisions when doing bends near the edges of obstacles, or even in cases which the robot had totally unexpected behaviour, caused by different light conditions in the environment forcing the recalculation of the thresholds for the correct identification of the objects position.

As possible improvements for future work, we would like to:

- Use HSI colour system instead of RGB, aiming to get better results in different light conditions.
- Use a safety margin around the obstacles to avoid collisions.
- Change the path planning algorithm to select the best path with the minimum possible number of turns, in order to save time on the robot's readjustment direction.
- Create a graphical user interface to simulate objects position identified by the system.

5. Acknowlegment

The authors would like to thank the financial support of the Brazilian National Council for Scientific and Technological Development (CNPq), the Foundation for Research Support of the State of Goias (FAPEG) and the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES).

References

[1] Ari, I., Uyar, A., and Akarun, L. Facial feature tracking and expression recognition for sign language. In *Computer and Information Sciences, 2008. ISCIS'08. 23rd International Symposium on* (2008), IEEE, pp. 1–6.

[2] Forsyth, D. A., and Ponce, J. *Computer Vision: A Modern Approach*, 2nd ed. Pearson, 2012.

[3] Fuke, Y., and Krotkov, E. Dead reckoning for a lunar rover on uneven terrain. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on* (1996), vol. 1, IEEE, pp. 411–416.

[4] Gonzalez, R. C., and Woods, R. E. *Digital Image Processing*, 3rd ed. Pearson/Prentice Hall, 2007.

[5] iRobot. irobot roomba: Vacuum cleaning robot. Online : http://www.irobot.com/For-the-Home/Vacuum-Cleaning/Roomba.aspx, 2015.

[6] Kadous, M. W., Sheh, R. K.-M., and Sammut, C. Caster: A robot for urban search and rescue. In *Proceedings of the 2005 Australasian Conference on Robotics and Automation* (2005), pp. 1–10.

[7] Nelson, W. L., and Cox, I. J. Local path control for an autonomous vehicle. In *Autonomous robot vehicles*. Springer, 1990, pp. 38–44.

[8] Prince, S. J. D. *Computer vision : models, learning, and inference*. Cambridge University Press, 2012.

[9] Russell, S. J., and Norvig, P. *Artificial Intelligence: A Modern Approach*, third edition ed. Prentice Hall, 2010.

[10] Secci, H. A. *Uma Introdução a Robôs Móveis*, 2008.

[11] Souza, A., Maia, R., Aroca, R., and Goncalves, L. Probabilistic robotic grid mapping based on occupancy and elevation information. In *Advanced Robotics (ICAR), 2013 16th International Conference on* (Montevideo, 2013), pp. 1–6.

[12] SZELISKI, R. *Computer Vision: Algorithms and Applications*. Springer, 2011.
[13] THRUN, S., BURGARD, W., FOX, D., ET AL. *Probabilistic robotics*, vol. 1. MIT press Cambridge, 2005.
[14] WESTON, J., MUKHERJEE, S., CHAPELLE, O., PONTIL, M., POGGIO, T., AND VAPNIK, V. Feature selection for svms. In *NIPS* (2000), vol. 12, pp. 668–674.

Determining humanoid soccer player position based on Goal detection

V. Lomas-Barrie[1], **M. Peña-Cabrera**[1], **and J. Durán-Ortega**[1]
[1]Instituto de Investigaciones en Matemáticas Aplicadas y Sistemas, Universidad Nacional Autónoma de México, Apdo. Postal 20-726, México D.F. , México

Abstract – *Find the goal in a game from the RoboCup Soccer Humanoid League using computer vision is one of the main tasks in this kind of competitions. Detect the contrary team's goal it is important, if one of the humanoid team members seeks its own goal the robot could share the information with the rest of the members team. In this paper, the Humanoid University team focus in a method which not only detect the contrary or own goal but also find the humanoid relatively position to the goal. The target is extract a goal descriptor vector (Boundary Object Function method) from different points of view (a goal's point-of-view map). With all of those vectors an Artificial Neural Network FuzzyARTMAP is trained. Thus, using the classification FuzzyARTMAP process, each humanoid can aim to the goal and figure it out where is itself located.*

Keywords: Humanoid, soccer, detection, BOF, FuzzyARTMAP, position.

1 Introduction

The RoboCup Soccer Humanoid League, is focused in design, develop and test several algorithms to improve the abilities of each robot in a Soccer Game. Walking, running, kicking the ball, detect the goal, ball, each robot positions, etc are a common challenge for the participants in this League.

The Humanoid University team, specifically the robot vision sub team, is seeking for new methods to detect the goal, the ball, each humanoid and even the contrary robot team.

The procedure to recognize the goal and the humanoid position are described in this paper split in three main sections. The first one, methodology, explain the algorithm, its details and considerations that were implemented. Color segmentation, extract the Boundary Object Function (BOF) [1] in order to train an Artificial Neural Network (ANN) FuzzyARTMAP [2] and the mapped field humanoid auto localization. The second part shows the tests and results. And finally, the conclusions.

1.1 Technical aspect of humanoid soccer player

There are three categories in the RoboCup Soccer Humanoid League: KidSize (40-90 cm height), TeenSize (80-140 cm height) and AdultSize (130-180 cm height) [3].

The Humanoid University team purpose is gain expertise in the KidSize category and in the next year compete in AdultSize category.

In KidSize category only 4 robots can play. One of them as a goal keeper. The object of the game is to score by getting the ball into the opposing goal (FIFA rules-like)

The goal is 110 cm height and 225 cm width. And is yellow painted. The light conditions depends where the site competition is built.

Every robot have to be similar to the human body. The type and the number of sensors are limited as many as a human has. Therefore, Therefore, it only allows a maximum of 2 cameras in the visible spectrum.

1.2 Other algorithms

In the RoboCup Soccer Humanoid League are used several algorithms to detect the goal, the ball, and the lines of the field.

An example of the algorithms based on geometric characteristics is the combination Canny Edge Detector [4] and the generalized Hough transform [5]. After some image processing, the edge of the field lines or goal posts obtained are converted in a mathematical representation. Only the lines, dots or circles that belongs to certain set of rules are accepted as the goal, the field lines or even the circle.

Algorithms based on the color of a group of pixels is another feasible option to identify marks. A feature vector contains the information of a pixel or group of pixels related to their spatial position and/or the color information [6]. A lot of Lookup tables (LUT) need to be generated to associate the color groups to an specific mark. Using Support Vector Machines (SVM) is an option to process the image.

The algorithm presented in this paper, combines colored based processing and geometrical conditions of highlighted objects.

2 Methodology

In order to explain the process to detect the goal and in consequence the humanoid position, first the steps are explained and then, in every subsection, the specific algorithms are shown.

The steps to detect the goal and auto locate the robot are:

1. Get an image of the goal. If the goal is not in the scene turn the head or the body until the goal appears. Note: It needs to complete this full procedure to determine if the goal is in the scene or not.

2. Convert the RGB image to a HSV (Hue-Saturation-Value) color space.

3. Apply a mask in order to get only the goal in the scene.

4. Extract the boundary points. Only the internal ones.

5. Calculate the goal centroid.

6. Calculate the BOF.

7. Match the correct BOF according to the robot position. Before, the ANN FuzzyARTMAP needs to be trained.

The main steps are described below.

2.1 Color segmentation

The video sensor used by the humanoid is color based. Three channels: red, green and blue are detected. The goal and the ball color are yellow (rule book, 2014). But the light conditions in the court change depending of several factors. For instance, the sun light during the day, the clothes color of the attending audience, the background or the camera flashes. Those changing aspects make difficult to recognize the yellow color.

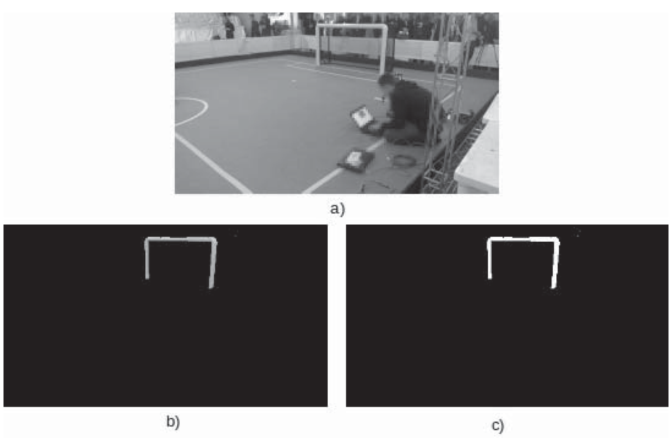


Figure 1 a) Original image. b) segmented image, c) mask image

Changing from RGB to HSV color spaces has been a commonly used technic in this kind of competitions. Due to yellow hue range is limited to certainly values, using a mask can segment only yellowed color objects. The Figure 1 shows the goal filtered using a HSV mask. The range mask is [[20,130,130] – [40,255,255]]. The hue range is from 20 to 40 degrees. The saturation varies from 50.9% to 100%; the value range from 50.9% to 100%. These values are adapted in every competition site where the humanoids play.

2.2 Boundary Object Function

The Boundary Object Function (BOF) [1] is a method to extract a characteristic vector from an object. BOF is based on determining the geometric distance between the boundary points of an object to its centroid. This vector is a discrete function that can be learned by an ANN FuzzyARTMAP. The algorithm has probed that it is invariant to scaling, translation and rotation.

The final implementation will be in an FPGA (Filed-programmable gate array), therefore, the team tested this method to recognize the goal. Thus, the BOF and ANN FuzzyARTMAP were implemented in C programming language and OpenCV.

An alternative method to find the contour object is stated in [7]; but it only works with objects whose centroids are contained within the object itself; the goal centroid does. This is because in the trajectory seeking the border point, could find more than one boundary point. In this way, the vector BOF will be filled with centroid-boundary-points disordered. This means that 2 border points with the same trajectory from the centroid might be in different position in the BOF array, which is inconvenient for the learn process. The FuzzyARTMAP might confuse the array elements and create an incorrect neural.

However, modifying some restrictions in the find border points algorithms the method is available to get the Boundary Object Function with objects whose centroid is outside of the object area (Figure 2).

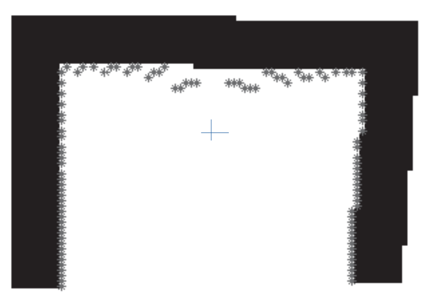


Figure 2 Boundary inside points and centroid

It should be noted that only the inside boundary points of the goal are necessary. Hence, will never exist boundary points surrounding the 360 degrees from the centroid.

Once the HSV image conversion has been done over the goal image, the steps to calculate the BOF modified algorithm are the next:

1. Convert the image to a binary image. 255 pixel value for the background and 0 pixel value for the segmented goal.

2. Calculate the goal centroid.

3. Trace a straight line which cross the centroid with an angle φ. The first iteration, φ = 0 degrees.

$$y = \tan(\varphi n)x + b_0 \tag{1}$$

4. This straight line is the trajectory scanning for a value change from 255 (background) to 0 (goal).

5. When occurs that change, it is for sure that a boundary point exist. Then the pixel coordinates are stored in the boundary point vector.

6. Increase φ until surround the 360 degrees from the centroid.

7. In case the scanner does not find a boundary point but the limit of the image has been reached, it breaks the loop and increase φ again until π is reached.

8. Normalize all BOF elements; divide each element by the maximum.

The Figure 3 shows the BOF graphic. The number of the BOF elements depends on the precision settings defined for how many straight lines are traced. More elements, better ANN matching, however more computing time is used. And vice versa. It is important to find a balance between these 3 characteristics. In this case 111 BOF elements were set.

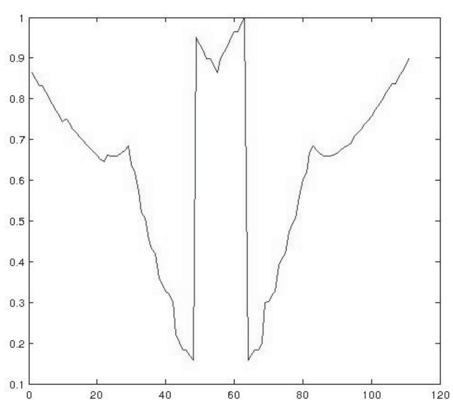


Figure 3 BOF

2.3 Mapping humanoids on the field

Recognizing the goal using BOF & FuzzyARTMAP is not the only purpose of this paper. Knowing an approximated position of each robot is very important for the strategy of the game. Not only for aim the opposite goal, and kick the ball in that direction, but also figure out where every robot is located for plan the next move (not discussed in this paper).

Taking several images of the goal from different views, the BOF goal will change. Hence doing a map of several points of view on the field and taking an image of the goal from each position is possible to obtain BOFs. That BOFs can train an ANN FuzzyARTMAP. Every BOF is a class, a position on the field. The Figure 4 shows a grids representing several classes.

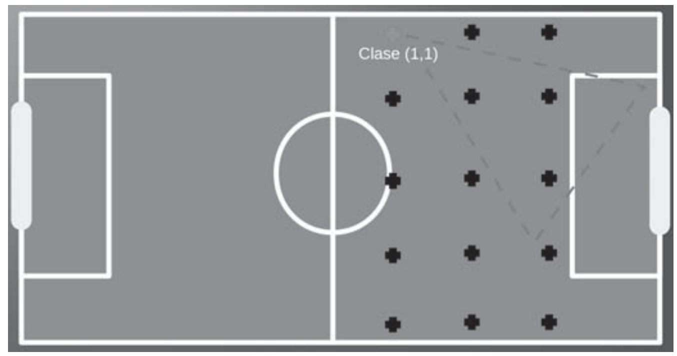


Figure 4 Mapping the field

Due to that the court is symmetrical, it is only necessary to train the ANN with the half of the field. Even only a quarter field could be used. But in order to improve the matching results it is better to use the half of the field.

The process shown in Figure 5 is an example of three main points of views converted in its corresponding BOFs.

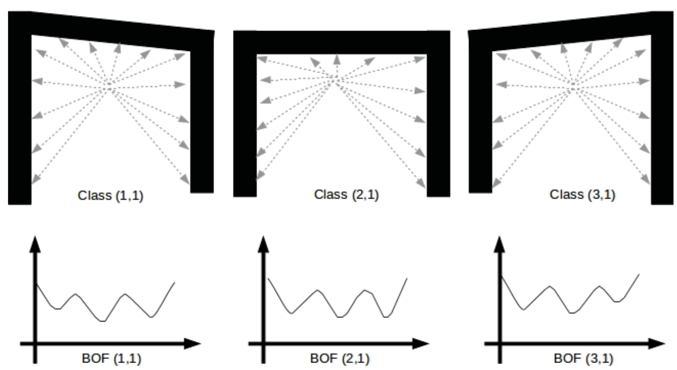


Figure 5 BOFs examples from different views

Highlight the fact that every recognized BOF represents a point of view, immediately the humanoid will know where it is located.

In the case of two or more BOFs are quite similar and the FuzzyARTMAP classifier does not notice any difference, an extra characteristic can be added to the class. In order to strength the matching process, the original distance from each boundary point to centroid can be attached.

In order to make the algorithm invariant to scaling and rotation, the BOF vector is normalized. However mapping the point of view could bring more than two BOF very similar. Therefore, preserve the original distances from boundary points and centroid help to distinguish several points of view. Larger the distances, the points of view are closer to the goal and vice versa. In an array are stored these ranges and their corresponding field position.

At the beginning of the game or when a robot is back to the game after repairing, one of the main tasks is to locate the opposite goal. Then, the robot turns its head until recognize the goal or the ball, it depends of the robot role. If the humanoid couldn't recognize any, starts walking in order to turn the whole body to find the goal.

Even, if a robot has fallen and before it stands up, the goal seeking procedure starts to detecting the opposite or own goal. However the goal keeper is marked with a specific color. Thus, whether is the own or opposite goal, the auto localization of each robot is guarantee. If it detects the goal and the team's goal keeper, it infer that only turning the body can aim to the opposite goal.

3 Test and Results

The process to test this new method consisted in taking three image of the goal from different positions; goal left (GL), goal center (GC) and goal right (GR). The coordinates (in meters) for each point of view is as follows: GL(4.5, 0.0), GC(4.5, 3.0) and GR(4.5, 6.0). The origin is the left corner of the opposite half field. To these three images, the BOF was extracted (Figure 6). Then the FuzzyARTMAP was trained. It was created 3 different classes.

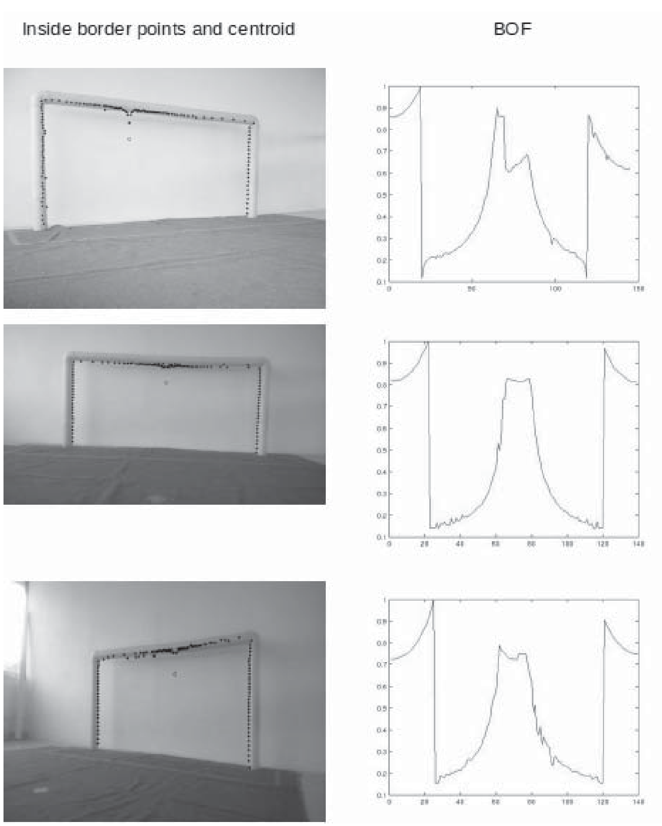


Figure 6 Processing the Classes GL, GC and GR

In the second stage, new images were taken, however, this time the FuzzyARTMAP worked as a classifier. In the Table 1 the results are shown.

From eleven points of view the algorithm was tested. In most of the cases a match occurred. And the recognition was succeeded. In two cases, there were mismatches. The image I7 from the point of view which its coordinates are (4.0, 3.2)m should have been classified with the GC class. Because the position is to close to the center (4.5, 3.0)m and a mismatch occurred.

Image Coordinates [m]	Class GL	Class GC	Class CR	Result
I1 (4.2, 0.0)	✓	X	X	Match
I2 (4.2, 0.2)	✓	X	X	Match
I3 (4.5, 2.8)	X	✓	X	Match
I4 (4.7, 3.0)	X	✓	X	Match
I5 (4.5, 3.2)	X	✓	X	Match
I6 (4.7, 3.2)	X	✓	X	Match
I7 (4.0, 3.2)	✓	X	X	Mismatch
I8 (4.2, 6.0)	X	X	✓	Match
I9 (4.7, 6.0)	X	X	✓	Match
I10 (4.6, 6.0)	X	X	✓	Match
I11 (4.6, 6.0)	X	✓	X	Mismatch

Table 1 Classification test

The same behavior can noticed with the image I11 correspondence with point of view (4.6, 6.0)m. Is approximated to the GC Class position.

4 Conclusions

The results of the tests were favorable. It succeeds in 72.72% of the cases. In spite of two mismatches, the proximity with other point of view confused the FuzzyARTMAP classes. The images I7 and I11 have similar BOFs with GL Class and GC Class, respectively. Thus the lack of more classes, not only among GL and GC, produces a wrong match. However, if more classes are calculated from all over the field, the classification would be more accuracy.

The fact of the consideration that the same goal viewing from different points of view is a different object, results in a convenient application for this BOF object recognition algorithm used mainly in manufacture area.

The main reason to probe this object recognition method in the goal detection is because BOF algorithm has fewer random accesses to the memory. Even, it is not necessary to store the whole image in memory neither the inter created image, resulted from the image processing.

Due to the purpose is to implemented this goal detection and auto localization method in hardware using descriptor languages; in an FPGA. With only fetch 3 consecutive lines of the image it is possible to get the centroid and border points of the goal. Implement the FuzzyARTMAP classifier and HSV segmentation on an FPGA is a future research task for the team. It is feasible get in only one chip the video processing algorithms attached to video sensor.

The restrictions of this method, described above, are the difficulties to watch the whole goal in only one frame. If the robot is to close to the goal, the size of the goal makes impossible recognizes it with this algorithm. But, as were mentioned before, exist the possibility to detect the own goal for the forward player or the opposite goal for the defense or the goal keeper.

If it is required more position accuracy then is necessary to train the ANN with more BOF's points-of-view. But it requires more computer process. Even though the computer time was not something to consider, the fact that the FuzzyARTMAP can learn several information it requires certain differences between the classes in order to distinguish them.

Auto localization allows share the information with the rest of the team. Thus, spread the possibilities to explore more algorithms to organize the team, knowing the approximated position of each member.

5 References

[1] M. Peña-Cabrera et al. (2004). “A Learning Approach for On Line Object Recognition in Robotic Tasks”, Mexican International Conference on Computer Science ENC , IEEE Computer Society Press.

[2] Carpenter Gail A. and Stephen Grossberg, John H Reynolds. (1991) “ARTMAP: Supervised Real-Time Learning and Classification of Nonstationary Data by Self-Organizing Neural Network”. Neural Networks. Pp 565-588.

[3] RoboCup Soccer Humanoid League. (2014). “RoboCup Soccer Humanoid League Rules and Setup For the 2014 Competition”.

[4] Canny, J. (1986). “A Computational Approach To Edge Detection”, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698.

[5] Duda, R. O. and P. E. Hart. (1972). "Use of the Hough Transformation to Detect Lines and Curves in Pictures," *Comm. ACM, Vol. 15*, pp. 11–15

[6] Budden, D., Mendes, A. (2014). “Unsupervised recognition of salient color for real-time image processing”, Document Lecture Notes in Computer Science.

[7] M. Peña-Cabrera, V. Lomas-Barrie. (2012). “Contour Object Generation in Object Recognition Manufacturing Tasks” Worldcom 2012.

SESSION

GENETIC ALGORITHMS + EVOLUTIONARY STRATEGIES AND COMPUTATIONS + OPTIMIZATION METHODS

Chair(s)

TBA

Bound Smoothing with a Biased Random-Key Genetic Algorithm

Thiago Alves de Queiroz[1], **Ivan da Silva Sendin**[2], **and Marcos Aurélio Batista**[3]
[1]Institute of Mathematics and Technology, Federal University of Goiás - Campus Catalão, Catalão, Goiás, Brazil.
E-mail: taq@ufg.br.
[2]Faculty of Computing, Federal University of Uberlândia, Uberlândia, Minas Gerais, Brazil.
E-mail: sendin@ufu.br.
[3]Department of Computer Science, Federal University of Goiás - Campus Catalão, Catalão, Goiás, Brazil.
E-mail: marcos.batista@catalao.ufg.br.

Abstract— *In this work we solve a subproblem of the distance geometry problem in molecular conformation. The latter aims to determine the three-dimensional structure of a molecule from a set of imprecisely distances. We are interested only in the bound smoothing subproblem, which aims to tighten bounds from a set of lower and upper bounds on distance for pair of atoms. We apply a Biased Random-Key Genetic Algorithm to solve this subproblem, where each entry of chromosomes indicates how to decrease a bound, with the fitness function measuring the violation of the triangle inequalities. Experimental results show the effectiveness of this algorithm to solve randomly generated instances for which real distances between atoms are known in advance.*

Keywords: Molecule Problem; Distance Geometry Problem; Triangle Inequality; Biased Random-Key Genetic Algorithm

1. Introduction

A important problem in computational biology is to determine the structure of a molecule, as a protein [1], [2]. Some experimental techniques have been applied to measure interatomic distances for molecules, as the Nuclear Magnetic Resonance (NMR) [3], [4] and X-ray crystallography [5]. After measuring the distance between atoms, the next step is to construct a three-dimensional structure, that is, solve a variant of the distance geometry problem, which is NP-hard accordingly to [6].

The distances measured with NMR technique are not precise, so many of these distances need to be given in lower and upper bounds. Then, it is necessary to tighten the bounds in order to construct a well structured molecule. In this case, a distance geometry problem has to be solved [7]. In accordance with [8], this problem can be organized in three subproblems (steps): (i) bound smoothing, (ii) embedding, and (iii) optimization.

Step (i) aims to improve the bounds given on the distance of each pair of atoms as well as to compute bounds for those distances that experimental methods could not estimate, so aiming to obtain tight bounds for each pair of atoms. In step (ii), once the bounds are tightened, it is computed approximate tree-dimensional coordinates for the atoms, and they are further improved in step (iii). Steps (ii) and (iii) are repeated until to satisfy all lower and upper bounds. More details of the distance geometry problem are given in [9].

Experimental approaches that have been used to calculate a distance for a pair of atoms are not precise. On the NMR, the structure must have a small quantity of residues, so the distance can be estimated for pairs of atoms that are at a short distance (smaller than 6 Å) On the other hand, the X-ray crystallography imposes a crystallization of the structure, which is very time consuming [10].

As only a set of distances can be estimated from experimental approaches, it is necessary to compute the remaining ones to obtain a complete structure. The distance geometry problem is closely related to those in the areas of sensor network localization [11], image recognition and euclidean distance matrix completion [12], an extensive review about distance geometry is avaliable at [13].

The major contribution to the distance geometry problem with applications in the field of molecular conformation was given by [9]. They proposed an algorithm, which determines coordinates for atoms from experimental data. Their algorithm solves the three subproblems previously mentioned. Another approaches used to solve the distance geometry problem were proposed by [14], based on graph reduction, in which an input graph is decomposed in subgraphs and, next, an embedding problem is solved; by [15], based on global optimization to solve a weighted function, which does not need to consider all bounds, so this function is transformed on easier functions that are quickly solved; by [16], that considered the geometric buildup algorithm, which can solve the problem in linear time if all distances are known; by [17], based on stochastic search; and, by [10] that developed a new version of the geometric buildup algorithm for a generalized version of the distance geometry problem. The latter aims to find the equilibrium positions and the maximal bounds for each pair of atoms.

We are only interested on step (i), that is, to solve the bound-smoothing subproblem. The objective of the Bound-smoothing is to check consistency and give information about the distance between pairs of atoms. So, this implies to tighten the initial lower and upper bounds that are know as well as to compute bounds for pair of atoms whose distance is unknown.

The bound-smoothing problem has been solved so far essentially by checking triangle (and tetrangle) inequalities for atoms-triplet (atoms-quadruple), since tighter bounds can be obtained if satisfying these inequalities. The computation of the triangle inequality is straightforward, while the tetrangle one requires to calculate the Cayley-Menger determinant, which is constructed on each quadruple of atoms (atoms-quadruple) and it has worst-case complexity time of $O(n^4)$.

Therefore, the literature has been focused on algorithms to solve instances with a large number of atoms by means of the tetrangle inequality. A parallel algorithm working in a shared memory architecture for bound smoothing was proposed in [8]. Their algorithm spends $O(n^3 logn)$ time with $O(\frac{n}{logn})$ processors, and it is based on classify the atoms in independent sets and, then, to color vertices of the resultant graph. They solved satisfactorily instances with up 430 atoms, for which their algorithm required only 5% of the time spent by a sequential algorithm.

Another parallel algorithm was developed in [18] for the Beowulf-type cluster of PCs, which requires $O(\frac{n^4}{p})$ time for $p < \frac{n}{6}$ processors. This algorithm organizes atoms in subsets, so atoms-quadruple can be classified at one out of five types according to the subset that each atom is in. They solved one instance with 630 atoms with efficiency arriving at 60%.

In order to solve the bound-smoothing problem satisfactorily, we considered the biased random-key genetic algorithm (BRKGA) [19]. The BRKGA has been applied with successes to solve classical optimization problems, such as in the area of covering [20], packing [21], scheduling [22], and clustering [23]. A C++ framework of this algorithm was made available in Internet by [24].

This paper is organized as follows. In Section 2, a formal description of the problem is given, while in Section 3, the Biased Random-Key Genetic algorithm is introduced and it is explained how this algorithm is used to solve the bound-smoothing problem. Numerical experiments are presented in Section 4 and they give details about the algorithm and its efficiency to deal with this problem. Some randomly generated instances are solved and the results are compared with the original structure. Finally, some conclusions and directions for further works are given in Section 5.

2. Bound-smoothing Problem

In the Bound-smoothing Problem (BsP), let $\mathcal{S}$ be the set of pairwise atoms with coordinates in $\mathbb{R}^3$, for m atoms. In the set $S_1 \subset \mathcal{S}$, for each pair of atoms $\{a, b\} \in S_1$, it is known the exact Euclidean distance d_{ab}. On the other hand, given $S_2 \subset \mathcal{S}$, for each pair $\{i, j\} \in S_2$, we have $l_{ij} \le d_{ij} \le u_{ij}$, where l_{ij} and u_{ij} are the lower and upper bounds on the distance d_{ij}. And, for the last set, namely $S_3 \subset \mathcal{S}$, there is no information on the distance d_{kl} for each pair $\{k, l\} \in S_3$. Moreover, $S_1 \cup S_2 \cup S_3 = \mathcal{S}$ and $S_1 \cap S_2 \cap S_3 = \emptyset$. The objective of the BsP is to compute bounds for the pair of atoms in S_3, and then, to tighten the bounds in S_2 and S_3 while satisfying distance constraints, as those imposed by triangle and tetrangle inequalities.

In the case of the triangle inequalities, any three points (atoms-triplet) have to satisfy the triangle inequality, so we can improve the lower and upper bounds computing this inequality for all atoms-triplet. In other words, for (i, j, k), with $l_{ij} \le d_{ij} \le u_{ij}$, $l_{ik} \le d_{ik} \le u_{ik}$ and $l_{jk} \le d_{jk} \le u_{jk}$, the triangle inequality $|d_{ik} - d_{jk}| \le d_{ij} \le d_{ik} + d_{jk}$ must hold. Then, the bounds can be improved by:

$$\begin{aligned} \overline{u}_{ij} &= \min\{u_{ij},\ u_{ik} + u_{jk}\}, \\ \overline{l}_{ij} &= \max\{l_{ij},\ l_{ik} - u_{jk},\ l_{jk} - u_{ik}\}, \end{aligned} \tag{1}$$

where $\overline{l}_{ij}$ e $\overline{u}_{ij}$ are the new values for the lower and upper bounds, respectively.

Observe that $\overline{u}_{ij}$ can be calculated independently of $\overline{l}_{ij}$. Moreover, if $u_{ik} \ge \overline{u}_{ik}$, $u_{jk} \ge \overline{u}_{jk}$ and $\overline{l}_{ij} = \max\{l_{ij},\ l_{ik} - u_{jk},\ l_{jk} - u_{ik}\}$, then we can again improve $\overline{l}_{ij}$ performing $\overline{l}_{ij} = \max\{l_{ij},\ l_{ik} - \overline{u}_{jk},\ l_{jk} - \overline{u}_{ik}\}$. Whenever $\overline{l}_{ij} > \overline{u}_{ij}$, it follows that the original values are restored.

A straightforward algorithm to update the bounds considers all atoms-triplet and first it computes upper bounds. Next, all triplets are checked for lower bounds. Algorithm 1 summarizes these steps.

Algorithm 1: Triangle inequality for the BsP.

Input : Lower bound L; Upper bound U; Sets S_1, S_2 and S_3.
Output: Tightened lower and upper bounds $\overline{L}$ and $\overline{U}$, respectively.

Let T be the set of all atoms-triplet.
foreach *triplet* $(i, j, k) \in T$ **do**
  if $\{i, j\} \notin S_1$ **then**
    $\overline{u}_{ij} \leftarrow \min\{u_{ij},\ u_{ik} + u_{jk}\}$.
$\overline{U} \leftarrow$ all upper bounds $\overline{u}$.
$U \leftarrow \overline{U}$.
foreach *triplet* $(i, j, k) \in T$ **do**
  if $\{i, j\} \notin S_1$ **then**
    $aux = \max\{l_{ij},\ l_{ik} - u_{jk},\ l_{jk} - u_{ik}\}$.
    if $aux \le \overline{u}_{ij}$ **then**
      $\overline{l}_{ij} \leftarrow aux$.
    else
      $\overline{l}_{ij} \leftarrow l_{ij}$.
$\overline{L} \leftarrow$ all lower bounds $\overline{l}$.
return *sets* $\overline{L}$ *and* $\overline{U}$.

We consider that u is from U and l is from L, while $\overline{u}$ is from $\overline{U}$ and $\overline{l}$ is from $\overline{L}$. After to tighten the bounds by applying the triangle inequality algorithm, a bound may still have some slack, that is, $u_{ik} + u_{jk} - u_{ij} > 0$ and $l_{ij} - l_{ik} - u_{jk} > 0$. In this case, it can be eliminated by computing all-pairs shortest paths [25].

A better way to improve the bounds is to check the tetrangle inequalities for all atoms-quadruple. The tetrangle inequality is associated with the calculation of Cayley-Menger determinants. For a quadruple (i, j, k, r), its determinant $\det_{CM}(d_{ij}, d_{ik}, d_{ir}, d_{jk}, d_{jr}, d_{kr})$ is:

$$\begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & (d_{ij})^2 & (d_{ik})^2 & (d_{ir})^2 \\ 1 & (d_{ji})^2 & 0 & (d_{jk})^2 & (d_{jr})^2 \\ 1 & (d_{ki})^2 & (d_{kj})^2 & 0 & (d_{kr})^2 \\ 1 & (d_{ri})^2 & (d_{rj})^2 & (d_{rk})^2 & 0 \end{vmatrix} > 0 \qquad (2)$$

Therefore, for each pair of atoms $\{k, r\}$, we obtain 32 inequalities from eq. (2) for the lower bound l_{kr} as well as 32 inequalities for the upper bound u_{kr}. However, only seven of these inequalities are non-redundant, with three for the upper bound u_{kr}:

$$\begin{aligned} &\det_{CM}(l_{ij}, u_{ik}, u_{ir}, u_{jk}, u_{jr}, u_{kr}) > 0, \\ &\det_{CM}(u_{ij}, l_{ik}, l_{ir}, u_{jk}, u_{jr}, u_{kr}) > 0, \\ &\det_{CM}(u_{ij}, l_{ik}, l_{ir}, u_{jk}, u_{jr}, u_{kr}) > 0. \end{aligned} \qquad (3)$$

And, another four ones for the lower bound l_{kr}:

$$\begin{aligned} &\det_{CM}(u_{ij}, u_{ik}, l_{ir}, l_{jk}, u_{jr}, l_{kr}) > 0, \\ &\det_{CM}(u_{ij}, l_{ik}, u_{ir}, u_{jk}, l_{jr}, l_{kr}) > 0, \\ &\det_{CM}(l_{ij}, l_{ik}, u_{ir}, l_{jk}, u_{jr}, l_{kr}) > 0, \\ &\det_{CM}(l_{ij}, u_{ik}, l_{ir}, u_{jk}, l_{jr}, l_{kr}) > 0. \end{aligned} \qquad (4)$$

Among the values of u_{kr} computed in eq. (3), the smallest one is compared to the initial u_{kr}, and the minimum between them are set to $\overline{u}_{kr}$. Similarly, the largest value of l_{kr} among those computed in (4) is compared to the given l_{kr}, and the maximum between them are set to $\overline{l}_{kr}$.

For an inequality $\det_{CM}(d_{ij}, d_{ik}, d_{ir}, d_{jk}, d_{jr}, d_{kr}) > 0$, in which we want to compute the value of d_{kr} knowing the remaining ones, it is necessary to solve a quadratic equation to find its roots. Algorithm 2 presents a straightforward way to tighten lower and upper bounds by computing the tetrangle inequalities.

In order to get better results, it is recommend first apply the triangle algorithm 1. Next, the tetrangle algorithm 2 can be repeated until the largest change in any of the bounds be smaller than a given tolerance value. Unhappily, apply the tetrangle inequality a lot of times may bring on a slow ratio of convergence, once the total number of iterations grows quickly. Observe that Algorithm 2 has worst-case complexity time of $O(n^4)$ as pointed out in [8].

Algorithm 2: Tetrangle inequality for the BsP.

Input : Lower bound L; Upper bound U; Sets S_1, S_2 and S_3.
Output: Tightened lower and upper bounds $\overline{L}$ and $\overline{U}$, respectively.

Let T be the set of all atoms-quadruple.
foreach *quadruple* $(i, j, s, t) \in T$ **do**
  foreach *pair* $\{k, r\}$ *out of the six ones in* (i, j, s, t) **do**
    if $\{k, r\} \notin S_1$ **then**
      $u'_{kr} \leftarrow$ minimum u_{kr} from eq. (3).
      $\overline{u}_{kr} \leftarrow \min\{u_{kr},\ u'_{kr}\}$.
$\overline{U} \leftarrow$ all upper bounds $\overline{u}$.
$U \leftarrow \overline{U}$.
foreach *quadruple* $(i, j, s, t) \in T$ **do**
  foreach *pair* $\{k, r\}$ *out of the six ones in* (i, j, s, t) **do**
    if $\{k, r\} \notin S_1$ **then**
      $l'_{kr} \leftarrow$ maximum l_{kr} from eq. (4).
      $\overline{l}_{kr} \leftarrow \max\{l_{kr},\ l'_{kr}\}$.
$\overline{L} \leftarrow$ all lower bounds $\overline{l}$.
return *sets* $\overline{L}$ *and* $\overline{U}$.

3. Biased Random-Key Genetic Algorithm

The BRKGA was introduced in [26] and [27]. Algorithm 3 summarizes a typical BRKGA framework, as that provided in [24].

The BRKGA has two key features that distinguish it from others traditional genetic algorithm: a standardized chromosome encoding that uses a vector with random keys (*alleles*) uniformly over the interval $[0, 1)$; and, a well-defined evolutionary process which uses parameterized uniform crossover [28].

Therefore, the BRKGA performs the crossover without caring about the feasibility of the solutions generated for the new individuals. This is possible because of the standard chromosome encoding that is responsible to guarantee the feasibility of the decoding function.

The decoding function is the most important of the BRKGA, since it maps the real vector for a valid solution of the problem under consideration, and it is also more time consuming than the other parts of the algorithm. In some problems where feasibility is hard to achieve, the decoder may generate invalid solutions, so a penalty factor is necessary into the fitness for the algorithm converges.

According to algorithm 3, the parameters that must be specified in a BRKGA framework are the size of the chromosomes t, the size of the population pop, the size of

Algorithm 3: BRKGA algorithm.

Generate an initial population Pop.
while *not attend a stopping criteria* **do**
  Decode each chromosome of Pop and extract a solution and its fitness.
  Sort chromosomes of Pop in non-increasing order of fitness. Consider the top pop_e chromosomes to be in an elite set Eli.
  Next generation Qn receives Eli.
  Next generation Qn receives pop_μ randomly-generated new chromosomes.
  Generate $off \leftarrow pop - pop_e - pop_\mu$ chromosomes offspring using the parameterized crossover, for a random parent from Eli and another from $Pop \backslash Eli$.
  Next generation Qn receives off.
  $Pop \leftarrow Qn$.
return *chromosome with the best fitness*.

the elite set pop_e, the number of mutants pop_μ introduced at each generation, and the inheritance probability ρ_e. In the framework of [24], it is also allowed to define the number of independent populations k and the number of threads for parallel decoding $MAXT$.

3.1 Decoder Phase

First, we apply the triangle algorithm 1 to tighten the bounds in S_2 and, next, to compute initial bounds for the pair of atoms in S_3. We do not consider the tetrangle algorithm 2 due its cycling and slow convergence. Let L and U be the new lower and upper bounds after apply the triangle algorithm.

The aim of the decoder is to extract from chromosomes a good solution for the BsP, in which L and U are tightened in accordance with a fitness function.

For the BRKGA, each chromosome ch has size equal to $|S_2| + |S_3|$, where each allele is sequentially associated to a pair of atoms from these sets. For a pair $a = \{i, j\}$, its allele with value v_a ranging in $[0, 1)$ is used to compute new bounds as specified in Algorithm 4.

We apply Algorithm 4 at each allele of the chromosome ch, so new lower $\overline{L}$ and upper $\overline{U}$ bounds are obtained according to the values in ch and the input bounds L and U.

Observe that $\overline{L}$ and $\overline{U}$ do not have bounds such that $\bar{l}_{ij} < l_{ij}$ and $u_{ij} > \overline{u}_{ij}$ for each pair of atoms $\{i, j\}$. On the other hand, some new bounds cannot satisfy triangle inequalities and, then, we compute the violation of these inequalities by Algorithm 5.

Algorithm 4: Bounds for a given allele.

Input : Lower bound L; Upper bound U; pair $a = \{i, j\}$ of atoms; value v_a of the allele.
Output: New bounds for $a = \{i, j\}$.
$avg \leftarrow u_{ij} - l_{ij}$.
$rg \leftarrow \frac{avg \times v_a}{2}$.
$\bar{l}_{ij} \leftarrow l_{ij} + rg$.
$\overline{u}_{ij} \leftarrow l_{ij} - rg$.
return $\bar{l}_{ij}$ *and* $\overline{u}_{ij}$.

Algorithm 5: Violation of triangle inequalities.

Input : Lower bounds $\overline{L}$ and L; Upper bounds $\overline{U}$ and U; Sets $S_1,\ S_2$ and S_3.
Output: Total violation of triangle inequalities.
Let T be the set of all atoms-triplet.
$viol \leftarrow 0$.
foreach *triplet* $(i, j, k) \in T$ **do**
  if $\{i, j\} \notin S_1$ **then**
    $minU \leftarrow \min_{\{\forall\ k \neq i \neq j\}}\{\overline{u}_{ik} + \overline{u}_{jk}\}$.
    if $\{i, j\} \in S_2$ **then**
      if $minU < u_{ij}$ *and* $minU > \overline{u}_{ij}$ **then**
        $viol \leftarrow viol + |\overline{u}_{ij} - minU|$.
    else
      if $minU > \overline{u}_{ij}$ **then**
        $viol \leftarrow viol + |\overline{u}_{ij} - minU|$.
foreach *triplet* $(i, j, k) \in T$ **do**
  if $\{i, j\} \notin S_1$ **then**
    $maxL \leftarrow \max_{\{\forall\ k \neq i \neq j\}}\{\bar{l}_{ik} - \overline{u}_{jk},\ \bar{l}_{jk} - \overline{u}_{ik}\}$.
    if $\{i, j\} \in S_2$ **then**
      if $maxL > l_{ij}$ *and* $maxL < \bar{l}_{ij}$ **then**
        $viol \leftarrow viol + |maxL - \bar{l}_{ij}|$.
    else
      if $maxL < \bar{l}_{ij}$ **then**
        $viol \leftarrow viol + |maxL - \bar{l}_{ij}|$.
return $viol$.

The Root Mean Square Gap (RMSG) has been used to measure the distance between lower and upper bounds [8]. It corresponds to the root mean square of the difference between upper and lower bounds for all the atoms distance, as described in eq. (5).

$$RMSG(\overline{L}, \overline{U}) = \sqrt{\frac{\sum_{\{i,j\} \in \mathcal{S}} (\overline{u}_{ij} - \bar{l}_{ij})^2}{|\mathcal{S}|}} \qquad (5)$$

Naturally, a small value for the RMSG may indicate that lower and upper bounds are close to each other. On the other hand, if the gap between each bound is significantly small, a valid structure cannot be constructed in the embedding and optimization steps of the distance geometry problem, even if all triangle inequalities are satisfied. It is important to mention that to construct a valid structure is necessary to satisfy other requirements, as the angle bond imposed to some pair of atoms.

With this in mind, the chromosome's fitness is the violation value returned by Algorithm 5. The BRKGA objective is to minimize such violation value.

It is worth to mention that a set of numerical tests aiming to minimize the RMSG, subject to satisfy triangle inequalities for all atoms-triplet, was also conducted on instances for which the exact distance between all pair of atoms are known. However, these results shown that the gap for each bound is very small, such that the exact distances are completely out of the initial bounds L and U.

4. Computational Experiments

All algorithms were implemented in the C++ programming language and the experiments occurred in a computer with 4.0 GHz Intel Core i7-4790K processor, 32 GB of memory RAM and GNU/Linux operating system. The BRKGA framework of [24] was used, where the decoder phase was implemented according to Section 3.1.

The experiments were conducted in a set of randomly generated instances as described in [29]. These instances have structure similar to that of original proteins and the exact distance between each pair of atom is known. We consider instances with the number of atoms ranging from 10 to 100, totaling 15 instances, each one with name brN, where N is the number of atoms.

Observe that we can perfectly check whether the exact distances are in the tighter bounds of the BRKGA solution, contrary if we consider real proteins as those available in the Protein Data Bank, for which the exact distances are unknown for almost all pair of atoms. We are interested to verify whether the BRKGA can be used to solve the distance geometry problem satisfactorily, while solving the BsP and returning tighter valid bounds as well.

For each instance, first it is generated coordinates of the atoms by observing the angle between atoms-triplet or -quadruple [30], so the exact Euclidean distance d is computed. Next, in order to generate lower and upper bounds according to a NMR simulation: the distance between atoms $\{i, i+i\}$ and $\{i, i+2\}$ can be measured precisely due to geometric considerations, so they are considered as the exact Euclidean distance ($l_{i,i+i} = u_{i,i+i}$ and $l_{i,i+2} = u_{i,i+2}$) and they are in the set S_1. The remaining bounds for each pair of atoms $\{j, k\}$ are given as:

- If $d_{jk} < 6$Å, then this distance can be estimated using the NMR, so we consider as bounds $l_{jk} = \lfloor d_{jk} \rfloor$ and $u_{jk} = \lceil d_{jk} \rceil$. This pair is in the set S_2;
- Otherwise, the NMR method cannot estimate the distance between j and k, so the lower and upper bounds are unknown. In this case, $\{j, k\}$ is in the set S_3.

The parameters used by the BRKGA are described next. We consider one independent population executing on one thread. So, we have $pop = 10t$, where $t = |S_2| + |S_3|$ is the size of each chromosome, $pop_e = 25\%$ of the population, $pop_\mu = 15\%$ of the population, and, $\rho_e = 65\%$. The BRKGA stops when a maximum number of iterations is reached, that is, if it reaches 2000 iterations.

4.1 Results

We present the results for all the instances in Table 1. Each row of this table has: name of the instance; total time spent (in seconds); fitness value; RMSG for L and U (after to apply the triangle algorithm); RMSG for the best solution computed with the BRKGA; difference (in percentage) between the RMSGs; percentage of reduction of the bounds for all pair of atoms (this is an average value); number of bounds that were tightened (in percentage); number of bounds tightened (in percentage) for which the respective exact distance is not in.

Table 1 presents the results for the set of 15 instances. Only 2 out of these instances had fitness equal to zero (see br10 and br20), that means that all triangle inequalities are satisfied. As we can observe, the time, fitness value and RMSGs increased accordingly the number of atoms increased too.

Although the RMSG was not considered in the fitness function, its value decreased of 55.69% on average (see column "RMSG BRKGA") especially due to the lower and upper bounds that were reduced too. Observe that 100% of the bounds were tightened, for all the instances, where each bound had an average reduction of 44.48%, on average. This average reduction ranged from 42% to 54% considering all the instances.

For the first 5 instances (br10 to br50) the runtime was less than 1 hour. The worst runtime was for the br100, where the BRKGA required more than 40 hours. On average, the BRKGA required approximately 10 hours, so it may not be a satisfactory result, since real proteins have hundreds or thousands of atoms. On the other hand, the parameters of the BRKGA, as the maximum number of iterations and the size of the population, can be decreased as an attempt to improve the runtime.

The last column of Table 1 shows the percentage of the exact distances that do not comply with the bounds found with the BRKGA. Note that we know the exact distance d between each pair of atoms, due to the way that the instances were generated. The objective of this column is to show whether the bounds computed with the BRKGA still contain a satisfactory quantity of exact distances. Observe that, on average, more than 50% of these bounds still keep the exact distance value, although the BRKGA do not know

Table 1: Results for the set of randomly generated instances.

Name	Time (s)	fitness	RMSG triangle	RMSG BRKGA	Difference RMSG (%)	Reduction on bounds (%)	Bounds tightened (%)	Distances out (%)
br10	0.590	0.000	0.709	0.383	45.978	53.882	100.000	46.429
br20	31.410	0.000	1.978	1.727	12.693	47.735	100.000	43.137
br30	282.090	3.600	9.751	4.325	55.651	43.636	100.000	42.593
br40	1,162.050	5.765	17.975	8.347	53.564	42.941	100.000	44.239
br50	3,438.620	15.794	56.005	31.669	43.452	44.275	100.000	53.014
br55	7,719.980	21.910	42.836	24.112	43.711	44.220	100.000	52.322
br60	11,899.100	22.949	58.614	32.759	44.111	44.881	100.000	55.354
br65	17,131.750	30.719	37.350	22.102	40.824	43.113	100.000	50.026
br70	24,398.860	39.458	55.489	31.650	42.962	44.324	100.000	53.951
br75	35,539.910	35.248	33.504	19.269	42.488	43.846	100.000	50.304
br80	49,359.350	55.042	22.823	13.520	40.761	42.744	100.000	46.121
br85	77,411.920	48.220	38.302	22.115	42.261	43.316	100.000	50.220
br90	95,908.230	61.434	51.918	30.286	41.665	42.342	100.000	50.653
br95	112,270.320	81.590	53.572	31.318	41.540	42.667	100.000	52.571
br100	147,035.210	78.890	646.378	354.204	45.202	43.259	100.000	55.207
Average	**38,905.96**	**–**	**75.15**	**41.85**	**42.46**	**44.49**	**100.00**	**49.74**

neither consider any information about the exact distances when solving the problem, since its execution is only guided on check the triangle inequalities. This is an interesting result and it demonstrates that the BRKGA can effectively solve the distance geometry problem.

5. Conclusions

The bound smoothing problem (BsP) that appears as one of the subproblems of the distance geometry problem is addressed in this paper. In the BsP, we have to tighten bounds for a set of pairwise atoms while they must obey distance constraints, as the triangle or tetrangle inequalities.

We solved the BsP with the Biased Random-Key Genetic Algorithm, in which the decoder phase generates a solution from a chromosome by increasing or decreasing lower and upper bounds of the pair of atoms. The chromosome fitness is calculated according to the number of triangle inequalities that are violated.

Good solutions were obtained for the instances under consideration, since the RMSG decreased of 42.46% on average, and the bounds had a reduction of 44.48% on average, as well as all bounds were tightened for all the instances. On the other hand, the number of exact distances that remained in the new tighter bounds was of 50.03% on average, even if all triangle inequalities were satisfied. This may lead to invalid molecules when solving the other subproblems of the distance geometry problem.

After all, we note that there is room for improvements by considering a new way to decoder chromosomes and extract solutions. It is also important to consider other fitness functions, since the computational runtime increased significantly as the number of atoms increased too, if using the triangle algorithm in the fitness function.

Acknowledgements.

The authors would like to thank the Brazilian National Council for Scientific and Technological Development (CNPq), Research Support Foundation of Goiás State (FAPEG) and Federal University of Uberlândia (grant 04/1014/074-PROPP).

References

[1] C. L. Brooks-III, M. Karplus, and B. M. Pettitt, *A Theoretical Perspective of Dynamics, Structure, and Thermodynamics*. New York: Wiley, 1988.

[2] T. E. Creighton, *Proteins: Structures and Molecular Properties. 2nd Edition*. Freeman and Company, 1993.

[3] A. T. Brünger and M. Nilges, "Computational challenges for macromolecular structure determination by x-ray crystallography and solution nmr-spectroscopy," *Quarterly Reviews of Biophysics*, vol. 26, pp. 49–125, 1993.

[4] J. Cavanagh, W. J. Fairbrother, A. G. Palmer, and N. J. Skelton, *Protein NMR Spec- troscopy: Principals and Practice*. Academic Press, 2006.

[5] J. Drenth, *Principals of Protein X-ray Crystallography*. Springer, 2006.

[6] J. B. Saxe, "Embeddability of weighted graphs in k-space is strongly np-hard," in *Proceedings of the 17th Allerton Conference in Communications, Control and Computing*, 1979, pp. 480–489.

[7] L. M. Blumenthal, *Theory and Applications of Distance Geometry*. London: Oxford University Press, 1953.

[8] K. Rajan and N. Deo, "Computational experience with a parallel algorithm for tetrangle inequality bound smoothing," *Bulletin of Mathematical Biology*, vol. 61, pp. 987–1008, 1999.

[9] G. M. Crippen and T. F. Havel, *Distance Geometry and Molecular Conformation*. England: Research Studies Press Ltd, 1988.

[10] B. Hendrickson, "The molecular problem: Determining conformation from pairwise distances," Ph.D. dissertation, Iowa State University, 2010.

[11] P. Biswas, T. Liang, T. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Journal on Transactions on Sensor Networks*, vol. 2, pp. 188–220, 2006.

[12] W. Rivera-Gallego, "A genetic algorithm for solving the euclidean distance matrices completion problem," in *Proceedings of the 1999 ACM Symposium on Applied Computing*, ser. SAC '99. New York, NY, USA: ACM, 1999, pp. 286–290. [Online]. Available: http://doi.acm.org/10.1145/298151.298353

[13] L. Liberti, C. Lavor, N. Maculan, and A. Mucherino, "Euclidean distance geometry and applications," *SIAM Review*, vol. 56, pp. 3–69, 2014.

[14] A. Sit, "Solving distance geometry problems for protein structure determination," Ph.D. dissertation, Cornell University, 1991.

[15] J. Moré and Z. Wu, "Global continuation for distance geometry problems," *SIAM Journal on Optimization*, vol. 7, pp. 814–836, 1997.

[16] Q. Dong and Z. Wu, "A linear-time algorithm for solving the molecular distance geometry problem with exact inter-atomic distances," *Journal of Global Optimization*, vol. 22, pp. 365–375, 2002.

[17] A. Grosso, M. Locatelli, and F. Schoen, "Solving molecular distance geometry problems by global optimization algorithms," *Computational Optimization and Applications*, vol. 43, pp. 23–37, 2009.

[18] N. Deo and P. Micikevicius, "Coarse-grained parallelization of distance-bound smoothing for the molecular conformation problem," in *Distributed Computing*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2002, vol. 2571, pp. 55–66.

[19] J. F. Gonçalves and M. G. C. Resende, "Biased random-key genetic algorithms for combinatorial optimization," *Journal of Heuristics*, vol. 17, pp. 487–525, 2011.

[20] M. G. C. Resende, R. F. Toso, J. F. Gonçalves, and R. M. A. Silva, "A biased random-key genetic algorithm for the Steiner triple covering problem," *Optimization Letters*, pp. 1–15, 2011.

[21] J. F. Gonçalves and M. G. C. Resende, "A parallel multi-population genetic algorithm for a constrained two-dimensional orthogonal packing problem," *Journal of Combinatorial Optimization*, vol. 22, pp. 180–201, 2011.

[22] J. F. Gonçalves, J. J. M. Mendes, and M. G. C. Resende, "A random key based genetic algorithm for the resource constrained project scheduling problems," *Computers and Operations Research*, vol. 36, pp. 92–109, 2009.

[23] C. E. Andrade, M. G. C. Resende, H. J. Karloff, and F. K. Miyazawa, "Evolutionary algorithms for overlapping correlation clustering," in *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation*. ACM, 2014, pp. 405–412.

[24] R. F. Toso and M. G. C. Resende, "A c++ application programming interface for biased random-key genetic algorithms," AT&T Labs Research, Florham Park, New Jersey," Technical Report, 2012.

[25] A. W. M. Dress and T. F. Havel, "Shortest-path problems and molecular conformation," *Discrete Applied Mathematics*, vol. 19, pp. 129–144, 1988.

[26] J. F. Gonçalves and J. Almeida, "A hybrid genetic algorithm for assembly line balancing," *J. of Heuristics*, vol. 8, pp. 629–642, 2002.

[27] M. Ericsson, M. G. C. Resende, and P. M. Pardalos, "A genetic algorithm for the weight setting problem in OSPF routing," *J. of Combinatorial Optimization*, vol. 6, pp. 299–333, 2002.

[28] W. M. Spears and K. A. DeJong, "On the virtues of parameterized uniform crossover," in *Proceedings of the Fourth International Conference on Genetic Algorithms*, 1991, pp. 230–236.

[29] C. Lavor, L. Liberti, A. Mucherino, and N. Maculan, "On a discretizable subclass of instances of the molecular distance geometry problem," in *Proceedings of the 2009 ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2009, pp. 804–805.

[30] C. Lavor, "On generating instances for the molecular distance geometry problem," in *Global Optimization*, ser. Nonconvex Optimization and Its Applications, L. Liberti and N. Maculan, Eds. Springer US, 2006, vol. 84, pp. 405–414.

Parallelization of the Rosen-Suzuki Fist Function and Himmelblau Function Using a Two-Population Evolutionary Algorithm

Michael Scherger
Department of Computer Science
Texas Christian University
Fort Worth, TX, 76129, USA
Email: m.scherger@tcu.edu

Roja Ramani Molupoju
Department of Computing Science
Texas A&M University – Corpus Christi
Corpus Christi, TX, 78412, USA
Email: molupoju.roja@gmail.com

Abstract – *(ICAI'15) The general nonlinear programming (NLP) problem requires finding the optimum point (minimum or maximum) of a function of n real variables subjected to some given constraints. This research presents the parallel implementation of the Rosen-Suzuki Frst function and the Himmelblau function using a two-population genetic algorithm. There are numerous situations in science and engineering where the optimum is bounded which adds complexity to the optimization problem. The approach used in this research by [32] uses evolves two populations (male and female). One population is evolved inside the feasible domain of the design space and the second population is evolved outside this feasible domain. In this research a version of this parallel, two-population algorithm was implemented. The results of this research show that the two-population parallel genetic algorithm is effective and accurate for these functions.*

Keywords: Evolutionary algorithms, constrained optimization problems, parallel algorithms.

1 Introduction

A constrained optimization problem is usually written as a nonlinear programming (NLP) problem of the following type. The problem seeks to minimize:

$F(x_1, \ldots, x_i, \ldots, x_n)$ $(1 \le i \le n)$

subject to side constraints:

$x_{i\ min} \le x_i \le x_{i\ max}$ $(1 \le i \le n)$

and inequality and equality constraints:
$g_j(x_1, \ldots, x_n) \le 0$ where $(1 \le j \le m)$
$h_j(x_1, \ldots, x_n) = 0$ where $(1 \le j \le p)$.

For these types of NLP problems, there are n variables, m inequality constraints, and p equality constraints. The function $F(X)$ is called the objective function, $g_j(X)$ is the j^{th} inequality constraint, and $h_k(X)$ is the k^{th} equality constraint. The i^{th} variable can varies in range from $[x_{imin} .. x_{imax}]$.

Searching the boundary between the feasible and the infeasible regions is critical for any function to find the global optimum.[30] The inability of evolutionary systems to search precisely the boundary area is the main reason for difficulty in locating the global optimum. Some of the constraints are active at the global optimum.

For this reason, in may constrained optimization problems, it is more difficult to locate the global optimum [23]. A two-population evolutionary computation approach was proposed in paper [31] where two populations of individuals are evolved: one of feasible and the other of infeasible by which the search pressure upon the boundaries of the feasible space can be increased. Feasible individuals (also called females) are the ones that are evolved inside (including the boundaries of the feasible space), while the infeasible individuals (also called males) are evolved outside the feasible space. These two populations are subject to mutation and asexual crossover. Female-male crossover ensures the search pressure on the boundaries of the feasible space. This crossover can be performed between two or more feasible and infeasible individuals, and with, or without, favoring the better-fit females or the better-ranked males.

When two populations, one of feasible and one of infeasible individuals, are distinctively evolved, these two populations interact systematically (rather than occasionally) for the purpose of exploring the boundaries of the feasible space. Since the criterion based on which individuals are assigned to the two populations does not change during evolution, the behavior of the two populations is easier to understand.

The two-population evolutionary algorithm described in this paper, was proposed by [31] and was also implemented with two other constrained optimization problems in [30]. This algorithm was initially tested for functions with two variables; this project works with the algorithm for functions with multiple variables and is tested. This algorithm is implemented using C++ and MPI on a small parallel computing cluster. The program was tested for two benchmark problems from literature:

the Rosen-Suzuki function and the Himmelblau function. Evaluation of the effectiveness of this technique is provided in sections 5 and 6.

2 Related Research

Genetic algorithms (GAs), developed by Holland (1975), have traditionally used a more domain independent representation, namely, bit-strings. However, many recent applications of GAs have focused on other representations, such as graphs (neural networks), Lisp expressions, ordered lists, and real-valued vectors.

GAs are highly abstract computational models inspired by natural selection [14]. Standard GAs apply an *a priori* defined fitness function (e.g. the function one wants to optimize) to an individual. They typically use an all-at-once calculation where individuals are evaluated immediately after their creation (i.e. birth). Fitness calculation in nature is substantially different [1]. It consists of a continuous series of tests during an individual's life originating from a complex environment. This environment is not only influenced by the animal's own actions but also by the other individuals as well as other processes occurring in the world (e.g. climatologically or geophysical changes). Summarizing one can say that - in contrast with GAs - nature uses a far more partial but continuous fitness evaluation in order to adapt to a complex world.

One of the key problems for using GAs in practical applications is to design the fitness function, particularly when we do not know where the global optimum is located [19] [22] [33]. A comparative estimate of how good as a solution turns out to be enough in most cases. However, for constrained problems, determining a way to estimate how close in an infeasible solution from the feasible region is difficult since most real-world problems have complex linear and non-linear constraints, and several approaches have been proposed in the past to handle them [14][33]. From those, the penalty function seems to be yet the most popular technique for engineering problems, but the intrinsic difficulties to define good penalty values makes harder the optimization process using a GA [6] [7] [19].

3 Two Population Parallelization Technique

Evolutionary algorithms are characterized by their repeated fitness evaluation of the individuals in the population. Therefore, it is natural to view them as parallel algorithms. In generational evolutionary algorithms, substantial savings in elapsed time can often be obtained by performing fitness evaluations in parallel. In the simplest form of parallelism, a manager process performs all the function of the evolutionary algorithm except evaluation of individuals, which are performed in parallel by worker processes operating on separate processors. The master process waits for all workers to return the evaluated individuals before varying on with the next generation.

The algorithm in this research is implemented using MPI in which there are *n* processes that work simultaneously using inter-process communication. This performs well when a complex function is being used and there is more number of iterations that are to be performed.

A "Manager-Worker" paradigm is used to implement the algorithm. In this paradigm, there is one Manager process and several workers processes and is depicted in Figure 1. The Manager generates uniform random points and sends these points to the workers; the workers process these points (check the feasibility and compute the value of the function at that point if it is a feasible point) and send back the processed results. The manager would then differentiate them into female and male based on their feasibility value (female if it is zero and male if not). Then the manager ranks the female individuals based on their fitness and the male individuals based on the number of constraints they violate. Generations of new individuals and mutating female and male individuals would also happen. The female-male pairs are formed with the females choosing their males in the rank decreasing order. Then the crossover is performed and all the steps are until the best female is constant for a given number of individuals.

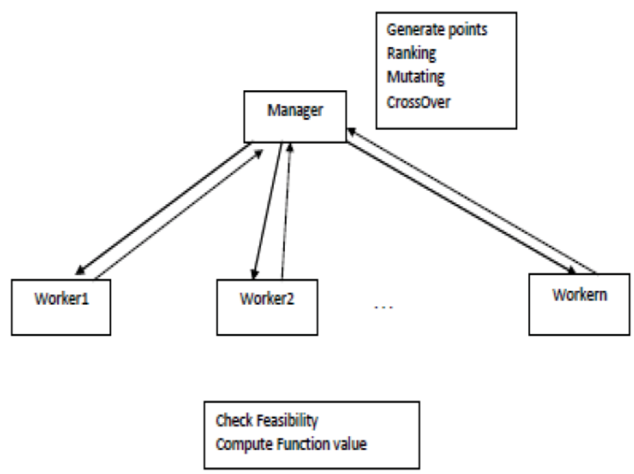


Figure 1: Manager-Worker paradigm for this research.

4 Test Functions

In this project, a technique based on the concept of co-evolution is used to create two populations that interact with each other in such a way that the objective function is minimized. The approach has been tested with two single-objective optimization problems with linear and non-linear inequality constraints and its results are compared with those produced by other GA-based and mathematical programming approaches.

This research is tested using two benchmark problems from literature: the Rosen-Suzuki function and the Himmelblau function. This section provides the description of these two constrained optimization problems.

4.1 Rosen –Suzuki Fist Function (in four variables)

The Rosen-Suzuki Fist problem is a function in four variables with three non-linear constraints on the variables. Hock and Schittkowski introduced it in 1981. The object function is

$$f(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4$$

The nonlinear constraints are:

$$0 \leq 8 - x_1^2 - x_2^2 - x_3^2 - x_4^2 - x_1 + x_2 - x_3 + x_4$$

$$0 \leq 10 - x_1^2 - 2x_2^2 - x_3^2 - 2x_4^2 + x_1 + x_4$$

$$0 \leq 5 - 2x_1^2 - x_2^2 - x_3^2 - 2x_1 + x_2 + x_4$$

The theoretical value for the optimum minimum for this function is -44 and is located at the point (0, 1, 2, -1). The best value that is obtained using this research is -43.9756 at the point (-0.00021, 0.99821, 2.00281, -0.98912). The best value reported in literature is (-0.0005463221, 1.000618, 2.000213, -0.9996195) at which the optimum value for the function is -44 using the GaNOP system[REF].

4.2 Himmelblau Function (in five variables)

Himmelblau originally proposed the Himmelblau nonlinear optimization problem in five variables in 1972. This function was selected since it was used as a benchmark for several other GA-based techniques. The Himmelblau function has five design variables (x1, x2, x3, x4, x5), six nonlinear inequality constraints, and ten boundary conditions. The problem is mathematically stated as follows:

Minimize:

$$f(X) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.29329x_1 - 40792.141$$

Subject to:

$$g_1(X) = 85.334407 + 0.0056858x_2x_5 + 0.00026x_1x_4 - 0.0022053x_3x_5$$

$$g_1(X) = 80.51249 + 0,.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2$$

$$g_1(X) = 9.300961 + 0.0071317x_2x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4$$

with:

$$0 \leq g_1(X) \leq 92$$
$$90 \leq g_2(X) \leq 110$$
$$20 \leq g_3(X) \leq 25$$
$$78 \leq x_1 \leq 102$$
$$33 \leq x_2 \leq 45$$
$$27 \leq x_3 \leq 45$$
$$27 \leq x_4 \leq 45$$
$$27 \leq x_5 \leq 45$$

5 Summary of Results

The testing and analysis of the results for the Rosen-Suzuki Fist and HimmelBlau functions is presented in this section. The parallel program for each of these functions was run in several configurations varying the number of processes and varying the female and male population sizes.

5.1 Rosen-Suzuki Fist Function

The theoretical value for the optimum minimum for this function is -44 and is located at the point (0, 1, 2, -1). The best value that is obtained using this research is -43.9756 at the point (-0.00021, 0.99821, 2.00281, -0.98912).

Following is a table (table 1) in which the average value for 50 runs with a particular number of processes is performed. The table shows the number of processes (in other words, no. of workers), the female and male population, the number of iterations (for which the best female would be constant), the total time for the run (algorithm time plus function value computation time), the number of function calls and the optimum point for the function. The table shows it only for up to 8 processes, the experiment is performed for up to 20 processes, however, the graph below is the one that is drawn using the number of processes and the total required for the completion of the algorithm.

No.of processes	No. of female points	No. of male points	No. of iterations	Function Value	Total Time	No. of function calls	Optimum Point
1	20	15	1000	-43.0923	11.37833	11571	(-0.04239,0.89619, 1.9886,-0.955717)
2	20	15	1000	-43.9503	9.514633	11578	(-0.000138,0.97898, 2.006346,-0.986454)
3	20	15	1000	-43.9362	9.362131	11598	(-0.002435,0.97898, 2.002316,-0.996354)
4	20	15	1000	-43.7521	8.182058	11611	(-0.01134,0.87698, 2.02318,-0.97867)
5	20	15	1000	-43.2259	9.039228	11620	(-0.00054,0.984798, 1.95489,-0.973141)
6	20	15	1000	-42.6588	8.117775	11635	(-0.00187,0.96715, 1.98712,-0.98715)
7	20	15	1000	-43.5648	9.82167	11635	(-0.00583,0.984923, 2.0182,-0.88342)
8	20	15	1000	-43.872	10.11201	14321	(-0.00034,0.98672, 2.00178,-0.97821)

Table 1: Rosen-Suzuki Fist function results.

Figure 2 shows the graph of time vs. number of processes for algorithm computed at the manager process. Figure 2 shows the graph for the number of processes vs. the total time (time for algorithm plus the function value computation time at the worker processes). From the graph, it can be seen that as the number of processes increases, the time decreases. The graph is not a decreasing linearly, since the total time is dependent on the function computation time calculated at the worker processes (also depends on the number of function calls). Figure 3 clearly shows all the computation times in a single graph that shows that the time for algorithm is increased whenever there is an increase in the function value computation time (which depends on the number of function calls).

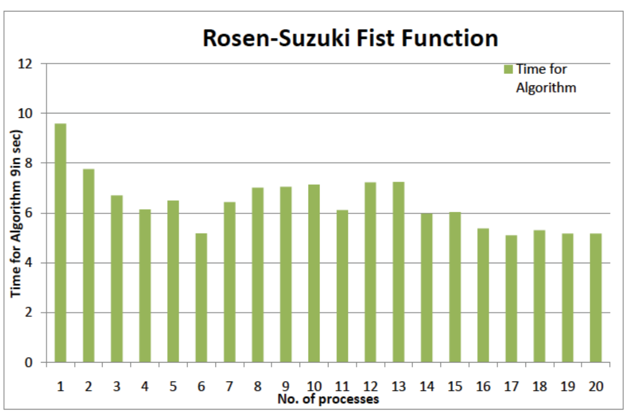


Figure 2: Time vs. number of processes for the Rosen-Suzuki Fist function.

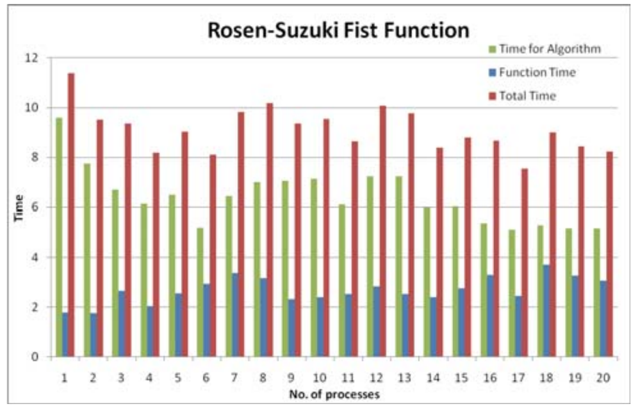


Figure 3: Algorithm, function, and total time vs. number of processes for the Rosen-Suzuki Fist function.

5.2 Himmelblau Function

This problem was originally proposed by Himmelblau and solved using the Generalized Reduced Gradient method (GRG). Gen and Cheng solved this problem using a genetic algorithm based on both local and global reference. The result shown in Table 4 is the best found with their approach.

The mean for the 100 runs performed is $f(X) = -30786.5$. The worst solution found is $f(X) = -30692.9$, which is better than the best solution previously reported. The best solution found with this algorithm is $f(X) = -30868.9$ (corresponding to $x1 = 78.7032$, $x2 = 33.7124$, $x3 = 27.83889$, $x4 = 44.2504$ and $x5 = 43.1159$).

There are 20 female and 15 male points generated every iteration. This was run using 8 processing units in which the best solution remained constant for 1000 iterations. The duration for the run is 0.419021 sec. The number of function calls is 59585. Table 2 summarizes the results of the two-population algorithm used in this research with other algorithms.

Design Variables	This algorithm	Gen[11]	Homaifar[14]	GRG[13]
x_1	78.7032	81.4900	78.0000	78.6200
x_2	33.7124	34.0900	33.00000	33.4400
x_3	27.8388	31.2400	29.9950	31.0700
x_4	44.2504	42.2000	45.0000	44.1800
x_5	43.1159	34.3700	36.7760	35.2200
$g_1(X)$	91.8574	90.522543	90.714681	90.520761
$g_2(X)$	100.517	99.318806	98.840511	98.892933
$g_3(X)$	20.0456	20.060410	19.999935	20.131578
$f(X)$	-30868.9	-30183.576	-30665.609	-30373.949

Table 2: Comparison of current algorithm with previous techniques for the Himmelblau function.

6 Conclusions and Future Work

Solving a constrained nonlinear programming problems using a co-evolutionary algorithm was implemented. The two-population evolutionary algorithm proposed by [31] was used in which two distinct populations are evolved; feasible (females) and infeasible (males). The interaction between these populations concentrate on he boundaries of the feasible space. Mutating the female and male populations induced search capabilities inside the feasible space and parallel to the feasible-infeasible boundary. Figure 4 and Table 2 presents a table of a summary of the results in this research.

Optimization Problem	Theoretical Value	This Algorithm	Best Solution	Constraint Values
Rosen-Suzuki Fist	Value: -44 Point: (0, 1, 2, -1)	Average: -43.5589 (-0.01342, 0.93123, 1.98791, -0.99127)	Value: -42.9756 Point: (-0.00021, 0.99821, 2.00281, -98912)	G1 = -0.02046 G2 = -1.04986 G3 = -0.00185

Figure 4: Summary of results for this research.

There is no penalty factor that is involved in constraint handling; hence there is a clear distinction can be made between the feasible and infeasible individuals. Parallel Implementation of the Co-evolutionary Genetic Algorithm has been a long awaited development for the evolutionary algorithms as mentioned in [7] [19] [31]. In this project, the algorithm has been implemented using MPI, which uses multiple processes that run in parallel, by which the computation time is reduced. This implementation worked well with several test problems that were previously solved using GA-based and mathematical programming techniques, producing in most of the cases results better than those previously reported. The technique is able to achieve such good results with relatively small populations and using a relatively low number of generations.

Functions of multiple variables used in various disciplines of Engineering can be solved for the global optimum using this research. This research provides sound results in terms of the computation time and the number of generations that need to be evolved for a more accurate value. The project is tested using the two benchmark problems from the literature. A detailed comparison of the results obtained from the project with that of those from the literature is presented.

7 Bibliography

[1] Adenike A. Adewuya, “New Methods in Genetic Search with Real-Values Chromosomes”, Thesis, Mississippi State University, 1993.

[2] T. Back, D. Fogel and Z. Michalewicz, Handbook of Evolutionary Computation, The Institute of Physics Publishing, 2000.

[3] Benacer R and Pham Dinh Tao, “Global Maximization of a Non-definite Quadratic Function over a Convex Polyhedron”, Fermat Days 85: Mathematics for Optimization, Amsterdam, Holland, 1986, pp. 65-76.

[4] Carlos R. Garcia-Alonso, Leonor M. Pérez-Naranjo, Juan C. Fernandez-Caballero, “Multiobjective evolutionary algorithms to identify highly auto correlated areas: the case of spatial distribution in financially compromised farms” *Annals of Operations Research*, Jan 2011.

[5] Carlos A. Coello Coello, “Theoretical and Numerical Constraint-Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art,” Computer Methods in Applied Mechanics and Engineering, Vol. 191, No. 11-12, pp. 1245-1287, 2002.

[6] Carlos A. Coello Coello, “Constraint-handling using an evolutionary multiobjective optimization technique”, *Civil Engineering and Environmental Systems*, 17:319-346, 2000.

[7] Carlos A. Coello Coello, “Use of a Self-Adaptive Penalty Approach for Engineering Optimization Problems”, May 1999.

[8] Carlos A. Coello Coello, “An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Design”, PhD thesis, Department of Computer Science, Tulane University, New Orleans, LA, Apr 1996.

[9] Pei-Chann Chang and Shih-Hshin Chen. The development of a sub-population genetic algorithm II (SPGA II) for multiobjective combinatorial problems, *Applied Soft Computing*, Vol. 9, No. 1, pp. 173-181, January 2009.

[10] C. Y. Cheong, K. C. Tan and B. Veeravalli. A multi-objective evolutionary algorithm for examination timetabling, *Journal of Scheduling*, Vol. 12, No. 3, pp. 121--145, April 2009 .

[11] David W. Coit, Alice E. Smith, David M. Tate, “Adaptive Penalty Methods for Genetic Optimization of Constrained Combinatorial Problems”, *Informs Journal On Computing*, Vol. 8, No. 2, Spring 1996, pp. 173-182, DOI: 10.1287/ijoc.8.2.173.

[12] A.E. Eiben, “Multiparent Recombination in Evolutionary Computing,” in A.Ghosh and S. Tsutsui, editors, Advances in Evolutionary

Computing, Natural Computing Series, Springer, pp. 175-192, 2002.

[13] Eshelman Larry J & Schaffer David J, “Real-coded Genetic Algorithms and Interval-Schemata”, In Whitley, D. L. (Ed.), Foundations of Genetic Algorithms 2 (pp. 187-202). California: Morgan Kaufmann, 1993.

[14] Mistuo Gen and Runwei Cheng, “Genetic Algorithms and Engineering Design”, John Weily & Sons, Inc, New York, 1997.

[15] Grefenstette J. J., “Predictive models using fitness distributions of genetic operators”, In *Foundations of Genetic Algorithms 3*, D. Whitley (Ed.), San Mateo, CA: Morgan Kaufmann.

[16] David M Himmelblau, “Applied Nonlinear Programming”, McGraw Hill, New York, 1972.

[17] Homaifar A, Lai S H Y and Qi X, “Constrained Optimization via Genetic Algorithms, Simulation”, 62(4):242-254, 1994.

[18] Antony Iorio, Xiaodong Li, “Parameter Control within a Co-operative Co-evolutionary Genetic Algorithm”, May 2002.

[19] Kenneth De Jong, William Spears, “On the State of Evolutionary Computation”, September, 1993.

[20] [Kenneth Dee Jong A., “Are genetic algorithms function optimizers?” *Proceedings of the Second International Conference on Parallel Problem Solving from Nature*, 1992.

[21] Kenneth De Jong, William Spears, Thomas Back, David Fogel, Hugo de Garis, “An Overview of Evolutionary Computation”, April, 1993.

[22] Kim J. H and Myung H, “Evolutionary programming techniques for constrained optimization problems,” *IEEE Trans. Evolutionary Computation*, vol. 1, no. 2, pp. 129–140, 1997.

[23] Michalewicz Z and Attia N, “Evolutionary Optimization of Constrained Problems”, In Proceedings of the 3rd Annual Conference on Evolutionary Programming, pages 98-108, World Scientific, 1994.

[24] Michalewicz Z and Schoenauer M, “Evolutionary Algorithms for Constrained Parameter Optimization Problems,” Evolutionary Computation, Vol. 4, No. 1, pp. 1-32, 1996.

[25] Papalambros P and Li H L, “Notes on the Operational utility of Monotonicity in Optimization”, ASME Journal of Mechanisms, Transmissions and Automation in Design, Vol.105, 1983, pp. 174-180.

[26] Papalambros P and Wilde D J, “Principles of Optimal Design – Modeling and Computation”, Cambridge Univ. Press, New York, 2000.

[27] Paredis Jan, “Coevolutionary Algorithms”, *Proceedings of the Seventh International Conference on Genetic Algorithms* ed TBaeck (San Mateo, CA: Morgan Kaufmann) pp 393-399, 1997.

[28] Ragsdell K M and Phillips D T, “Optimal Design of a class of Welded Structures Using Geometric Programming”, ASME Journal of Engineering for Industry, Vol. 98, pp. 1021-1025, 1997.

[29] Rosenbrock, H. H., "An automatic method for finding the greatest or least value of a function", *The Computer Journal* 3: 175–184, doi:10.1093/comjnl/3.3.175, ISSN 0010-4620, MR0136042.

[30] Scherger M.C., Molupoju R. R., “The Parallelization of Two Constrianed Optimization Problems Using Evolutionary Algorithms”, Genetic and Evolutionary Methods, WorldComp, July, 2014.

[31] Simionescu P.A., Dozier G.V. and Wainwright R.L. “A Two-Population Evolutionary Algorithm for Constrained Optimization Problems," IEEE World Congress on Computational Intelligence, Vancouver, Canada, July 16-21, 2006.

[32] Simionescu P.A., Beale D.G. and Dozier G.V. "Constrained Optimization Problem Solving Using Estimation of Distribution Algorithms," IEEE World Congress on Evolutionary Computation, Portland, OR, June 20-23, 2004.

[33] Min-Jea Tahk and Byung-Chan Sun, “Co evolutionary Augmented Lagrangian Methods for Constrained Optimization”, IEEE Transactions On Evolutionary Computation, Vol. 4, No. 2, July 2000.

[34] Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, Qingfu Zhang, “ Multiobjective evolutionary algorithms: A survey of the state-of-the-art,” *Swarm and and Evolutionary Computation*”, March 2011.

Mining simultaneously emerging and decaying patterns from temporal quantitative data using genetic algorithm

Anelisa P. da Silva, Dalton M. Tavares, Marcos A. Batista, Sérgio F. da Silva
Institute of Biotecnology
Federal University of Goiás
Catalão-GO 30332–0250
Emails: anelisa.silva25@gmail.com,dmatsuo@gmail.com,marcos.batista@pq.cnpq.br,sergio@ufg.br

***Abstract*—Several areas of knowledge produce quantitative temporal data, which demand the development techniques to identify patterns of them. Identification of emerging and decaying patterns are important in many applications because they can indicate trends that require decision-making or interventional measures. However, the literature have few researches about these kinds of patterns. This article proposes an approach for mining emerging and decaying patterns from quantitative temporal data sets using a genetic algorithm. Rules representing implications of temporal episodes are encoded into chromosomes of the genetic algorithm and these chromosomes are evolved by genetic operators. The quality of each chromosome is evaluated based on how exactly the occurrence frequency of a rule fits a straight line induced by linear regression. The decision whether the pattern is either emerging, decaying, or have no trend is taken based on the straight line slope and the regression coefficient. To prevent that the genetic algorithm converges around a single solution was used a diversity preserving method. Experiments with three quantitative temporal databases show that the results are promising, allowing to mine various emerging and decaying patterns in a single execution of the method.**

I. Introduction

Currently there are numerous sources of quantitative temporal data such as: economy, finance, communications, environment sciences, agronomy, meteorology and agrometeorology. Many of these data sources are the result of great efforts involving design of complex experiments [1], [2] and employment of technologies for data collection and data storage [3], [4], [5], [6]. However, such efforts will be useful only if there be tools or domain specialists to transform data into concrete knowledge. As the capacity of data collection and data storage are far beyond the human capacity for data analysis, and methods of statistical analysis and graphics have limited competence to analyse simultaneous three or more variables, the great hope to transform historical data in concrete knowledge falls on data mining methods.

Data mining methods are promising to convert quantitative temporal data into concrete knowledge aimed at decision making, forecasting and investigation. In recent years, various temporal data mining methods have been proposed [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18]. However, the existing methods in the literature are not effectively applicable to mine rules (implications) from quantitative temporal data and they do not identify emerging and decaying patterns dictated by the frequency of rules occurrence.

We argue that emerging and decaying patterns in the shape of quantitative temporal rules are of great importance to make predictions. For instance, a rule such as:

$$\langle\langle \text{Max. Temp. } (^oC\ ) = [32, 40], \text{ month} = [\text{Nov, Dec}]\rangle$$

$$\Rightarrow$$

$$\langle \text{Max. Rainfall/Day (mm)} = [100, 200], \text{ month} = [\text{Dec, Jan}]\rangle$$

is emerging;

provide useful knowledge to predict that maximum temperature between 32-40^oC implicating maximum rainfall/day between 100-200mm will be more frequent in a near future than in the past. One such prediction can lead to decision-making or interventional measures to avoid catastrophes by strong rainfall. The identification of this kind of pattern can be very useful in several areas of knowledge, as those listed previouslly. However, the algorithms from the literature can not mine such patterns.

In this study we define emerging pattern as a rule whose occurrence frequency increase in the time. On the contrary, we define decaying pattern as a rule whose occurrence frequency decrease in the time. To mine the quantitative temporal association rules we use a real-coded genetic algorithm. Real-coded genetic algorithms are of increasing interest in quantitative association rule mining. However, the main problem of genetic algorithms to mine association rules, which is its nature for unimodal optimization, has been overlooked in several researches. To amenize this limitation, preventing that the genetic algorithm search converges around a single solution, we use a diversity preserving method. Also, it is worth mention that the proposed method does not perform previous variable discretization. It handles numeric data during the whole rule mining process, in contrast with many other approaches that require previously data discretization to discover rules.

We have experimented the proposed method on three real-world data sets. Our results show that it can mine several emerging and decaying pattern described by quantitative temporal rules in a single execution of the method. Additionally, the method does not require the specification of critical parameters such as limiar of support and/or confidence of the rules as required by traditional algorithms from the literature.

The remainder of the paper is organised as follows: Section II presents core definitions regarding the rule of interest.

Section III presents a genetic algorithm to mine emerging and decaying patterns described by quantitative temporal association rules. The results of applying the proposed algorithm to three data sets are reported and discussed in Section IV. Finally, section V summarizes the conclusions.

II. Background

We extend the notion of temporal association rules mining presented in [19] to effectively deal with quantitative temporal data. The concepts presented in [19], which form the theoretical basis of temporal association rules mining, deal only with binary (or discrete) variables, eg., `rain/no_rain` in a specific time instant. This type of representation is obviously not representative for quantitative temporal data mining because the intensity of each event is usually of great importance. Thus, we propose the concept of **episodeset** as an extension of the concept of **itemset** for the domain of quantitative temporal data.

Definition 1. *Episodeset*. *Let* $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m\}$ *be the set of observed variables. Let the interval between observations be called the granularity of time (*τ*), which may be daily, weekly, monthly, among others. Let* $E = \{e_1, e_2, \ldots, e_m\}$ *be a set of episodes recorded in each time period* t*, where* e_i *is an episode associated with the variable* v_i*. A* **episodeset** $E^{(t)}, 1 \le t \le n$ *(where* n *is the number of time periods of the temporal data set), registered in the data set* $\mathcal{D}$ *is called a* **super-episodeset** *or* m-**episodeset** *because it is assumed that all variables have their values recorded*[1]*. Thus, a data set* $\mathcal{D}$ *is a set of* n *super-episodesets* E*, over* n *time periods.*

Definition 2. Base cycle. *A base cycle* Φ_i *is a contiguous sequence of super-episodes from the original database. The length of a base cycle* l *is given by the number of super-episodes composing the base cycle, being user defined. Mathematically, each base cycle corresponds to the time interval* $[t_{i.l}, t_{(i+1).}l)$*. The number of base cycles* n_{BC} *is given by* $\lfloor |\mathcal{D}|/l \rfloor$*, where* $|\mathcal{D}|$ *is the number of m-episodesets in the database.*

Definition 3. Quantitative temporal association rule. *A quantitative temporal association rule is an implication of the form* $\mathbf{X} \Rightarrow \mathbf{Y}$ *(if* $\mathbf{X}$ *then* $\mathbf{Y}$*), where* $\mathbf{X}$ *and* $\mathbf{Y}$ *are conjunctions of episodes associated with observed variables. Consider that* $\mathbf{V_X}$ *denotes the subset of variables associated with the episodic conditions* $\mathbf{X}$ *and* $\mathbf{V_Y}$ *denotes the subset of variables associated with the episodic conditions* $\mathbf{Y}$*. The intersection of* $\mathbf{V_X}$ *and* $\mathbf{V_Y}$ *should be empty, ie.,* $\mathbf{V_X} \cap \mathbf{V_Y} = \emptyset$*.*

Definition 4. Episodic condition. *An episodic condition is a condition of an variable interval in a given time interval. Episodic conditions are represented in the form:* $\langle v_i(\text{unit of } v_i) = [v_{0i}, v_{1i}]$ *in the period (time unit)* $= [t_0, t_1]\rangle$*.*

A quantitative temporal association rule composed by the conjunction of two episodic conditions in the antecedent and one episodic conditions in the consequent is given following:

$\langle\langle$Acc. Rainfall (mm) $= [150-220]$, month = [Nov-Dec]$\rangle$AND

$\langle$Avg. Temperature(^{o}C) $= [25-32]$, month = [Nov-Dec]$\rangle$

$$\Rightarrow$$

$\langle$PlantGrowing (cm) $= [40-50]$, month = Dec-Jan$\rangle\rangle$

Definition 5. A rule happen in a given base cycle. *A given rule* $\mathbf{r}$ *happen in the super-episodesets of a base cycle* Φ_i*, denoted by* $\mathbf{E}[i]$*, iff all the episodic conditions of* $\mathbf{r}$ *occur in* $\mathbf{E}[i]$*. Matematically, we can consider a function as:*

$$happen(\mathbf{r}, \mathbf{E}[i]) = \begin{cases} 1, & \textit{if } \mathbf{r} \textit{ occurs in } \mathbf{E}[i] \\ 0, & \textit{otherwise.} \end{cases} \tag{1}$$

In this research the rules are mined by the genetic algorithm described in the following section.

III. Genetic algorithm to mine emerging and decaying patterns

The technique proposed for mining emerging and decaying patterns from quantitative temporal data is based on a specific genetic algorithm. Thus, the technique is described in terms of the steps and operators of the genetic algorithm, given in Algorithm 1.

Algorithm 1 General genetic algorithm outline.

Require: chromosome coding, fitness function, constraints (atributes that go in the rule antecedent / atributes that go in the rule consequent), the maximum interval length by attribute ($piv.(\max \mathbf{v}_i - . \min \mathbf{v}_i)$), the minimum time window ($janTime$)

Ensure: Quantitative temporal association rules.

1: Generate randomly the population of chromosomes ($\mathcal{C}$) according to chromosome coding;
2: Evaluate each chromosome $\mathcal{C}$ of the population according to the fitness function;
3: Apply the niching method;
4: Select the chromosomes by the roulette-whell method to build the matting pool;
5: Apply uniform crossover to pairs of individuals from the mating pool;
6: Apply uniform mutation on the chromosomes generated by crossover;
7: Select the fittest chromosomes between parents and children for the next generation;
8: While the maximum number of generations is not reached, return to step 2 .
9: Return the set of patterns codified by the chromosomes population.

a) Chromosome coding

Each variable from a data set is associated with a gene of chromosome. If we have m observed variables, we have m genes: G_1, G_2, G_3, …, G_M. Each gene G_i is an episode related to variable $\mathbf{v}_i, i = 1 \ldots m$ and is coded as shown in Figure 1. In Figure 1, w is a weight that is compared against to a threshold to determine whether the episodic condition represented by a given gene will or not by part of the rule. *AC* is a flag that indicate whether the episodic condition represented by the gene will be part of the antecedent (if AC

[1] In real world data collection, it is common that some variables have missing or distorted values. These failure can be due to equipment or human. Data mining on these cases typically uses previously, techniques to treat missing and/or distorted values.

= 0) or of the consequent (if AC = 1) of the rule. v_0 and v_1 are the lower and upper bounds of the variable interval. t_0 and t_1 are the lower and upper bounds of the time interval.

w	AC	v_0	v_1	t_0	t_1

Fig. 1. Gene coding.

b) Fitness function

The fitness function estimates how well the occurrence frequency of a rule fits a straight line. To do this calculation we have that the Equation 1 return a binary serie of n_{CB} values. This binary serie is first summarized using a sliding-window mechanism. Basically, it is counted the number of occurrence of the rule in each window. Then, we apply linear regression to infer a straight line that best fits the data. According to the straight line slope and the regression coefficient we name the pattern represented by the rule calcule the fitness of the rule coded by the chromosome. The Algorithm 2 describes the whole process.

Algorithm 2 Fitness function.

Require: $Binvec$ (a binary vector that describes the rule occurrence in each base cycle), W_s (window size), α and β parameters
Ensure: Fitness value *Fitness*($\mathcal{C}$) and, the kind of the pattern $\mathcal{C}$ represented by *Pattern*($\mathcal{C}$);

for $j = \lfloor W_s/2 \rfloor$ **to** $n_{CB} - \lfloor n/2 \rfloor$
  $y[j - \lfloor W_s/2 \rfloor] = \sum_{k=j-\lfloor W_s/2 \rfloor}^{j+\lfloor W_s/2 \rfloor} Binvec[k]$
  $x[j - \lfloor W_s/2 \rfloor] = \lfloor W_s/2 \rfloor$
$s_{xx} = \sum_{i=0}^{n_{BC}-W_s} \left(x_i - \overline{x}\right)^2$;
$s_{yy} = \sum_{i=0}^{n_{BC}-W_s} \left(y_i - \overline{y}\right)^2$;
$s_{xy} = \sum_{i=0}^{n_{BC}-W_s} \left(x_i - \overline{x}\right)\left(y_i - \overline{y}\right)$;
slope = s_{xy}/s_{xx};
$r_c = \frac{s_{xy}}{\sqrt{s_{xx}s_{yy}}}$;
if *slope* $> \alpha$
  Fitness($\mathcal{C}$) $= |r_c|$;
  Pattern($\mathcal{C}$) = "emerging";
else if *slope* $< \beta$
  Fitness($\mathcal{C}$) $= |r_c|$;
  Pattern($\mathcal{C}$) = "decaying";
else
  Fitness($\mathcal{C}$) $= |r_c|$;
  Pattern($\mathcal{C}$) = "random";

c) Genetic operators

Selection for reproduction is performed by roulette-whell method. Pairs of individuals selected for roproduction are combined by uniform crossover: a binary mask of the size of the chromosome coding is shuffle to indicate which parent chromosome provide each gene to first child; the second child is generated by the mask complement. Each chromosome selected to mutation have one of its mutated genes. The mutation can occur in the weight w, in *AC* (when no restrictions are applied on the variables that make up the antecedent and consequent from the rules), or in the lower limits (v_0 and t_0) and upper limits (v_1 and t_1) of the variable and time intervals.

d) Niching

We use a method of niching called clearing, described in [20]. This method needs a distance measure to estimate the individual similarity. Following we describe the method of clearing and proposed distance measure.

The method of clearing corresponds to the concept of niching enunciated by J.H. Holland in 1975: the sharing of resources by a population of individuals having some similarity. However, instead of sharing the available resources, the method of clearing provides all resources of a niche to the best individual of the niche. This allows to perform a multimodal genetic algorithm optimization that search for various local and global optima at the same time. Furthermore, the method of clearing allows the GA reduce the problem of genetic drift when used in conjunction with an appropriate selection operator.

The method of clearing is applied between the fitness evaluation of chromosomes and selection for reproduction. The method makes use of a distance (dissimilarity) measure between chromosomes (whose phenotype, in our case, corresponds to association rules) to determine whether they belong to the same niche (subpopulation or cluster). Each subpopulation will have a dominant chromosome, the one with the highest fitness value. If a chromosome belongs to a subpopulation, then their dissimilarity to the dominant chromosome is smaller than a given threshold σ, called clearing radius. The method of clearing preserves the fitness of the fittest chromosome while decreases to zero the fitness of the other chromosomes of the subpopulation. Thus, it assigns all the resourses of a niche for a single chromosome: the winner.

Also, the method of clearing is generalizable to accept multiple winners chosen from the best individuals of each niche [20]. The capacity of a niche is defined as the maximum number of chromosomes that a niche may support. If the capacity is greater than 1, the population will probably have more than one winner. If the niche capacity is equal to the population size, the effect of clearing disappears and the search behaves like a standard GA search. Niche capacity between 1 and the population size provides intermediate behaviour between the maximum effect of clearing and standard GA search.

The algorithm 3 illustrates the clearing method [20]. Consider the population of chromosomes $\mathcal{C}$ and the number of chromosomes $n_{\mathcal{C}}$ as global variables. Let σ be the clearing radius and κ the niche capacity. The variable *nbWinner* counts the number of winners of the population associated with the current niche. The population of chromosomes $\mathcal{C}$ can be considered a vector $n_{\mathcal{C}}$ chromosomes .

The clearing algorithm (Algorithm 3) uses three functions:

- *FitnessSort* ($\mathcal{C}$): sorts the population of chromosomes in descending of fitness.
- *Fitness*($\mathcal{C}[i]$): returns the fitness of the i-th chromosome of the population $\mathcal{C}$.
- *Dsitance*($f(\mathcal{C}[i])$, $f(\mathcal{C}\ [j])$): returns the distance between phenotypes of two chromosomes;
- $f(\mathcal{C}[i])$: returns the phenotype of chromosome $\mathcal{C}[i]$. In our case, the phenotype is a quantitative temporal association rule.

Algorithm 3 Clearing niching.

Require: σ (clearing radius), κ (niche capacity).
Ensure: Clearing – allocation of resources to a niche to the fittest individuals.

FitnessSort($\mathcal{C}$);
for i = 0 **to** $n_{\mathcal{C}} - 1$
  if *Fitness*($\mathcal{C}[i]$) > 0
    nbWinners = 1;
    for $j = i + 1$ **to** $n_{\mathcal{C}}$ - 1
      if *Fitness*($\mathcal{C}[j]$)$>$ 0 AND *Distance*($f(\mathcal{C}[i])$, $f(\mathcal{C}[j])$) $< \sigma$
        if *nbWinners* $< \kappa$
          nbWinners = *nbWinners* + 1;
        else *Fitness*($\mathcal{C}[j]$) = 0;
          Fitness($\mathcal{C}[j]$) = 0;

e) Distance between association rules

Let be two temporal quantitative association rules:
$\mathbf{r} = (C_A^{(\mathbf{r})}(v'_1)$ AND $\ldots$ AND $C_A^{(\mathbf{r})}(v'_j)) \Rightarrow (C_C^{(\mathbf{r})}(v'_{(j+1)})$ AND $\ldots$ AND $C_C^{(\mathbf{r})}(v'_m))$ and
$\mathbf{s} = (C_A^{(\mathbf{s})}(v''_1)$ AND $\ldots$ AND $C_A^{(\mathbf{s})}(v''_j)) \Rightarrow (C_C^{(\mathbf{s})}(v''_{(j+1)})$ AND $\ldots$ AND $C_C^{(\mathbf{s})}(v''_m))$
where $C_A^{(T)}(.)$ and $C_C^{(T)}(.)$ are episodic conditions from rule *T*. Each episodic condition have the shape:

$$v_i \in [v0_i, v1_i] \text{ in the time interval } [t0_i, t1_i]$$

where v_i is some variable from the data set. For a given rule $\mathbf{r}$, the intersection of the variable composing the episodic conditions of the rule antecedent ($V_A^{(\mathbf{r})}$) and the variables composing the rule consequent($V_C^{(\mathbf{r})}$) must be empty, i.e., $V_A^{(\mathbf{r})} \cap V_C^{(\mathbf{r})} = \emptyset$.

The distance between $\mathbf{r}$ and $\mathbf{s}$, denoted by *Distance*($\mathbf{r}, \mathbf{s}$), is given in Algorithm 4. For each episodic condition in the rule $\mathbf{r}$ antecedent, $C_A^{(\mathbf{r})}$, it is verified if exists some episodic condition in the rule $\mathbf{s}$ antecedent, $C_A^{(\mathbf{s})}$, which is comparable with $C_A^{(\mathbf{r})}$. Two episodic conditions are comparable only if they refer to the same variable. For all two comparable episodic conditions, we calculate the distance between them. Otherwise, we increment the distance counter by 1 (one). Then the distance counter is divided by the number of episodic conditions $nC_A^{(\mathbf{r})}$. The same calculation is done for the consequent episodic conditions. Finally the distance between the two rules is given by $(dist_A + dist_C)/2$.

The distance between two episodic conditions, $DistanceEp(C_1, C_2)$, is given by the algorithm 5.

Algorithm 4 *Distance*($\mathbf{r}, \mathbf{s}$) – Distance between two quantitative temporal association rules.

Require: $\mathbf{r}$, $\mathbf{s}$ (two quantitative temporal association rules).
Ensure: Distance (*dist*) between $\mathbf{r}$ and $\mathbf{s}$.

$nC_A^{(\mathbf{r})}$ = *numCond*($C_A^{(\mathbf{r})}$);
$dist_A = 0$;
for i = 0 **to** $nC_A^{(\mathbf{r})} - 1$
  if $\exists C_A^{(\mathbf{s})}(v_k)$, such that $v_{j_k} == v_{j_i}, v_{j_i} \in V_A^{(\mathbf{r})}$
    $dist_A = dist_A + DistanciaEp(C_A^{(\mathbf{r})}(v_{j_i}), C_A^{(\mathbf{s})}(v_{j_k}))$;
  else
    $dist_A = dist_A + 1$;
$dist_A = dist_A / nC_A^{(\mathbf{r})}$;

$nC_C^{(\mathbf{r})}$ = *numCond*($C_C^{(\mathbf{r})}$);
$dist_C = 0$;
for i = 0 **to** $nC_C^{(\mathbf{r})} - 1$
  if $\exists C_C^{(\mathbf{s})}(v_{j_k})$ auch that $v_{j_k} == v_{j_i}, v_{j_i} \in V_C^{(\mathbf{r})}$
    $dist_C = dist_C + DistanciaEp(C_C^{(\mathbf{r})}(v_{j_i}), C_C^{(\mathbf{s})}(v_{j_k}))$;
  else
    $dist_C = dist_C + 1$;
$dist_C = dist_C / nC_C^{(\mathbf{r})}$
$dist = (dist_A + dist_C)/2$;

Algorithm 5 *DistanceEp*(C_1, C_2) – Distance between two temporal episodes associated to the same variable.

Require: C_1, C_2, v (two episodic conditions associated to the same variable v).
Ensure: Distance between C_1 and C_2, stored in *distEp*.

Calculate:
  $\min v$ /*minimum value that v takes in the data set*/
  $\max v$ /*maximum value that v takes in the data set*/
  $\min t$ /*minimum value that t takes in the data set*/
  $\max t$ /*maximum value that t takes in the data set*/
$dv = \min\{(v_1^{(C_1)} - v_0^{(C_1)}), v_1^{(C_2)} - v_0^{(C_2)})\} - (\min\{v_1^{(C_1)}, v_1^{(C_2)}\} - \max\{v_0^{(C_1)}, v_0^{(C_2)}\})$;
$distV = dv/(\max v - \min v)$;

$dt = \min\{(t_1^{(C_1)} - t_0^{(C_1)}), t_1^{(C_2)} - t_0^{(C_2)})\} - (\min\{t_1^{(C_1)}, t_1^{(C_2)}\} - \max\{t_0^{(C_1)}, t_0^{(C_1)}\})$
$distT = dt/(\max t - \min t)$;
$distEp = (distV + distT)/2$;

IV. Results

In this section were report four case studies, conducted to validate the proposed theory and techniques. All case studies were performed on a Mac Pro, Processor 3.0 GHz Intel, 8-Core Intel Xeon, with 64GB DDR3 1867MHz, using OS X 10.8.3 operating system. The methods were implemented in C language. In the studies of case, the number of mined rules (*numRules*) and runtime (*runtime*) given in seconds, were analyzed to each algorithm configuration. In the technique setting we vary the size of population of chromosomes ($n_{\mathcal{C}}$), the number of generations (*nGEN*), the radius of *clearing*(σ) and the maximum allowable range of variable interval. The maximum allowable range of variable interval is defined by $piv.(\max v - \min v)$, where piv is the interval $[0 - 1]$. In the case studies we report only the value of piv used. The other parameters of GA, fixed in all experiments are crossing rate of 80%, mutation rate of 3% per chromosome and window size ($W_s = 5$). The results report an average of three runs for each algorithm configuration.

Case study 1: Stock Prices

In this case study we use the *Stock Prices* data set (http://www.stat.ucla.edu/cases/) that corresponds to the daily stock prices from January 1988 to October 1991 of ten airline companies of the United States (ten variables, because the stock price of each company is a variable). Figure 2 is the plot stock prices of each company in the time period.

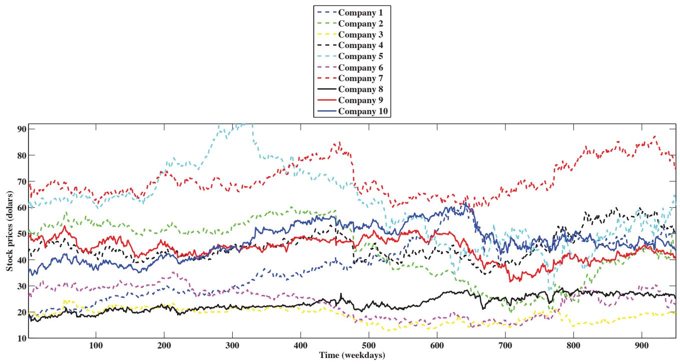


Fig. 2. Stock prices of airline companies from 1988 to 1991.

In this study, we imposed the restriction that the company10 will be part of the consequent of the rule. Table IV shows the the number of rules mined and runtime for each technique configuration. The general conclusions of the experiments are given in subsection named Results Analysis.

	n_C	*nGen*	σ	*piv*	*numRules*	*exec. time (sec.)*
	100	350	0.3	0.1	24.8	2.08
	100	350	0.3	0.2	29	2.63
	100	350	0.5	0.1	20.5	1.73
Stock Prices	100	350	0.5	0.2	33	2.6
	50	250	0.3	0.1	10.5	0.56
	50	250	0.3	0.2	15.5	0.68
	50	250	0.5	0.1	10.8	0.56
	50	250	0.5	0.2	15.5	0.65

TABLE I. GA CONFIGURATION *versus* NUMBER OF MINED RULES (*numRules*) *versus* RUN TIME IN SECONDS FOR THE DATA SET

Case study 2: Piracicaba

In this case study we used the Piracicaba data set, collected by Embrapa – a Brazilian agricultural research corporation. The data set consists of three variables taken monthly: average maximum temperature (*Tmax*), average minimum temperature (*Tmin*), and accumulated rainfall (*Acr*). The data set corresponds to a period of 47 years, from 1961 to 2008. Figure 3 is the plot of the variables collected in the time period. In this study we imposed the restriction that accumulated precipitation (*Acp*) will be part of the consequent of the rule. Table IV summarizes the results obtained.

	n_C	*nGen*	σ	*piv*	*numRules*	*exec. time (sec.)*
	100	350	0.3	0.1	20.8	1.01
	100	350	0.3	0.2	20.6	1.04
	100	350	0.5	0.1	5.2	0.41
Piracicaba	100	350	0.5	0.2	5.6	0.46
	50	250	0.3	0.1	10.2	0.23
	50	250	0.3	0.2	12.6	0.28
	50	250	0.5	0.1	5.4	0.16
	50	250	0.5	0.2	4.4	0.15

TABLE II. GA CONFIGURATION *versus* NUMBER OF MINED RULES (*numRules*) *versus* RUN TIME IN SECONDS FOR THE DATA SET

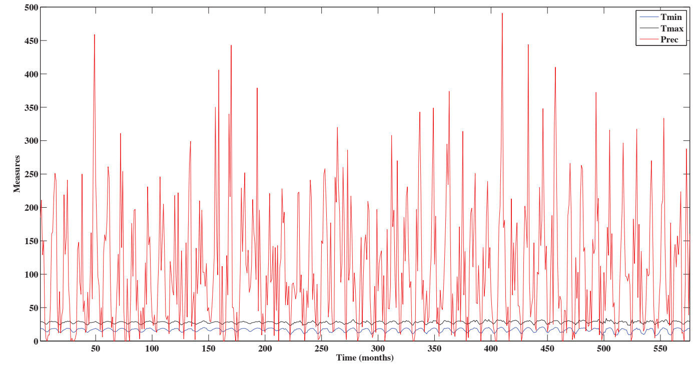


Fig. 3. Monthly meteorological measurements from Piracicaba: jan/1961 to Dec/2008.

Case study 3: Productivity of cane sugar in Piracicaba

In this study of case we use a data set regarding productivity of cane sugar in Piracicaba, provided by Cepagri (Centro de Pesquisas Agrometeorológicas e Climáticas Aplicadas a Agricultura)-UNICAMP. The data set consists of four variable taken monthly in Piracicaba: average of minimum temperature (*Tmin*), average maximum temperature (*Tmax*) average precipitation (*Prec*) and average productivity of cane sugar (*Prod*) in tonnes per hectare (ton/ha). The data set corresponds to the period from 2003 to 2009. Figure 4 shows the ploting of the data set. In this study we imposed the restriction that productivity (*Prod*) will be part of the consequent of the rule. Table IV summarises the results obtained.

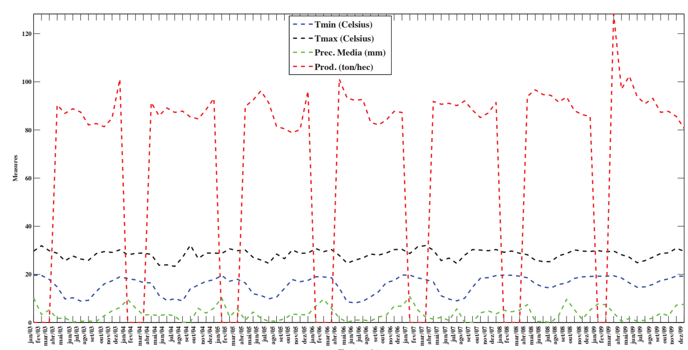


Fig. 4. Agrometeorological measures related to productivity of cane sugar in Piracicaba from 2003 to 2009.

n_C	*nGen*	σ	*piv*	*numRules*	*exec. time (sec.)*
100	350	0.3	0.1	24.6	1.03
100	350	0.3	0.2	28.5	1.34
100	350	0.5	0.1	9.5	0.34
100	350	0.5	0.2	10.7	0.38
50	250	0.3	0.1	13.3	0.20
50	250	0.3	0.2	12.3	0.22
50	250	0.5	0.1	5.3	0.07
50	250	0.5	0.2	7.9	0.13

TABLE III. GA CONFIGURATION *versus* NUMBER OF MINED RULES (*numRules*) *versus* RUNTIME IN SECONDS, FOR THE DATA SET PRODUCTIVITY OF SUGAR CANE IN PIRACICABA.

Results Analysis

We can see that the technique can mine various patterns (emerging and decaying) of high quality (according to the fitness measure, in a single execution of the algorithm by controlling the clearing radius parameter (σ). The size of population (n_C) and number of generations (*nGen*) behaved as expected, ie., the higher these values, the larger is the number of rules mined. The smaller the clearing radius theoretically greater will be the capacity of the environmental to accommodate niches and, therefore, the greater the amount of mined rules. The clearing radius parameter behaved as expected in all studies of case. The parameter *piv* also behaves, in average, as expected: on average, the greater the size of allowable interval, the greater the number of possible rules.

V. Conclusion

Building on existing concepts in the literature of temporal association rules mining, we propose an extension to deal efectively with quantitative temporal data without loss of information. To demonstrate the validity of the proposed concepts, we developed a genetic algorithm that operates according to these concepts in order to identify emerging and decaying patterns. It is worth noting that the proposed genetic algorithm does not require prior data discretization, as well as critical parameters specific from data sets such as the support and confidence thresholds, required by classical association rule mining algorithms. As can be seen, the proposed genetic algorithm has several adjustable parameters, however, the adjustment of these parameters is intuitive, as described in Result Analysis subsection. The limitations of traditional classical genetic algorithms to perform search multimodal is treated by using the classic mechanism of clearing to preserve diversity. Our results show that it is possible to mine several pattern of high quality in a single execution of the method.

It is also worth noting that, as is knowledge of the authors, the type of pattern mined by the proposed approach, and highly informative, and is not identified by any method from the literature. Thus, it is expected that the proposed concepts can be use as core concepts for the development of new methods for quantitative temporal association rule mining. Furthermore, the proposed method can be applied to quantitative temporal databases coming from different fields of knowledge aiming at discover valuable and non-trivial knowledge.

Acknowledgment

The authors would like to thank FAPEG and CNPq for the financial support.

References

[1] J. Antony, *Design of Experiments for Engineers and Scientists*, 1st ed. Elsevier, 2003.

[2] D. C. Montgomery, *Design and Analysis of Experiments*, 8th ed. Wiley, 2012.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–105, 2010.

[4] A. R. Al-Ali, I. Zualkernan, and F. Aloul, "A mobile gprs-sensors array for air pollution monitoring," *IEEE Sensors Journal*, vol. 10, no. 10, pp. 1666–1671, 2010.

[5] L. M. L. Oliveira and J. J. P. C. Rodrigues, "Wireless sensor networks: a survey on environmental monitoring," *Journal of Communications*, vol. 6, no. 2, pp. 143–151, 2011.

[6] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A low power wireless sensor network for gully pot monitoring in urban catchments," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1545–1553, 2012.

[7] S. Amo, N. A. Silva, R. P. Silva, and F. S. Pereira, "Tree pattern mining with tree automata constraints," *Information Systems*, vol. 35, pp. 570–591, 2010.

[8] B. Catania and A. Maddalena, "A unified framework for heterogeneous patterns," *Information Systems*, vol. 37, pp. 460–483, 2012.

[9] D.-A. Chiang, C.-T. Wang, S.-P. Chen, and C.-C. Chen, "The cyclic model analysis on sequential patterns," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 11, pp. 1617–1628, 2009.

[10] J. K. Febrer-Hernández and J. Hernández-Palancar, "Sequential pattern mining algorithms review," *Intelligent Data Analysis*, vol. 16, pp. 451–466, 2012.

[11] T. Fu, "A review on time series data mining," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164–181, 2011.

[12] Y. Hai and X. Li, "A general temporal association rule frequent itemsets mining algorithm," *International Journal of Advancements in Computing Technology*, vol. 3, no. 11, pp. 63–71, 2011.

[13] Y. J. Lee, J. W. Lee, D. J. Chai, B. H. Hwang, and K. H. Ryu, "Mining temporal interval relational rules from temporal data," *Journal of Systems and Software*, vol. 82, no. 1, pp. 155–167, 2009.

[14] N. R. Mabroukeh and C. I. Ezeife, "A taxonomy of sequential pattern mining algorithms," *ACM Computing Surveys*, vol. 43, no. 1, pp. 1–41, 2010.

[15] M. Plantevit, A. Laurent, D. Laurent, M. Teisseire, and Y. W. Choong, "Mining multidimensional and multilevel sequential patterns," *ACM Transactions on Knowledge Discovery from Data*, vol. 4, no. 1, pp. 1–37, 2010.

[16] N. Tatti and B. Cule, "Mining closed strict episodes," *Data Mining and Knowledge Discovery*, vol. 25, no. 1, pp. 34–66, 2011.

[17] X. Yan, C. Zhang, and S. Zhang, "Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support," *Expert Systems with Applications*, vol. 36, pp. 3066–3076, 2009.

[18] M. Zhang and C. He, "Survey on association rules mining algorithms," *Advancing Computing, Communication, Control and Management*, vol. 56, pp. 111–118, 2010.

[19] C.-H. Lee, M.-S. Chen, and C.-R. Lin, "Progressive partition miner: an efficient algorithm for mining general temporal association rules," *IEEE Transaction on Knowledge and Data Engineering*, vol. 15, no. 4, pp. 1004–1017, 2003.

[20] A. Pétrowski, "A clearing procedure as a niching method for genetic algorithms," in *Proceedings of 3rd IEEE International Conference on Evolutionary Computation*, New York, USA, 1996, pp. 798–803.

Two Repair Schemes Applied to a Multiply-Constrained Harvest Scheduling Problem

P. C. Geiger, W. D. Potter

Abstract: Many real world problems are too complex to solve with traditional programming methods in a reasonable amount of time. Stochastic optimization techniques have been applied to this class of problems with success. These stochastic methods present their own problems when applied to problems with infeasible regions. Two methods for repairing invalid solutions are presented here, with a comparison to a penalty-function method, which represents the standard approach.

Keywords: Genetic Algorithm, Harvest Scheduling, Stochastic Methods, Repair

Section 1.0 Introduction:

This paper compares two novel repair methods for optimizing timber harvest schedules. The first method

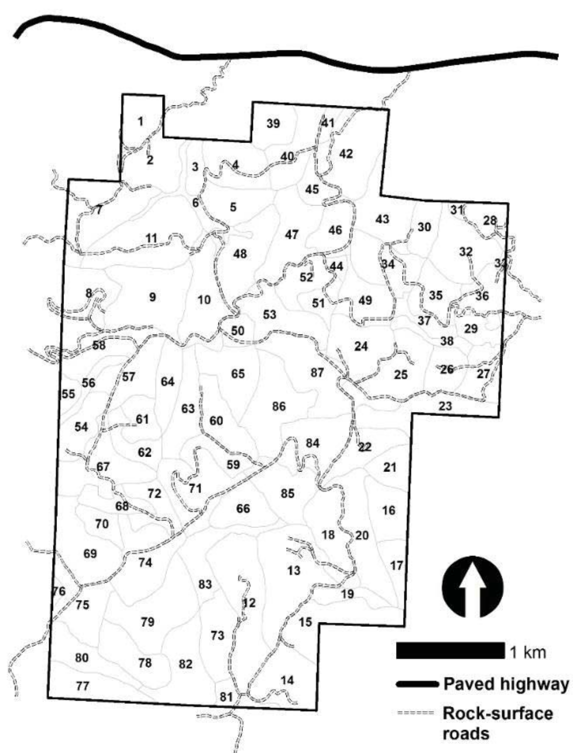


Figure 1: Map of the Lincoln Tract

presented is an implementation of a greedy local search to repair invalid solutions with the aim of improving them. The second method is a stochastic repair method that does not take the value of a solution into account during repair. Both of these methods were used in conjunction with the genetic algorithm, a population based stochastic optimization method. Both methods produced good results, with several important differences in the quality of their solutions.

Section 1 of this paper introduces harvest scheduling and the genetic algorithm. Section 2 explains the repair methods that are being compared, section 3 describes the experimental method, and the results are presented in section 4.

Section 1.1 Harvest Scheduling:

Scheduling the harvest of timber land takes factors of ecology, revenue, and law into account, creating a complex set of rules that each harvest must follow. The problem in its simplest form is to harvest timber at set intervals while producing the same amount of timber each time. This takes growth rates and timber maturity into account.

For this comparison, a real tract of timber land called the Lincoln Tract was chosen. The Lincoln Tract consists of 87 stands. The optimization objectives, and plot yields for the

Lincoln Tract were first described in **Forest Management and Planning** (Bettinger, 2008).

The problem is complicated by legal and ecological constraints. Each stand must be mature, and adjacent stands can only be cut if their total is less than 120 acres. Additionally, only 50% of the forest may be cut during the 30 year harvest schedule.

The fitness function used in all of these experiments can be mathematically expressed this way:

$$Minimize \sum_{i=1}^{n} (H_i - T)^2$$

- i is the harvest period
- n is the number of harvest periods (6)
- H_i is the total harvest in period *i*
- T is the target harvest (13,950)

Both undercutting and overcutting is penalized by this function, reinforcing the goal of even-flow harvesting.

Section 1.2 Optimizing using A Modified Genetic Algorithm:

Finding an optimal solution to this problem is NP-Hard. There are many general optimization that can be applied to this class of problem. Previously, both Genetic Algorithms and Particle Swarm Optimization have shown some promise (Potter, 2009).

Some variation on greedy repair has been used for harvest scheduling as early as 1995 (Liu, 1995). Liu's algorithm was applied to a much simpler harvest scheduling problem with a different set of objectives, but the greedy repair combined with another greedy operator yielded good results. Later, Bäck performed a similar comparison of operators on the set covering problem with constraints and again found that repair operators yielded stronger results than the penalty method for GAs (Bäck, 1996). Bettinger's work with Raindrop Optimization as applied to harvest scheduling has shown promise for repair operators as well (Potter, 2009).

The Genetic Algorithm (GA) is a stochastic, population based meta-heuristic used for optimization developed by Holland in 1975 (Holland, 1975). The GA loosely mimics the process of natural selection. In this case, each schedule is analogous to the genome of an individual organism in a population. Mating is selective, and each generation hopefully produces better offspring. For our problem, each harvest schedule is represented as an array of length 87 with integers 1-6 representing harvest times and zero representing a no-cut scenario.

[1,4,0,2,3,6,0…..]

A problem arises when two chromosomes are crossed: they may produce unviable offspring. In nature, this individual would die, but in a computer simulation, this often represents a waste of good information. Repair and penalty functions are the two most popular methods for helping the genetic algorithm produce valid solutions.

In the repair method, small changes are made to solutions that are violating one or more of the constraints, to eliminate their constraint violation. Usually, some deeper knowledge of the problem must be known to produce good repair. The repair methods described later rely very little on intimate knowledge of the problem, and could therefore be used on any problem where a) valid assignments can be identified independently and b) the impact of those changes can be measured.

Penalty functions put pressure on the algorithm to produce individuals that obey constraints. This is done by adding a penalty value to the fitness of any individual that violates constraints. We have taken the penalty function method as our baseline comparison because like the repair methods presented here, it can be implemented without much domain specific knowledge and it is a popular solution.

Section 2.0 Algorithms

Section 2.1 Stochastic Repair:

The stochastic repair algorithm developed for these experiments is a fairly simple algorithm with two goals: 1. Eliminate bias during repair and 2. Bring the schedule back within the bounds of the hard constraints without creating any new violations. The first goal is achieved by identifying all of the stands currently in violation, then randomizing their fix order. The second goal is achieved by fixing the violation of each of these stands progressively until the entire schedule is back in compliance with the hard constraints. No consideration is given to whether these fixes improve the fitness of the solution beyond making them viable.

Stochastic Repair Algorithm:

While (Schedule Violates C1->Cx)

List<Stand> Violators = Stands in violation of any constraint.

//Stand X is chosen randomly.

Stand X = Violators(Random Stand);

List<TimePeriod> Legal = All Periods that Are Legal for X;

//No cut is added as a choice this also prevents a situation where no assignment can be made, and prevents bias against a no-cut assignment.

Legal.Add(No Cut Assignment);

//Period Y is chosen randomly from among all legal choices for the stand.

TimePeriod Y = Legal.ChooseRandom;

//Stand X is assigned Time Period Y, bringing it, and possibly other stands, out of violation.

StandX.TimePeriod = Y;

The first step of the repair is to identify the stands that are currently causing a violation. Because the algorithm never allows a stand that is too young to be cut, we only have to check for violations caused by stand-groups that exceed our limit for size. If stand A,B,C together violate our aggregate rule, but we remove A from the assignment, it is possible for B and C to then come into compliance with our rule. Therefore, we only fix a single stand at a time, we do not want to disrupt the schedule any more than necessary.

It is important to choose the stand to repair at random, not by its placement in the list of stands in violations. Choosing a stand at random prevents us from biasing our schedules arbitrarily by the order in which we have assigned our stands in a list. The structure of our list should not have arbitrary influence on our outcome. For the same reason, we do not want to bias the algorithm towards choosing a cut period over the no-cut option. Therefore, the no-cut option is always available as an assignment.

Section 2.2 Greedy Stochastic Repair:

The first of these principals – avoiding order bias is preserved in the greedy stochastic repair, but by its nature the greedy algorithm is biased in its selection of assignments. The greedy repair algorithm used for these tests is a simple way to attempt to strategically minimize the outcome of a schedule while also satisfying the hard constraints of the problem. Certainly, the idea of greedy local search with stochastic elements is not new. Algorithms such as GSAT [Selman, 1992] have been studied since the 80s. This algorithm combines greedy local search with a stochastic ordering to help prevent order bias.

Greedy Repair Algorithm:

While (Schedule Violates C1->Cx)

List<Stand> Violators = Stands in violation of any constraint.

//Stand X is chosen randomly.

Stand X = Violators(Random Stand);

List<TimePeriod> Legal = All Periods that Are Legal for X;

//No cut is added as a choice this also prevents a situation where no assignment can be made, and prevents bias against a no-cut assignment.

Legal.Add(No Cut Assignment);

//Period Greedy is stand G where G minimizes the fitness function the most.

TimePeriod G = From Legal Select Stand G Where G Minimizes Objective Function the Greatest;

StandX.TimePeriod = Greedy;

Section 2.3 Proportional Penalty

The proportional penalty algorithm relies on the basic GA framework that all of the solutions share, except instead of repairing solutions that violate one of the constraints, it adds a penalty to the objective function. A penalty function that is too high can prevent a GA from exploring through invalid territory, but a penalty function that is too low will prevent a GA from converging on a legal solution (Deb, 2000).

Section 3 Experimental Design:

After running some initial tests for tuning the Genetic Algorithm's parameters, the following parameter settings were found to yield good results for both the Greedy and Stochastic repair operators. Crossover was set to .9, mutation rate was set to .005, a maximum of 5000 generations was used, elitism, and tournament selection for crossover, with a tournament size of 3. Both algorithms performed better with low mutation rates.

Although the algorithm often converged well before generation 5000, some of the best results in the test sets did not reach convergence until beyond the 4000th generation. The extra generations added time to the experiment, but did not affect the results of the runs where convergence happened earlier. The algorithm also performed poorly with mutation rates over .01 and below .005. Although the algorithm seems sensitive to disruption from mutation, it could not explore enough after convergence to find good answers when the mutation rate was set too low.

The schedule representation was a list representing the 87 stands. Each stand could be assigned an integer 0-6. A 0 represents a no-cut and 1-6 represent the periods in 5 year increments. Each schedule was initialized with a random, but valid assignment. Since each schedule had valid age-requirements to begin with and mutation was also restricted to valid age requirements, the repair algorithms only had to deal with the second hard constraint, which is that no aggregation of cut stands can be over 120 acres.

A baseline performance measure was established through running the algorithm with the penalty scheme described earlier. Several penalty functions were tested, but the proportional scheme was chosen because it faired the best in our test runs. It was run 10 times with population sizes of 1000, 2000, 3000 and 4000 respectively.

Both repair algorithms were then run 10 times each with increasingly large population sizes. It was hypothesized that the greedy algorithm would perform better than the stochastic algorithm with smaller population sizes because it would be more able to find good answers with sub-par starting points. The algorithms were exercised multiple times because Genetic Algorithms and the repair algorithm employed in this experiment are both non-deterministic and it is possible to get very good and very bad answers with the same technique. Running multiple times also yields valuable information about the reliability of results from the algorithms.

Section 4 Results:

The baseline results were generally good, with a population size of 3000 yielding the best overall results, with the exception of a single outlier that can be seen in the figure below.

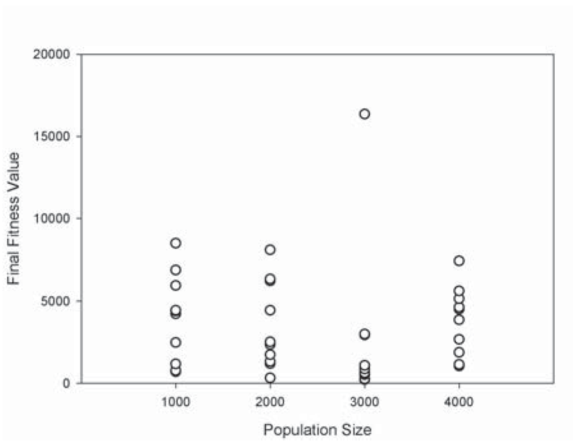


Figure 2: Baseline Results

The clusters of answers were fairly tight with a population size of 3000 again showing the tightest clustering along with the best results for our baseline experiment.

The results of Greedy Repair were far more consistent than the unbiased method. This is likely due to similar optimizations being achieved at early stages that push the algorithm towards a local optima instead of the global best. However, it also means that good, if not great, results can be achieved in fewer executions of the algorithm. The consistency, and good results with smaller populations, could outweigh the loss of best results under some circumstances. The difference between best results and worst results with a population of 4000 is just over 1000, if a single outlier is omitted. This represents a very tight clustering.

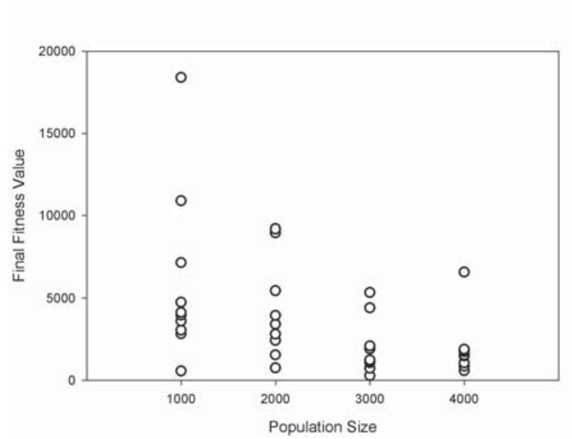


Figure 3: Greedy Repair Results

The results from stochastic repair are promising. The algorithm was able to find very good solutions to the scheduling problem, with the best result being 66. With large population sizes good results were common, but the variation in the results tended to be large. Our best achieved results were with population size of 4000 and the fitness values between best and worst varied by more than 5000, showing that the consistency of the stochastic algorithm was not close to the consistency of the greedy algorithm. However, the algorithm produced the best result recorded during these experiments at a fitness of 66.02, which is very good. The algorithm trades off the chance at great results for consistency between runs.

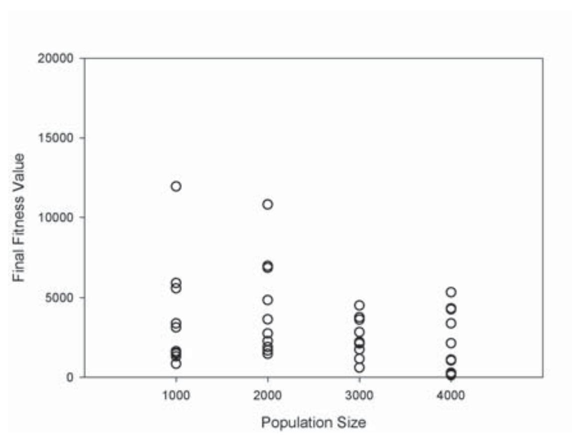


Figure 4: Stochastic Repair Results

Section 5 Conclusion:

Both repair algorithms tested yielded good results on the problem set with sufficiently large populations, but showed different characteristics with respect to consistency and best results observed. Their performance strengths were different, with stochastic yielding better "best" results while the greedy algorithm was more consistent. This pattern holds when compared to the baseline penalty scheme as well.

Metric	Baseline	Greedy	Stochastic
Best	223.811	586.49	132.659
Worst	16354.658	6589.19	5341.923
Average	2645	1910.3	2226.094
Standard Deviation	4671.124	1612.06	1857.502

Table 1: Results Table

The penalty method does not yield results quite as good as the stochastic method, nor is it as consistent as the greedy method. The simple stochastic repair method is preferable on many grounds because it doesn't add much computation time and yields good results. However, if one is constrained by time and cannot run many trials, the greedy method may be better because it consistently gives good results.

In either scenario, a repair method seems to be a better choice over the penalty method for this problem.

Works Cited

T. Bäck, M. Schütz, S. Khuri, "A comparative study of a penalty function, a repair heuristic, and stochastic operators with the set-covering problem," Artificial Evolution, Lecture Notes in Computer Science Voume 1063, 1996, pp. 320-332, 1996.

J. E. Beasley and P. C. Chu, "A genetic algorithm for the set covering problem," European Journal of Operational Research, vol. 94, no. 2, pp. 392–404, 1996.

P. Bettinger, K. Boston, J. Siry, and D. Grebner. Forest Management and Planning, 1st Edition. Elsevier, 2008.

K. Deb. "An efficient constraint handling method for genetic algorithms. Comput. Methods Appl. Mech Engrg. 186 (2000) 311-338 Elsevier.

J. Holland. *Adaptation in Natural and Artificial Systems.* University of Michigan Press, 1975.

G. Liu, "Evolution Programs, Simulated Annealing and Hill Climbing Applied to Harvest Scheduling Problems", Thesis, University of British Columbia, 1995.

W.D. Potter, E. Drucker, P. Bettinger, F. Maier, M. Martin, D. Luper, M. Watkinson, G. Handy, and C. Hayes, "Diagnosis Configuration, Planning and PathFinding: Experiments in Nature-Inspried Optimization," in Natural Intelligence for Scheduling, Planning, and Packing Problems, edited by R. Chiong, Springer-Verlag, Studies in Computational Intelligence (SCI) series, 2009.

D. Orvosh and L. Davis, "Using a genetic algorithm to optimize problems with feasibility constraints," in Proceedings of the 1st IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, pp. 548–553, IEEE, June 1994.

B. Selman, H. Levesque, and D. Mitchell (1992). A new method for solving hard satisfiability problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI'92)*, pages 440–446.

On the Evolutionary Model of Intelligence

I. Weisband
Weisband Informatik, Neuss, Germany

Abstract - *Social evolution is the process that has created and continues to develop not only the society, but human being self. Evolution is almost alone known and comprehensible natural mechanism of intelligence. It would be false do not to check if the same mechanism is used in the human brain. There are some phenomena known but incomprehensible in psychology: Sigmund Freud's concept of unconscious mind (1895), Carl Gustav Jung's concept of transformation (1912) and his psychological functions (1921). In this work we are trying to build the common theoretical model of evolutionary process for both sociological and psychological levels which could explain these phenomena.*

Keywords: evolutionary computation, model, transformation, clan, elite, intelligence

Some AI approaches are simultaneously hypotheses about the mechanism of the natural intelligence. One of such hypotheses is the concept of social evolution founded 1852 (few years before Darwin) by Herbert Spencer [1, 2, 3]. The task of creation of people and human society is certainly an intellectual and creative one. It seems likely that much of this work was done not by the Darwinian mechanism of biological evolution but rather by the Spencer's mechanism of social evolution.

The most important principle of Spencer's sociology is the likening society to an evolving organism. The further step is to assume the identity of the principles of information processing in society and in a single organism (brain, mind). We will use the social evolution theoretical model described in [4] in order to check if such mechanism can be modelled and used in the human brain.

There are few terms from this work we must define and explain in order to make this work independent from the referred one.

1 Clans, Elite, Basis

One of the central terms in our conception is "clan" which is defined as a set of people united by a particular model of the real world they share and with which they attribute their identity. We suppose only the casual groups of people (i.e. – passengers of one bus) do not correspond to this definition. Nation, party, corporation, team, even family have a shared by all members model they bear. Bearing of such model is often the aim and the sense of the clan's existence (confessions, scientific communities or schools). The assumption that groups rather as individuals are units of selection is popular [3].

Clans build a hierarchy – members of clans are most not people, but other clans. We can define the term "rank" of a clan. Let us assign to a clan which members are people (family, team) the rank 1. The super-clan of this clan has the rank 2 – i.e. company can consist of a set of teams – and so on. A man can be considered as a clan of the rank 0.

The hypothesis we discuss in this work is, that the last sentence presents not a degenerate case as it is often in mathematics. Furthermore, we can speak about the negative ranks of clans. The term "clan" corresponds in this case with the popular in AI term "agent".

Clan can take in its super-clan either the position of elite or the position of simple member. Elite – is a clan, claiming one type of command in its super-clan. The set of all clan members, regardless of affiliation to the elites, called a basis. The dichotomy basis/elite can be good illustrated by the Serenity prayer: "God, grant me the serenity to accept the things I cannot change, the courage to change the things I can, and the wisdom to know the difference." [5]

The work [4] presents the hypothesis that social evolution will be provided not so much by the competition of clans for survival, as by the competition of elite for command. Vilfredo Pareto was talking about constant cycling and change of elite; he called history the "graveyard of aristocracies" [6].

The informational sense of this phenomena is based on the assumption, that no clan can adapt to the environment (adaptation is the main term in any evolution theory) if it has no adequate model of this environment. So the competition of elite for command will contribute to creation the more adequate model of the real world if namely the elite bearing the more adequate model of real world will win.

But only the basis can play the role of arbiter between elite because no elite can be objective. It makes democracy the best form of power. Basis can't evaluate the models of elite; it evaluates rather the elite self. The survival and progress of people is caused with following phenomena: basis selects most just the elite with more adequate model. Why? We shall discuss it below.

2 Unconscious Mind

The most important discoveries in psychology have been made at the turn of the 19th and 20th centuries by Sigmund Freud (1856-1939) and Carl Gustav Jung (1875 – 1961).

In the center of these discoveries is situated the concept of unconscious mind. At present, the idea of the mind, as an iceberg in which consciousness is only a small top is almost universally accepted. But no one understands why and what for the mind is so constituted. From the point of view of computer science, building of information system in which most of information including vital is hidden from the user and can be accessed only by using of special psychoanalytical techniques seems absurd.

However, this phenomenon becomes absolutely clear, if we assume that the structure of the psyche like the structure of human society. The last has a highly hierarchized, multi-level system of communities: countries, parties, nations etc. Above we have defined the term "clan", as the total for all of them. This justifies the use of the outdated term "subconscious" for our purposes.

Definition: The subconscious mind of a clan is the conscious mind of all its sub-clans.

Let us show it on the simplified, two-layer model of society: the country and the people who inhabit it. Consciousness of a country is the information that circulates in the media officially approved by the government. Obviously, this information may differ far and away from that which people owned individually. The more totalitarian the ruling elite of the state is the more intense is the level mismatch.

According to the ideas of psychoanalysts' jokes hint of repressed in unconscious contents without naming them directly. A political joke evoked in the Soviet Union stronger laughter than a sexual one.

Example: "On the agenda of the collective farm Communist party meeting, there are two issues: the building of the shed and the building of communism. Due to the lack of wooden boards the meeting moved immediately to the second question." Obviously, we are talking here about a prohibited, taboo subject: the socialist economy is inefficient.

The collapse of the Soviet regime was caused mostly by repression of this fact. It shows how ignoring some contents of the subconscious can be fatal to the system. The example of China shows that timely informed content of the subconscious could not only save the system, but also to ensure a sharp jump in its development.

One can assume, that intelligence can be bound with the possibility to make the unconscious conscious.

3 Process of Transformation - make the Unconscious Conscious

Carl Gustav Jung highlighted the process of "transformation" – transport of unconscious contents to the conscious mind. He said: "Until you make the unconscious conscious, it will direct your life and you will call it fate." We suppose that process of transformation and process of social evolution are two manifestations of the same process.

One can suppose that this process must be cyclical, because for adaptation to the environment one needs always new and new ideas about the changing real world be launched and implemented. Let us look for phases of this cyclical process.

Jung described [7] together with the well-known dichotomy of extraversion (E) vs. introversion (I) two further dichotomies: intuition (N) vs. sensation (S) and feeling (F) vs. thinking (T). The last four phenomena he has called "psychological functions" [8].

Aushra Augustinavichiute [9] was the first, who supposed, that psychological functions of Jung are not pure „psychological" phenomena, but belong to the fundamental natural laws of information processing in systems of some kind. She showed that the activity of psychological functions is arranged in a cycle which she compared to the Carnot cycle of energy transformations in the internal combustion engine. The four "psychological" functions correspond to four evolutionary phases described in [4, 10].

Hypothesis: We highlight in the life of elite four phases: emergence of an idea (hypothesis, model) and the elite around it (N); elite struggle for public acceptance (F); rise to power and the implementation of the bearing model of the elite in the life of society (S); institutionalization of this model and embedding it into the existing system of public institutions (read - discovered earlier models, T).

Why we have just four phases of our process? It can be the result of two further dichotomies (dichotomy seems to be a very important system-forming concept in our research domain):

- Bottom-up vs. top-down processes. The structure of clans is hierarchical and information flows in such structure can be directed upwards or downwards.

- Idea-oriented vs. system-oriented processes. A process of the first kind refers on the new born idea: the generation, development, implementation of it. A system-oriented process refers rather on the place of the new idea in the existing system of old ones.

4 Four phases of the transformation (evolutionary) process

Before we start discussing these four phases of information processing we would like to make two important suggestions and give one example.

1. After reading of sections 4.1-4.4 one can make the conclusion our work is dedicated to philosophy, psychology, sociology or history. This conclusion is absolutely false. This work is dedicated exclusively to the computer science because the natural mechanism of information processing is our only research subject.

2. In the focus of our research is the term “model” [11]. All information processes we discuss have some models of real world as their processing objects. Normally will be used several other words in order to express this concept. So a new born model on the phase “N” is called “idea”. On the phase “F” can be used the term “ideological position”. At phase “S” it can be called “aim”. And on the phase ”T” the terms “conception”, “tradition” or “institution” can be used.

And now we consider an example of elite developing. The basic idea of Karl Marx: "The bourgeoisie exploits the proletariat." When Georgy Plekhanov translated "Communist Manifesto" in Russian language and established in 1883 the group of "Emancipation of Labor" (the early years consisted of four or five people), it was a typical N-elite, implementing this idea. Social Democrats quarreled hoarse, then shook hands and went home. Then came a time when the Mensheviks and the Bolsheviks ceased bowing in the streets of Geneva and met, crossed to the opposite sidewalk. They no longer spent time trying to prove something to each other, but actively promoted their ideas among the masses. This means that these elite moved into "emotional” F-phase. By October 1917, the Bolsheviks were turned by Lenin into a cohesive fighting S-elite. They seized power and not just avoided to say hello with their opponents, but killed them. Stalin then turned Communists in oiled machine - T-elite, and the elite of these elite became nomenclature - the ruling elite of the Soviet state. Behind the bipolar view of the world (the USSR and Warsaw Pact countries against the U.S. and NATO) was hidden possibly the confrontation between two T-elites: nomenclature and "gentlemen". Gentlemen elite won.

4.1 N: Intuition - the Creation of Ideas and Elite around Their

The N-process is idea-oriented bottom-up process. Its content is the generating and developing of an idea, but not the implementation of it. Jesus Christ and early Christianity is a good sample of a young elite creating and developing around a new model of the real world.

"Since intuition is essentially unconscious process, the essence of its very difficult to comprehend consciousness" - writes Jung [7]. That is, intuition can be a very initial phase of transformation, yet almost completely submerged in the unconscious.

This process dominated i.e. in times of appearance of big world religions: about 3000 BC – gods of ancient Egypt; about 1500 BC – the pantheon of gods of ancient Greece (and later - Rome), Judaism; about the time of Jesus Christ; between 1400 and 1775: Renaissance, Reformation and Enlightenment. The common feature of all those movements is the intellectual and spiritual nature of those. The most of ideas are related to the term “sense”, i.e. “sense of life”. Therefore ideas have for a society not pure the utilitarian meaning – to feature the possible changes and developments of system. They provide the unity of the society too. There are national ideas, idea of God and so on. But the information processing is the main function of phase “N”. The big world religions won, because they mirrored the reality more adequate as early pagan beliefs. The N-time is a time of active exploration. The period 1400-1775 was the “Age of Discoveries” too. The ships of Columbus, Magellan and Vasco da Gama were plugging the seas.

As an example of N-periods, we can consider also small (only 15 years) and significantly less well-known period. Plekhanov creates the "Emancipation of Labor" group in 1883. Within the 15-year period, close to the beginning of which is this date arise social democratic and labor parties in Spain, France, Switzerland, Austria, Sweden, Italy and Holland. Also other ideas were activated. In 1881, Eugen Dühring, "genuine founder of Nazi anti-Semitism“ [12], published a book entitled "The Jewish Question as a Racial, Moral, and Cultural Question." Wave of pogroms, that swept the south of Russia in 1881, led to the emergence in 1881-82 years clubs, societies and groups, aiming to revive of the Jewish people in Eretz Israel. Ideas of the Enlightenment, already victorious in many developed countries of the West, received support in Turkey, where the doctor and poet Abdullah Cevdet founded in 1889 the Young Turk movement. The relationship of these four types of elite - Communist, Nazi, Zionist and democratic - dictate a lot in the history of the next century.

4.2 *F: Feeling - Battle of Ideas*

The F-process is system-oriented bottom-up process. Its content is the evaluation of the new idea and comparing it with old models which must be replaced by the new one.

The Russian-speaking community in the world is going through a period of sharp polarization around the issue of Ukraine. The last such period began about 120 years ago and led to a series of Russian Revolutions (1905, 1917) and Civil War (1918). Another example would be the splitting France on Dreyfusards and Antidreyfusards around 1995. Thus, we have now infrequent opportunity to watch the process F from the inside.

The first thing that catches the eye: polarization does not follow state or national borders. Public opinion is split in both countries: Ukraine and Russia. Moreover, in the opposite camps very often are members of the same family. This suggests that neither national nor economic interests play a decisive role; the process in question has some different nature. (It is the information processing nature.)

The second observation: the inability to convince the opponent by any logical arguments. It would lead to inanity of any discussions. But these discussions are watching by those persons who have not yet adjusted to the conflict. But for these persons logical arguments play a rather minor role. First of all, they're watching moral conduct of the parties to the conflict. Thus, the number of early Christians grew particularly cool in those periods when the Roman authorities have made them especially severe persecution. Actually, the death of the Savior is a striking confirmation of this rule.

Despite of attempts to falsify, considered information process, which has almost purely emotional basis, provides a surprisingly powerful tool for an adequate choice between two or more competing elite, and therefore - between their models of the real world. Simplifying - who cheats or uses harsh terror is obviously less sure of the rightness of their ideas, and is struggling only for obtaining or maintaining of certain privileges and wealth. It provides the foundation for the successful functioning of democracy.

The F-function provides also mechanisms for selection of the “correct” ideas. In psychology it manifests itself in emotions - inner struggle between the different ideas.

In the life of a society emotions find correspondents in all kinds of migration. The social migration and its radical form – revolution - represent the struggle of different points of view and provide the switch of a society correspondent to the dominating ideas. The period of 1775-1850 is a sample of a world F-period. It lasted from 1776 (US Independence Declaration), throw 1789 (Grate French Revolution), Napoleon wars, the revolutionary phase of Risorgimento in Italy (1815-1849), Revolutions of 1830 in Belgium, Poland, France and to all-European Revolution of 1848. In cultural history it is the period of romanticism. “Romanticism is a complex artistic, literary, and intellectual movement that originated in the second half of the 18th century… The movement validated strong emotion as an authentic source of aesthetic experience, placing new emphasis on such emotions as trepidation, horror and terror and awe.” [13].

Not seeing the opportunity to fight for their ideas on the old place, groups of people (Huguenots from France, Jews from Egypt) resort to geographical migration. Often this leads to the birth of some new states and death of old other. Thus, the Aegean migrations (approximately 1250 to 850 BC) founded the Hellenistic civilization and ruined the Cretan-Mycenaean culture. At the same time, Egypt has lost the role of a grate empire. In the Germanic migrations of 250 to 650 came into motion Huns, Goths, and Teutons. In their drive to the west they have destroyed the Roman Empire. A rapid growth of geographical migration began in 18 century: were colonized South and North America, Australia, New Zealand, Siberia. In the 19th century over 50 million people left Europe for the Americas.

4.3 S: Sensing – Implementing Ideas in Reality

The S-process is idea-oriented top-down process. Its content is the implementation of the new idea in the real life of a clan.

So, in the N-period will be generate ideas. In the F-period they go in struggle for the minds of people. To begin of the S-period the chaos will be continued, but the dominated idea or the dominated complex of ideas will be formed. Clever and strong guys - “sensing types” [8] - understand it. The strength of those people is based not on pure tyranny, but on the good understanding of the trends. Therefore they have the support of the biggest part of society. Hear will be used the force and violent. Such strong personalities as Charles Martel in Francia or Napoleon in France come to the end of an F-period with the inevitability of a hurricane.

Fisch [14] characterizes so the period 1850-1914 years: “The second half of the 19th century is ... less phase of great theoretical concepts and fundamental ideas than time the introduction and implementation of such ideas. From theory arises practice, from ideas - reality. Here are just a few examples. Democracy, and also a modern parliamentary, was not invented after 1848. But prior to that date, she played only a minor role in the political practice of European states, and after 1918, and even more - after 1945, at least formally, was no longer even the subject of any debate. Still less nationalism became invention in 1848. But it was after 1848, it became one of the biggest political forces. The industrial revolution in 1848 has long begun. But only then it turned into a phenomenon, which subdues all the character and style of life in Europe. Socialism and liberalism in 1848 were fully developed concepts. In politically effective movements they become in many areas only then. ... This preponderance of practice over theory, implementation and insistence over creating concepts has the effect that the era in retrospect emits less gloss than the time of spiritual flight. Practical issues are becoming more important than originality of ideas. ... The greatest merit of this behavior is that only through him great masses of people can directly benefit from initial concepts.”

This is the best description of the (introverted) S-phase we have ever seen, despite of Jörg Fisch is a Swiss historian and have never worked on the concept discussed here.

4.4 T: Thinking - Law and Order Instead of Arbitrary Power

The T-process is system-oriented top-down process. Its content is the balancing and optimizing of the models system after implementation of a new model.

The napoleons provide implementation of new ideas with the strength of their authority. But they can die or be won by their enemies. They and their followers can make errors. Bismarck won the concentration of power for emperor Wilhelm I, but it results in the catastrophe of two world wars challenged by politic of Wilhelm II. Therefore all what was formerly supported with the authority of the ruler and with the fear to contradict him will be in the T-phase “institutionalized"- equipped with specific mechanisms of its realization.

The T-model can be represented by legal mechanisms, strong political principles or the patterns of behavior in certain situations. Thus, the codex Napoleons, the department structure of France and more other things was not cancelled by Bourbons (legal mechanisms). Britain was always against the "strongest man" in Europe, to ensure the conservation of European equilibrium (political principles).

The low-oriented T-countries was or are: the Mesopotamia in times of Hammurabi (about XVIII century BC) - he created the famous codex of laws and he made decisions for private persons and against the state interests. The Roman Republic was founded around 500 BC. It was designated by very strong consistency of its domestic and foreign policy, and adherence to lows and legislation. Around 450 BC in Rome was formed the Twelve Tables statute book, which was issued in twelve wooden tablets in the Roman Forum. It was obviously the T-empire. Iron logic of the policy, strength of the established rules, and presence of clear principles are its main features. Also, the Western kings, emperors and popes from 1025 to 1400 were characterized by these properties. Precisely the strength of principles and not the guns has allowed the papacy to gain the upper hand over to Germanic emperors. The modern west nations was created mostly between X and XIV centuries: British (UK, later - US), Germans. There are ca. 20 times more juries in USA as i.e. in Japan.

5 Conclusions

The evolutionary model of intelligence allows better understanding of several phenomena known but incomprehensible in psychology: unconscious mind, emotions, other psychological functions and psychological types and functions discovered by C.G. Jung. The information processing sense of some sociological phenomenon (self-identification, revolutions and wars) is with this model better understandable too.

All of these can be interpreted as assertions that this model is adequate.

In order to further proof of this hypothesis we presume to build a computer model of this information processing mechanism.

6 Références

1. McElreath, R. B. (2007). Mathematical models of social evolution. Chicago: The University of Chicago Press.

2. Frank, S. (1998). Foundations of Social Evolution. Princeton, New Jersay: Princeton University Press.

3. Marshall, J. (2015). Social Evolution and Inclusive Fitness Theory. Princeton, New Jersay: Princeton University Press.

4. Weisband, I. (2014). Modelling of the Natural Evolutionary Computation in the Human Society. ECTA, 6-th International Conference on Evolutionary Computation Theory and Applications, (pp. 314-319). Rome, Italy.

5. Shapiro, F. (2014, April 28). Who Wrote the Serenity Prayer? The Chronicle Review.

6. Pareto, V. (1923). Trattato di sociologia generale. Firenze.

7. Jung, C. G. (1976). Psychological Types. Princeton: Princeton University Press.

8. MBTI. (1999). Retrieved from www.knowyourtype.com

9. Augustinavichiute, A. (1998). Socionics: book 1. Introduction. S-Petersburg: Terra Fantastica.

10. Вайсбанд, И. (2010). 5000 лет информатики. Москва: Черная белка.

11. Уёмов, А. (1971). Логические основы метода моделирования. Москва: Мысль.

12. Гливаковский, А..Нацизм: Учение о расовой борьбе. Россия, 1993, 1

13. Enlightenment. (2001). Retrieved from Wikipedia: http://en.wikipedia.org/wiki/Age_of_Enlightenment

14. Fisch, J. (2002). Europa zwischen Wachstum und Gleichheit 1850-1914. Stuttgart: Ulmer.

Cooperative Games with Monte Carlo Tree Search

CheeChian Cheng and Norman Carver
Department of Computer Science, Southern Illinois University, Carbondale, IL 62901, USA

Abstract— *Monte Carlo Tree Search approach with Pareto optimality and pocket algorithm is used to solve and optimize the multi-objective constraint-based staff scheduling problem. The proposed approach has a two-stage selection strategy and the experimental results show that the approach is able to produce solutions for cooperative games.*

Keywords: Monte Carlo Tree Search, Multi-Objective Optimization, Artificial Intelligence, Cooperative Game.

1. Introduction

Monte Carlo Tree Search (MCTS) is thriving on zero-sum games like the Go board game and there are many MCTS extensions that further optimize the selection strategy and improve the performance of the algorithm, e.g. RAVE [5]. In this paper, we are proposing to use MCTS to solve cooperative games and utilize the Staff Scheduling problem to show that the approach is able to produce solutions for cooperative games. The objective of a cooperative game is to maximize the team's utility while at the same time optimizing the individuals utilities.

1.1 State-Of-The-Art

Optimum or near optimum solutions to the staff scheduling problem can be found by various approaches like Genetic Algorithm, Constraint Programming, Integer Programming and Single Player Monte Carlo Tree Search. As pointed out in the Single Player MCTS approach [4], the MCTS is easy to implement and the algorithm can be terminated if it runs out of resources, e.g. the search will end if it runs out of time.

2. Background Work

2.1 Tree

A tree structure is a data structure in computer science that represents information in a hierarchical manner (non-linear) resembling a tree as shown in Fig 1 [3]. Information is stored in tree nodes. A tree node can be a root node, a branch node or a leaf node. A root node is a node that does not have a parent node. A branch node is a node that has a parent node and has at least one child node. A leaf node is a node with a parent node but without a child node. Root, branch and leaf nodes are labeled as A, B and C respectively in Fig 1.

2.2 MiniMax Tree

A minimax tree is a tree structure (in decision theory and game theory fields) that represents a zero-sum game's utilities as shown in Fig. 2 [7]. A zero-sum game is a competitive game where players are competing against each other to win the game, i.e. the gain of one player is the loss to other players. A minimax strategy is the strategy to minimize possible loss to a player when the player is making a decision.

2.3 Multi-armed Bandit

The objective of Multi-armed Bandit is to find the optimal strategy for a gambler to pull the levers of a row of slot machines in order to gain maximum reward. It is measured by the regret ρ as shown in Eq. 1.

$$\rho = T\mu^* - \sum_{t=1}^{T} r_t \tag{1}$$

where T is the number of rounds the gambler has pulled the levers, μ^* is the maximal reward mean and r_t is the reward at time t. The regret, ρ is the difference between the reward for optimal strategy and the total collected rewards after T rounds [1]. The average regret is ρ/T and it can be minimized if the gambler played enough rounds as $lim_{T\to\infty} \frac{\rho}{T} \to 0$.

2.4 Monte Carlo Tree Search

Monte Carlo Tree Search has been used with various problems, including but not limited to, the Go board game,

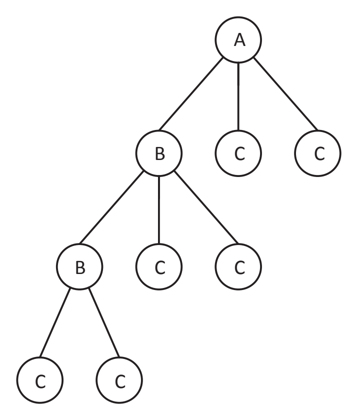


Fig. 1: Tree Structure.

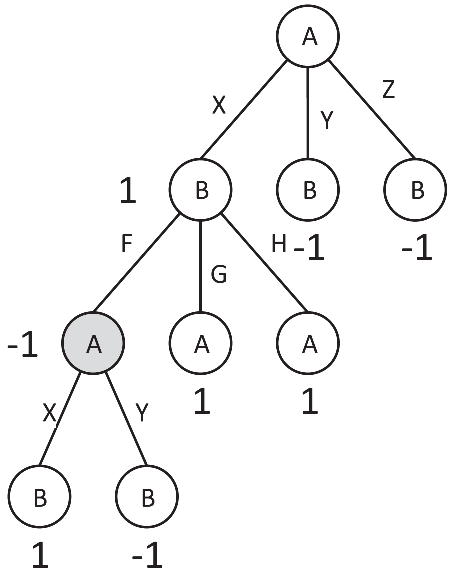


Fig. 2: MiniMax Tree.

real time strategy games, platform video games and staff scheduling optimization. Monte Carlo Tree Search consists of four (4) phases [2], they are:

1) Selection
2) Expansion
3) Simulation
4) Back propagation

In the Selection phase (Fig. 3), the best UCT (as shown in Eq. 3) of a tree node is selected by working recursively from the root node of the tree until a leaf node is reached. The leaf node (or terminal node) will be selected as the candidate for next phase. In the Expansion phase, a legal move will be randomly (uniform distribution) selected from all the possible moves based on the selected node (from selection phase). A new node (resulting from the move) will be added to the selected node as a child node as shown in Fig. 4. The simulation phase will begin with randomly sampling the possible moves for the new node (as shown in Fig. 5) until it reaches a terminal condition, e.g. until the game is over or all players can no longer make a move. The reward is then calculated and back propagated from the simulation phase's terminal node up to the root node of the Monte Carlo tree as shown in Fig. 6. While traversing through each node during the back propagation phase, the visit counter in each node is incremented by one to indicate how many times it has been visited.

2.5 Pareto Optimal

In game theory, Pareto Optimality is a situation where one's utility cannot be improved without degrading others' utility. A Pareto front consists of all the Pareto optimal points, as shown in Fig. 7 [8].

3. The Problem

Creating a monthly schedule for staffing is a frequent task for schedulers in any human resource centric enterprise. Keeping the staff happy is extremely important in a human resource centric enterprise as it boosts the morale of the staff, improves productivity and the staff retention rate. Staff schedule preferences are soft constraints for scheduling, which the schedulers will try their best to accommodate. Hard constraints are the rules that cannot be broken, for instance, double booking a staff, back-to-back shifts and not honoring staff's off day request.

3.1 Hard Constraints

Hard constraints are the rules that the schedulers cannot violate. If the scheduler breaks the hard constraints, the solution/schedule is considered not feasible. The hard constraints are:

- Hard Constraint #1 - Each staff must work at least the minimum contract hours. (HC#1)
- Hard Constraint #2 - It must be at least 12 hours apart between any two shifts for a staff to work in. (HC#2)
- Hard Constraint #3 - Scheduler cannot assign a shift to a staff on his/her requested off-day. (HC#3)

3.2 Soft Constraints

Soft constraints are the constraints that the schedulers will try to accommodate as much as possible. The soft constraints are:

- Soft Constraint #1 - Staff's work day preference, i.e. weekday or weekend. (SC#1)
- Soft Constraint #2 - Staff's shift preference. (SC#2)

4. Proposed Approach

We propose to use Monte Carlo Tree Search with Pareto optimality and pocket algorithm utilizing the multi agents approach. Each agent in turn makes its move by either exploring or exploiting its current situation. Each tree node

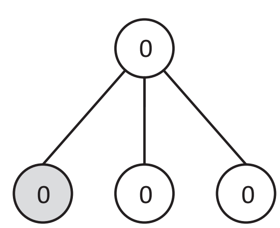


Fig. 3: Selection.

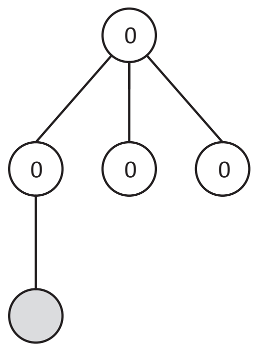


Fig. 4: Expansion.

in the Monte Carlo Tree has a utility vector that consists of the team's utility and each agent's utility as shown in Eq. 2. Each phase of Monte Carlo Tree Search is described in the following subsections.

4.1 Utility Vector

Utility vector, as shown in Eq. 2 contains the utility for each agent and the team. The u_{team} is the utility for the whole team and contains the value in the range of 0 and 1. The u_{team} indicates whether the solution is a feasible solution or a non-feasible solution. The u_{team} will be set to 1 if the solution is feasible otherwise it will be set to 0. A feasible solution is a solution where all agents' assignments do not violate the hard constraints. u_i is the utility for the agent i and it is in the range of 0 and 1. The agent's utility is the measurement of how well the solution is accommodating to the agent's preferences, i.e. soft constraints.

$$\vec{U} = [u_{team}, u_1, u_2, ..., u_n] \qquad (2)$$

where u_{team} is the utility for the team, u_i is the utility for agent i.

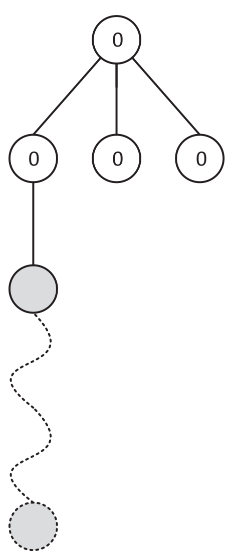


Fig. 5: Simulation.

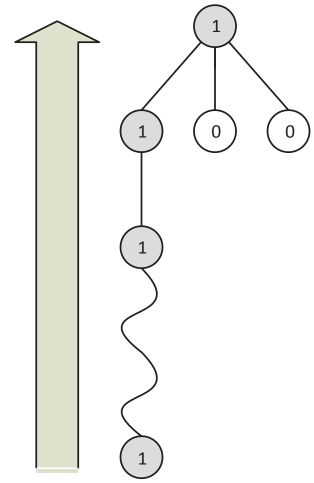


Fig. 6: Backpropagation.

4.2 Selection

During the selection phase, each agent in turn make its move. When it is the agent's turn to make a move, the agent will select a tree node in 2 sub phases.

1) team utility selection
2) agent's utility selection

Starting from the root of the tree, a tree node will recursively be selected, until a tree leaf is reached. In the first phase, the tree node with the highest mean team utility will be selected if it is not fully explored. If there are multiple tree nodes, the node with the highest mean utility for the agent will be selected. The tree node is being selected for

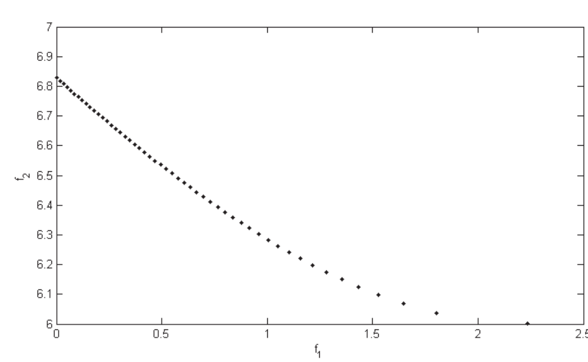


Fig. 7: Pareto Front.

the next phase (simulation). If there is no possible moves for the agent from the selected node, the next agent will be chosen to make a move. A node will be labeled as a terminal node if no agents can make a move from it. The UCT for a team or an agent i is shown in Eq. 3 [6].

$$UCT = \frac{\sum X_j}{n_j} + 2C\sqrt{\frac{2\ln(n)}{n_j}} \tag{3}$$

where $\sum X_j$ is the total utility for the team or agent for the child node j, n_j is the number of visits count in the child node j, n is the number of visits count for the parent node, and C is a constant.

Each tree node has the following properties:

- Agent
- Move
- Number of visits
- Utility vector

The node's "Agent" indicates which agent's turn it is to make a move, while "Move" is the move that the agent has made. "Number of visits" is a counter indicating the number of times the tree node has been visited during the simulation phase. The "Utility vector" keeps track of the team and individual agent utilities.

Algorithm 1 Selection

INPUT: TreeNode, AgentList, CurrentAgent
OUTPUT: SelectedNode

```
CurretNode = TreeNode
while CurrentNode is not leaf do
    Candidates = tree nodes with highest UCT_team that are
    not fully explored
    if there are multiple candidates then
        Candidates = Candidates with the highest UCT_j
    end if
    if no Candidates then
        Indicate the tree is fully explored and stop the search
        return null
    end if
    CurrentNode = randomly (uniform) pick one of the
    Candidates
end while
return CurrentNode
```

4.3 Simulation

In order to harness the advantage of the multi-armed bandit model [1], the simulation phase will simulate all possible moves available to the agent. An agent will be skipped if the agent can no longer make a move from the node. The simulation phase will cease when no agents can make any moves. At the end of each simulation phase, the utilities for the team and the individual agents are computed.

Algorithm 2 Simulation

INPUT: TreeNode, AgentList
OUTPUT: SubTrees

```
InitialAgent = TreeNode.Agent
result = [] // empty
Get all possible moves for this tree node
for each move in possible moves do
    NextAgent = next agent (round-robin based on current
    agent and agent list)
    NewTreeNode = new tree node with agent and move
    SubTree = Sample(NewTreeNode, AgentList)
    Append SubTree to result (as an array)
end for
return result
```

Algorithm 3 Sample

INPUT: TreeNode, AgentList
OUTPUT: SubTree

```
NextAgent = next agent of TreeNode.Agent (round-robin)
Get all possible moves for this tree node and current agent
CurrentAgent = NextAgent
NewMove = move
repeat
    Create a tree node to represent the NewMove and
    CurrentAgent
    Add the NewNode as a child to CurrentNode
    CurrentNode = NewNode
    CurrentAgent = next agent of CurrentAgent (round-
    robin)
    Get all possible moves for CurrentNode and CurrentA-
    gent
    NewMove = randomly (uniform) pick one of the pos-
    sible move
until no agents can make a move
Calculate utilities for current node
return TreeNode
```

4.4 Expansion

The results from the simulations (as sub-trees) will be merged into the Monte Carlo Tree. Each simulation result will be tested for Pareto optimality. The Pocket algorithm is used to remember the Pareto optimal solutions.

Algorithm 4 Pocket Algorithm

INPUT: Pocket, terminal nodes

for each node in terminal nodes **do**
  Remove the solutions from the Pocket that are sub-optimal to node's utility vector
  if node's utility vector is not sub-optimal among the solutions from the Pocket **then**
    Add node to the Pocket
  end if
end for

4.5 Pocket Algorithm

Pocket algorithm consists of two (2) phases, they are:

1) Removing suboptimal solutions
2) Adding solutions from the simulation phase to the solution pool (also known as pocket)

In phase 1, each node from the simulations phase is compared to all the nodes in the solution pool. The nodes in the solution pool will be removed if they are suboptimal to the nodes from the simulation phase. In phase 2, each node from the simulation phase is compared to all the nodes in the solution pool and the node from the simulation phase will be added to the solution pool if they are not suboptimal to any solutions in the pocket.

4.6 Back Propagation

The utility vector is propagated from the terminal node back up to the root node of the Monte Carlo Tree. The tree node will be marked as fully explored if all child nodes have been explored. The leaf node (or terminal node) will always be marked as fully explored.

Algorithm 5 Back Propagation

INPUT: node

$\vec{U}$ to node's utility vector
Node = parent node
while node is not root **do**
  Add $\vec{U}$ to node's utility vector
  Increment node's visit counter
  Node = parent node
end while

5. Experiments and Results

The proposed approach has been applied to the constraints based staff scheduling problem, where each staff viewed as an agent and the shifts being the moves that the agents can make. Each agent has its own constraints including off days, preferred working shifts and preferred working days. Table 1 shows the preferences and their respective off-day requests for 3 staff members.

Table 1: Experiment Criteria.

Staff	Off Day	Preference
#1	1st day of the month	Prefer to work during week days
#2	2nd day of the month	Prefer to work during weekend

Several experiments were conducted with the criteria in Table 1. The results are discussed in the following subsection. All experiments were run with Internet Explorer 11 on an Intel Xeon 3.4GHz, 8GB RAM. The programs were written in typescript (a superset of javascript). The utility for each agent is computed as per Eq. 4. Each experiment was run with 100 simulations and C was set to 1.414.

Table 2: Experiments.

Days	Item	
	Day	Date
2	Thursday	Jan/1/2015
	Friday	Jan/2/2015
3	Thursday	Jan/1/2015
	Friday	Jan/2/2015
	Saturday	Jan/3/2015
4	Thursday	Jan/1/2015
	Friday	Jan/2/2015
	Saturday	Jan/3/2015
	Sunday	Jan/4/2015

$$u_i = 0.5 \times N \tag{4}$$

where N is the number of soft rules that comply with the preferences of agent i.

5.1 Result Discussion

A feasible solution is the solution that does not violate any hard rules. Fig. 9 and Fig. 11 show the feasible solutions for the 3-day and 4-day schedules respectively. Fig. 10 and Fig. 12 show the solutions that violated the hard rules and so are considered not feasible. For instance, in the 4-day

```
Reward Vector = [1, 0.50, 0.00, 0.50]
Euclidean distance = 0.7071067811865476
  Player = 2  Thu 1/1 7:0 - 1/1 19:0
  Player = 3  Fri 1/2 19:0 - 1/3 7:0
  Player = 1  Fri 1/2 7:0 - 1/2 19:0
  Player = 3  Thu 1/1 19:0 - 1/2 7:0
```

Fig. 8: Solution to 2-day Schedule with $U_{team} = 1$.

```
Reward Vector = [1, 0.50, 0.25, 0.33]
Euclidean distance = 0.6508541396588878
  Player = 2  Thu 1/1 7:0 - 1/1 19:0
  Player = 2  Sat 1/3 7:0 - 1/3 19:0
  Player = 1  Fri 1/2 7:0 - 1/2 19:0
  Player = 3  Thu 1/1 19:0 - 1/2 7:0
  Player = 3  Sat 1/3 19:0 - 1/4 7:0
  Player = 3  Fri 1/2 19:0 - 1/3 7:0
```

Fig. 9: Solution to 3-day Schedule with $U_{team} = 1$.

```
Reward Vector = [0, 0.50, 0.50, 0.50]
Euclidean distance = 0.8660254037844386
  Player = 1  Thu 1/1 19:0 - 1/2 7:0
  Player = 2  Sat 1/3 19:0 - 1/4 7:0
  Player = 1  Fri 1/2 7:0 - 1/2 19:0
  Player = 3  Fri 1/2 19:0 - 1/3 7:0
  Player = 2  Sat 1/3 7:0 - 1/3 19:0
  Player = 1  Thu 1/1 7:0 - 1/1 19:0
```

Fig. 10: Solution to 3-day Schedule with $U_{team} = 0$.

```
Reward Vector = [1, 0.50, 0.33, 0.25]
Euclidean distance = 0.6508541396588878
  Player = 3  Thu 1/1 19:0 - 1/2 7:0
  Player = 3  Fri 1/2 19:0 - 1/3 7:0
  Player = 3  Sat 1/3 19:0 - 1/4 7:0
  Player = 2  Sun 1/4 7:0 - 1/4 19:0
  Player = 2  Thu 1/1 7:0 - 1/1 19:0
  Player = 3  Sun 1/4 19:0 - 1/5 7:0
  Player = 2  Sat 1/3 7:0 - 1/3 19:0
  Player = 1  Fri 1/2 7:0 - 1/2 19:0
```

Fig. 11: Solution to 4-day Schedule with $U_{team} = 1$.

```
Reward Vector = [0, 0.50, 0.50, 0.38]
Euclidean distance = 0.8003905296791061
  Player = 3  Fri 1/2 19:0 - 1/3 7:0
  Player = 2  Sun 1/4 19:0 - 1/5 7:0
  Player = 3  Sun 1/4 7:0 - 1/4 19:0
  Player = 3  Fri 1/2 7:0 - 1/2 19:0
  Player = 1  Thu 1/1 7:0 - 1/1 19:0
  Player = 3  Thu 1/1 19:0 - 1/2 7:0
  Player = 2  Sat 1/3 7:0 - 1/3 19:0
  Player = 2  Sat 1/3 19:0 - 1/4 7:0
```

Fig. 12: Solution to 4-day Schedule with $U_{team} = 0$.

schedule solution in Fig. 12, Staff #1 has an off day request for Jan/1 but the schedule indicates that Staff #1 has to work on Jan/1 from 7am-7pm, thus violating hard constraint HC#3. Take note, however, that the solution does align with the preferences of each staff, i.e. the proposed approach tried to schedule individual staff such that the schedule tends to accommodate with the preferences of individual staff. In Fig. 8, Staff #2 has a zero utility as the solution does not accommodate any of his/her preferences.

6. Conclusion

MCTS has been thriving on competitive zero-sum games like Go. In this paper, we have proposed an approach to solve and optimize cooperative games with MCTS using Pareto optimality and pocket algorithm. Unlike the approach from [4] that used a scalar value (scoring function) for each solution, the solutions from this new approach always align to the soft constraints, regardless of the hard constraints.

References

[1] Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256.

[2] Browne, C., Powley, E., Whitehouse, D., Lucas, S., Cowling, P., Rohlfshagen, P., Tavener, S., Perez, D., Samothrakis, S., and Colton, S. (2012). A survey of monte carlo tree search methods. *Computational Intelligence and AI in Games, IEEE Transactions on*, 4(1):1–43.

[3] Carrano, F. and Savitch, W. (2002). *Data structures and abstractions with Java*. Pearson Education.

[4] Cheng, C., Carver, N., and Rahimi, S. (2014). Constraint based staff scheduling optimization using single player monte carlo tree search. In *Proceedings of the 2014 International Conference on Artificial Intelligence*, pages 633–638. CSREA Press.

[5] Gelly, S. and Silver, D. (2007). Combining online and offline knowledge in uct. In *Proceedings of the 24th international conference on Machine learning*, pages 273–280. ACM.

[6] Kocsis, L. and Szepesvári, C. (2006). Bandit based monte-carlo planning. In *Machine Learning: ECML 2006*, pages 282–293. Springer.

[7] Russell, S. (2009). Artificial intelligence: A modern approach author: Stuart russell, peter norvig, publisher: Prentice hall pa.

[8] Wang, W. and Sebag, M. (2013). Hypervolume indicator and dominance reward based multi-objective monte-carlo tree search. *Machine learning*, 92(2-3):403–429.

Solving Unit Commitment Problem Based on New Stochastic Search Algorithm

H. A. Shayanfar *
Department of Elec. Engineering
College of Tech. and Engineering
South Tehran Branch
Islamic Azad University
Tehran, Iran

O. Abedinia
Electrical Engineering Department
Semnan University
Semnan,
Iran

N. Amjady
Electrical Engineering Department
Semnan University
Semnan,
Iran

hashayanfar@yahoo.com, oveis.abedinia@gmail.com, n_amjady@yahoo.com

Abstract— *Unit commitment problem is one of the large scale nonlinear hybrid integer programming problems which is considered in this paper. Thus, a new stochastic search algorithm has been implemented for solving the mentioned problem. For this purpose, Modified Invasive Weed Optimization (MIWO) has been proposed which is a bio-inspired numerical technique and inspired from weed colonization and motivated by a common phenomenon in agriculture that is colonization of invasive weeds. The proposed algorithm is tested on the power systems in the range of 10–140 generating units for a 24-hours scheduling period and compared to Quantum inspired Evolutionary Algorithm (QEA), Improved Binary Particle Swarm Optimization (IBPSO) and Mixed Integer Programming (MIP).*

Keywords: Unite Commitement, Invasiv Weed Optimization, Stochastic Search.

I. Introduction

Unit commitment (UC) aims to schedule the most cost-effective combination of generating units to meet forecasted load and reserve requirements, while adhering to generator and transmission constraints. Generally, UC is completed for a time horizon of one day to one week and determines which generators will be operating during which hours. This commitment schedule takes into account the inter-temporal parameters of each generator (minimum run time, minimum down time, notification time, etc.) but does not specify production levels, which are determined five minutes before delivery. The determination of these levels is known as economic dispatch and it is "the least-cost usage of the committed assets during a single period to meet the demand" [1-5].

The main purpose of optimal Unit Commitment Problem (UCP) for power system is to find the on/off state of each generating unit and the generation of every committed unit for a given horizon, under various operating constraints, consists of fuel constraints, multiple emission requirements, ramp rate limits, minimum up and down time limits and proper spinning reserves. Since the optimal commitment programming can save huge amount of costs and improve reliability of power system, many methods have been proposed to solve the UCP, such as Lagrange Relax (LR) [6], Dynamic Programming (DP) [7], and Genetic Algorithm (GA) [8]. However, they all have some disadvantages such as; the main problem of LR in the difficulty encountered in obtaining feasible solutions. DP is flexible but it may lead to "curse of dimensionality". The shortcoming of GA is massive calculations and it is difficult in dealing with nonlinear constraints.

This paper proposes a new stochastic algorithm to solve the UC problem. Moreover, we thought a new way to update the on/off status of the units in the form of probability. Meanwhile, the Lambda-iteration method is adopted to solve the economic dispatch problem. The Lambda-iteration method and the proposed modified IWO algorithm are run at the same time for the purpose of finding the solution that has the least total production cost. Furthermore, the correction method and several adjustment techniques are proposed to ensure that the solutions are diverse in the iterative process and satisfy all the constraints.

* Corresponding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

II. Problem Definition

The formulation for the unit commitment is described in detail in this section. The objective of the UC problem is to minimize the total production cost consisting of the generation cost and the start-up cost of the generating units under the circumstance where, the operational constraints and the constraints of the generating units are satisfied in the scheduling period [9-10]. Where, it can be as;

$$F_T = \sum_{t=1}^{T}\sum_{i=1}^{NT}[C_i(P_i^t.u_i^t) + S_i u_i^t(1-u_i^{t-1})] \quad (1)$$

Where, F is the total production cost; T the number of hours in the scheduling period; NT the number of generating units; and u^t_i on/off status of the unit i at hour t, 1 represents the on status of the unit i at hour t, 0 represents the off status of the unit i at hour t. $C_i(p_i^t)$ is the generation cost function of unit i. It is normally a quadratic polynomial represented by;

$$C_i(p_i^t) = a_i(p_i^t)^2 + b_i(p_i^t) + c_i \quad (2)$$

Where, p^t_i generation output of unit i at hour t; and a_i, b_i, c_i are parameters of unit i. S_i is the start-up cost of unit i which is related to the duration time of the off state of unit i. It can be expressed by:

$$S_i = \begin{cases} H_{SCi} & M_{DTi} < X_{OFFi}^t \le M_{DTi} + C_{SHi} \\ C_{SCi} & M_{DTi} + C_{SHi} < X_{OFFi}^t \end{cases} \quad (3)$$

Where, HSCi is hot start-up cost of unit i; CSCi the cold startup cost of unit i; Xt OFFi the duration time during which unit i keeps off status at hour t; CSHi cold start time of unit i; and MDTi the minimum down time of unit i.

Also, for the constrains;

1) System power balance constraint:

$$\sum_{i=1}^{N} u_i^t p_i^t = D^t \quad (4)$$

2) Thermal Generator Constraints:

a) Unit’smaximum up/down reserve contribution constraints:

$$US_i^{max} = d\% \times P_{i,r}^{max} \quad (5)$$
$$DS_i = d\% \times P_{i,r}^{max} \quad (6)$$

b) Unit’s up/down spinning reserve contribution constraints:

$$US_i(t) = min\{ US_i^{max}, P_{i,r}^{max} \_ P_{i,r}(t)\} \quad (7)$$
$$DS_i(t) = min\{ DS_i^{max}, P_i(t) \_ P_{i,r}^{min}\} \quad (8)$$

c) Unit’s ramping up/down capacity constraints:

$$UR_i(t) = min\{ UR_i^{max}, P_{i,r}^{max} \_ P_i(t)\} \quad (9)$$
$$DR_i(t) = min\{ DR_i^{max}, P_i^{max} \_ P_{i,r}^{min}\} \quad (10)$$

d) Unit generation limits:

$$P_i^{\min}(t) \times U_i(t) \le P_i(t) \le P_i^{\max}(t) \times U_i(t) \quad (11)$$

$$P_i^{\max}(t) = \begin{cases} \min\{P_{i,r}^{\max}, P_i(t-1) + UR_i^{\max}\} & \text{if } U_i(t) = U_i(t-1) = 1 \\ \min\{P_{i,r}^{\max}, P_i(t-1) + SR_i\} & \text{if } U_i(t) = 1, U_i(t-1) = 0 \end{cases} \quad (12)$$

$$P_i^{\min}(t) = \begin{cases} \min\{UR_i^{\max}, P_i(t-1) - DR_i^{\max}\} \\ P_{i,r}^{\min} \text{ if } \quad U_i(t) = 1, U_i(t-1) = 0 \end{cases}$$
$$if \quad U_i(t) = U_i(t-1) = 1$$

e) Minimum up/down time constraints:

$$[t_{ON,i}(t\text{-}1)\text{-}T_{ON,i}] \times [U_i(t\text{-}1)U_i(t)] \ge 0 \quad (13)$$
$$[t_{OFF,i}(t\text{-}1)\text{-}T_{OFF,i}] \times [U_i(t\text{-}1)U_i(t)] \ge 0 \quad (14)$$

III. Modified Invasive Weed Optimization

Invasive Weed Optimization (IWO) is inspired from weed colonization and motivated by a common phenomenon in agriculture that is colonization of invasive weeds. Actually, the weeds have shown with adaptive nature and very robust which turns them to undesirable plants in agriculture. Since its advent IWO has found several successful engineering applications like tuning of robot controller [11], optimal positioning of piezoelectric actuators [12], development of recommender system [13], antenna configuration optimization [15], computing Nash equilibria in strategic games [16], DNA computing [17], and etc.

IWO is a meta-heuristic algorithm which mimics the colonizing behavior of weeds. In this algorithm, the process starts with initializing a population. It means that the population of initial solutions is randomly generated over the problem space. Then the population members produce seeds depending on their relative fitness in the population. In other words, the numbers of seeds for each member are beginning with the value of S_{min} for the worst member and increases linearly to S_{max} for the best member [15]. This technique can be summarized as:

A. *Initialization*

In this step, a finite number of weeds are initialized at the same element position of the conventional array which has a uniform spacing of "$\gamma/2$" between neighboring elements.

B. *Reproduction*

The individuals, after growing, are allowed to reproduce new seeds linearly depending on their own, the highest, and the lowest fitness of the colony (all of plants). The maximum (S_{max}) and minimum (S_{min}) number of seeds are predefined parameters of the algorithm and adjusted according to the structure of problem. The schematic seed production in a colony of weeds is presented in Fig. 1. In this figure, the best fitness function is the lower one [11].

C. *Spatial distribution*

The generated seeds are being randomly distributed over the d-dimensional search space by normally distributed random numbers with mean equal to zero; but varying variance. This step certifies that the produced seeds will be generated around the parent weed, and leading to a local search around each plant. However, the Standard Deviation (*SD*) of the random function is made to decrease over the iterations, which is defined as:

$$SD_{ITER} = (\frac{iter_{\max} - iter}{iter_{\max}})^{pow}(SD_{\max} - SD_{\min}) + SD_{\min} \quad (15)$$

SD_{max} and SD_{min} are the maximum and minimum standard deviation, respectively. This step ensures that the probability of dropping a seed in a distant area decreases nonlinearly with iterations, which result in grouping fitter plants and elimination of inappropriate plants.

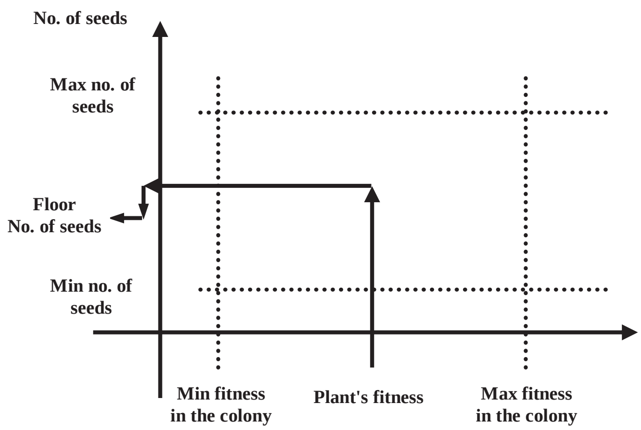


Figure 1. Schematic seed production in a colony of weeds

IV. Numerical Results

In this section the proposed algorithm has been tested over UC problem on the power system with 10, 20, 40, 60, 80, 100, 120 and 140 generating units in the 24-h scheduling period. The 10-unit data is presented in Table 1 and the power demand in the scheduling period is shown in Table 2. The 20, 40, 60, 80, 100, 120 and 140-units data are obtained by duplicating the 10-unit data, whereas, the power demand is proportional to the number of units. Furthermore, the spinning reserve is set to be 10% of the power demand.

Table. 1. Ten Unit System Data

Unit	P_{max}/MW	P_{min}/MW	a/($/MW^{2}h)	b/($/MW^{2}h)	c/($/h)	Min up/h	Min dn/h	Hot start cost/$	Cold start cost/$
1	455	150	0.00048	16.19	1000	8	8	4500	9000
2	455	150	0.00031	17.26	970	8	8	5000	10000
3	130	20	0.002	16.6	700	5	5	550	1100
4	130	20	0.00211	16.5	680	5	5	560	11200
5	162	25	0.00398	19.7	450	6	5	900	1800
6	80	20	0.00712	22.26	370	3	3	170	340
7	85	25	0.00079	27.74	480	3	3	260	520
8	55	10	0.00413	25.92	660	1	1	30	60
9	55	10	0.00222	27.27	665	1	1	30	60
10	55	10	0.00173	27.79	670	1	1	30	60

Table. 2. Load Demand

Hour	Demand/MW	Hour	Demand/MW
1	700	13	1400
2	750	14	1300
3	850	15	1200
4	950	16	1050
5	1000	17	1000
6	1100	18	1100
7	1150	19	1200
8	1200	20	1400
9	1300	21	1300
10	1400	22	1100
11	1450	23	900
12	1500	24	800

Parameters are set as follows: the number of population size 5; the it_{max} is equal to 100, *dim* is considered 1, P_{max} is 5, S_{max} is 25, S_{min} is 0. The program is written in MATLAB R2011a and executed on a 2.5 GHz CPU with 4-GB RAM personal computer. In order to have a comprehensive understanding of the proposed method, 50 trials are done on every test system.

Since the best solution of the 10-unit system of the proposed method is the same as that of QEA, the units' power output of the best solution can be seen in [18].The best solution of the 20-unit system is presented in Table 3

Table. 3. Unit Output of the 20 Unit System's Best Solution

Hour	Generating unit																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	455	455	245	245	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	455	455	295	295	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	455	455	383	383	0	0	0	0	25	0	0	0	0	0	0	0	0	0	0	0
4	455	455	455	455	0	0	0	0	40	44	0	0	0	0	0	0	0	0	0	0
5	455	455	455	455	0	0	130	0	25	25	0	0	0	0	0	0	0	0	0	0
6	455	455	455	455	130	0	130	130	25	24	0	0	0	0	0	0	0	0	0	0
7	455	455	455	455	130	130	130	130	45	45	0	0	0	0	0	0	0	0	0	0
8	455	455	455	455	130	130	130	130	30	39	0	0	0	0	0	0	0	0	0	0
9	455	455	455	455	130	130	130	130	97.5	98	98	20	20	0	0	0	0	0	0	0
10	455	455	455	455	130	130	130	130	162	163	162	33	33	0	0	0	0	0	0	0
11	455	455	455	455	130	130	130	130	162	163	162	98	98	98	10	0	0	0	0	0
12	455	455	455	455	130	130	130	130	162	163	162	98	80	25	10	0	0	0	0	0
13	455	455	455	455	130	130	130	130	162	163	98	65	80	25	10	10	10	10	0	0
14	455	455	455	455	130	130	130	130	97.5	87	33	65	33	25	10	10	10	10	10	10
15	455	455	455	455	130	130	130	130	30	38	25	20	25	25	0	0	0	0	0	0
16	455	455	310	334	130	130	130	130	25	32	25	20	0	0	0	0	0	0	0	0
17	455	455	260	260	130	130	130	130	25	25	30	25	0	0	0	0	0	0	0	0
18	455	455	360	360	130	130	130	130	25	25	33	33	0	0	0	0	0	0	0	0
19	455	455	455	455	130	130	130	130	30	30	105	30	0	0	0	0	0	0	0	0
20	455	455	455	455	130	130	130	130	162	167	105	25	0	0	0	0	0	0	0	0
21	455	455	455	455	130	130	130	130	105	105	105	25	30	25	0	0	10	10	10	0
22	455	455	455	455	0	0	130	0	105	105	0	25	33	25	0	10	0	0	0	0
23	455	455	433	433	0	0	0	0	25	24	0	0	0	0	0	0	0	0	0	0
24	455	455	345	345	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

and 4, which have never been illustrated in detail before. We can see that the generation cost in the scheduling period is 1114874 and the start-up cost is 8400 MW/$, so the total production cost is 1123292. Also, the convergence trend of proposed method over different case studies has been presented in Fig. 2.

Table. 4. Cost of the 20-Unit System's Best Solution

Hour	Generation cost	Start-up cost	Spinning reserve	On/off status
1	27363.24	0	420	11110000000000000000
2	29109.00	0	320	11110000000000000000
3	33111.24	900	280	11110000100000000000
4	37194.44	900	240	11110000110000000000
5	39456.23	560	274	11110001001110000000
6	44277.22	2220	332	11110000000000000000
7	46000.23	0	233	11110000000000000000
8	48302.45	1100	363	11111111111111000000
9	53768.98	1200	309	11111111111111100000
10	60087.54	640	302	11111111111111110000
11	63782.45	120	312	11111111111111111000
12	67687.34	120	322	11111111111111111111
13	61112.34	0	301	11111111111111110000
14	53834.98	0	302	11111111111111000000
15	48287.87	0	260	11111111111100000000
16	43034.48	00	564	11111111111000000000
17	41284.98	0	665	11111111111000000000
18	44635.66	0	464	11111111111000000000
19	47827.45	0	254	11111111111000000000
20	61004.42	640	289	11111111111000111000
21	53456.98	0	279	11111111111000110000
22	44438.87	0	232	11111111111000000100
23	34537.23	0	180	11100001111000000000
24	30012.45	0	220	11100011101000000000

The best, worst and mean values of the total production cost, together with the mean computation time by MIP [18], QEA [19], IBPSO [20] and proposed method for various test systems are shown in Table 5.

Table. 5. Comparison of Simulation Results for Different Systems

Unit	Algorithm	Cost			Mean time
		Best	Worst	Mean	
10	MIP	564647			2
	QEA	563938	564672	563969	19
	IBPSO	563977	565312	564155	27
	Proposed	563933	564222	563945	2
20	MIP	1123908			5
	QEA	1123607	1125715	1124689	28
	IBPSO	1125216	1125730	1125448	55
	Proposed	1123287	1124078	1123768	9
40	MIP	2243020			11
	QEA	2245557	2248296	2246728	43
	IBPSO	2248581	2249302	2248875	110
	Proposed	2242880	2244572	2243581	30
60	MIP	3361614			29
	QEA	3366676	3372007	3368220	54
	IBPSO	3367865	3368779	3368278	172
	Proposed	3361681	3364101	3363112	50
80	MIP	4483194			38
	QEA	4488470	4492839	4490126	66
	IBPSO	4491083	4492686	4491681	235
	Proposed	4482013	4486732	4484502	70
100	MIP	5601857			47
	QEA	5609550	5613220	5611797	80
	IBPSO	5610293	5612265	5611181	295
	Proposed	5601272	5608321	5604172	99
120	Proposed	6722630	6732536	6726624	115
140	Proposed	7891535	7905537	7898747	132

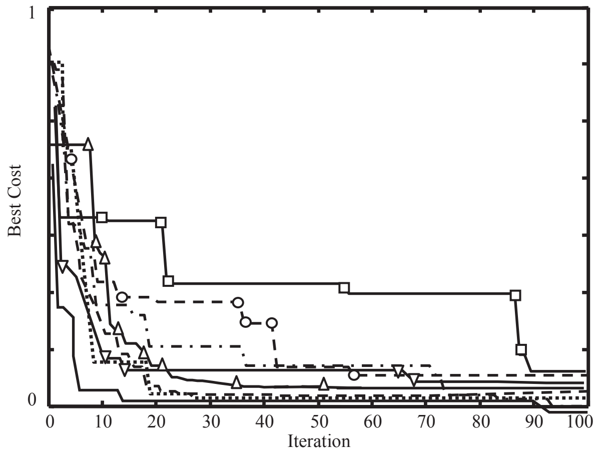


Figure 2. Fitness convergence, Solid; 10 unit, Dashed; 20 unit, Dotted; 40 unit, Dashed-Dotted; 60 unit, Upward-pointing triangle; 80 unit, Downward-pointing triangle; 100 unit, Circul-pointing; 120 unit, Square-pointing; 140 unit

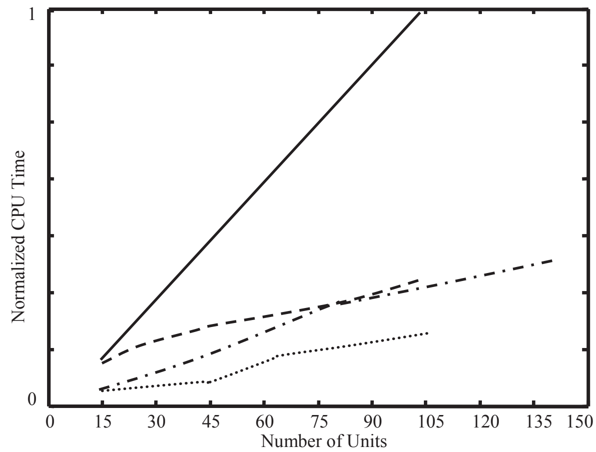


Figure 3. Comparison of different algorithms computation time, Solid; IBPSO, Dashed; QEA, Dotted; MIP, Dashed-Dotted; Proposed

We can see that the best solution of the proposed algorithm is better in most of the test systems and the best solution of proposed algorithm is very close to that of the MIP method in the 60-unit test system. From Fig. 3, it can seen that the proposed method is faster than the proposed method in all the test systems and QEA algorithm in 10, 20, 40 and 60-unit test systems. Although the calculation time of the proposed method is longer than that of the MIP method, the calculation time of the proposed method increases almost linear with the number of the units, which means that it has the capacity of solving large-scale UC problems.

V. Conclusions

In this paper, a new stochastic search algorithm has been implemented for solving the unit commitment problem. Thus, Modified Invasive Weed Optimization (MIWO) has been implemented over this problem which is a bio-inspired numerical technique and inspired from weed colonization and motivated by a common phenomenon in agriculture that is colonization of invasive weeds. The simulation results show that the total production cost of proposed method is less expensive than those of the other methods in the range of 10–100 generating units. In addition, the CPU time of this method increases almost linear with the size of the units, which is favorable for the large-scale power systems.

References

[1] C. D. Yue, C.M. Liu, and M. L. L. Eric, “A transition toward a sustainable energy future: Feasibility assessment and development strategies of wind power in Taiwan,” Energy Pol., Vol. 29, pp. 951–963, 2001.

[2] T. J. Chang,Y.T.Wu,H.Y.Hsu, C. R. Chu, and C.M. Liao, “Assessment of wind characteristics and wind turbine characteristics in Taiwan,” Renew. Energy, Vol. 28, pp. 851–871, 2003.

[3] A. G. Bakirtzis and P. S. Dokopoulos, “Short term generation scheduling in a small autonomous system with unconventional energy sources,” IEEE Trans. Power Syst., Vol. 3, No. 3, pp. 1230–1236, Aug. 1988.

[4] C. Brunetto and G. Tina, “Optimal hydrogen storage sizing for wind power plants in day ahead electricity market,” Renewable Power Generation, Vol. 1, No. 4, pp. 220–226, Dec. 2007.

[5] S. Chakraborty, M. Weiss, and M. G. Simoes, “Distributed intelligent energy management system for a single-phase high-frequency AC microgrid,” IEEE Trans. Ind. Electron., Vol. 54, No. 1, pp. 97–109, Feb. 2007.

[6] A. Merlin, and P. Sandrin, “A new method for unit commitment at Electricite de France”, IEEE Transactions on Power Applied Systems, Vol. PAS-102, pp. 1218-1255, 1983.

[7] W. L. Snyder, Jr., H. D. Powell, Jr., and J. C. Rayburn, “Dynamic programming approach to unit commitment”, IEEE Transactions on Power Systems, Vol. 2, No. 2, pp. 339-350, 1987.

[8] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis, “A genetic algorithm solution to the unit commitment problem”, IEEE Transactions on Power Systems, Vol. 11, No. 1, pp. 83-92, 1996.

[9] J. Juban, L. Fugon, and G. Kariniotakis, “Uncertainty estimation of wind power forecasts”, in Proceedings of the European Wind Energy Conference EWEC08, Brussels, Belgium, 2008.

[10] A. Tuohy, P. Meibom, E. Denny, and M. O’Malley, “Unit commitment for systems with significant wind penetration”, IEEE Transactions on Power Systems, Vol. 24, pp. 592–601, 2009.

[11] A. R. Mehrabian and C. Lucas, “A novel numerical optimization algorithm inspired from invasive weed colonization,” Ecological Informatics, Vol. 1, pp. 355-366, 2006.

[12] H. Sepehri-Rad and C. Lucas, “A recommender system based on invasive weed optimization algorithm,” in Proc.

IEEE Congress on Evolutionary Computation, pp. 4297 – 4304, 2007.

[13] A. R. Mehrabian and A. Yousefi-Koma, “Optimal positioning of piezoelectric actuators of smart fin using bio-inspired algorithms,” Aerospace Science and Technology, Vol. 11, pp. 174 – 182, 2007.

[14] B. Panida, B. Chanwit, O. Weerakorn, "Optimal congestion management in an electricity market using particle swarm optimization with time-varying acceleration coefficients", Computers and Mathematics with Applications, Vol. 3, pp. 1-10, 2010.

[15] N.G. Pavlidis, K.E. Parsopoulos, and M.N. Vrahatis, “Computing Nash equilibria through computational intelligence methods,” J. of Computational and Applied Mathematics, Vol. 175, No. 1, pp. 113–136, 2005.

[16] B. Dadalipour, A. R. Mallahzadeh, Z. Davoodi-Rad, “Application of the invasive weed optimization technique for antenna configurations,” in Proc. Loughborough Antennas and Propagation Conf., Loughborough, pp. 425-428, Mar. 2008.

[17] X. Zhang, Y. Wang, G. Cui, Y. Niu, and J. Xu, “Application of a novel IWO to the design of encoding sequences for DNA computing,” Computers and Mathematics with Applications, 2008.

[18] TW. Lau, CY. Chung, KP. Wong, Quantum-inspired evolutionary algorithm approach for unit commitment. IEEE Trans Power Syst., Vol. 24, No. 3, pp. 1503–1512, 2009.

[19] KA. Juste, H. Kita, E. Tanaka, An evolutionary programming solution to the unit commitment problem. IEEE Trans Power Syst., Vol. 14, No. 4, pp. 1452–1459, 1999.

[20] YW. Jeong, JB. Park, SH. Jang, A new quantuminspired binary PSO: Application to unit commitment problems for power systems. IEEE Trans Power Syst., Vol. 25, No. 3, pp. 1486–1495, 2010.

Biographies

Heidar Ali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Smart Grids, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 495 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM’10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

Improving the Performance of Particle Swarm Optimization for Iris Recognition System Using Independent Component Analysis

Omaima N. Ahmad AL-Allaf
Faculty of Sciences and Information Technology
AL-Zaytoonah University of Jordan, P.O. Box 130, Amman (11733), Jordan
omaimaalallaf@zuj.edu.jo

Abstract – *Recently, iris recognition had been gained growing interest from researchers because of its high accuracy against other person identification techniques. This work presents an iris recognition system based on particle swarm optimization (PSO) and Independent Component Analysis (ICA) as feature extraction algorithm.*
Many experiments were conducted using different: swarm size (20, 40 and 80), PSO iterations (100 and 200) and ICA feature vector length (64, 128, 256 and 512). The results showed that: the best performance of the iris recognition system (high PSNR, high recognition rate and lower MSE) will be increased when increasing the ICA feature vector length to 512, increasing the swarm size to 80 and decreasing the number of iterations to 100. Best obtained value of recognition rate is 98%, best PSNR value is 38 and lower MSE value is 0.0011. The suggested iris recognition system has the ability to recognize un trained iris images but with lower performance.

Keywords: — Iris Recognition, Practical Swarm Optimization (PSO), Independent Component Analysis (ICA)

1 Introduction

Recently, biometric systems for person identification are becoming more important because of increasing the security requirements of organizations in private and public sectors in societies [1][2][3]. These systems provide persons' identification based on their physical features such as iris, face, voice and fingerprints [4].
Traditional methods for personal identification include the token-based methods that use ID cards/keys for authentication and knowledge-based methods that use password for identification. These methods are not reliable. As a promising way of authentication, biometrics aims to recognize a person using physiological and/or behavioral characteristics such as fingerprints, face, voice, and so on [1][2]. Iris recognition can be regarded as the most reliable biometric identification system available that recognizes a person by iris pattern [5][6]. It can obtain high accuracy due to the rich texture of iris patterns. Therefore, biometric systems based on iris patterns play important role in network security and high level security systems [7] because iris has unique patterns and no two iris patterns are the same [8]. A biometric system consists of: sensor module to take biometric data; feature extraction; matching part to compare the feature vectors are with those in template; and decision making to establish the user's identity [9].
Many studies in the literature focused on the development of iris recognition systems based on different approaches, models and algorithms. Kevin, et al (2008) [10] addressed algorithms, experimental evaluations, databases and applications used for iris recognition systems. And Richard (1997) [11] described in details the design and operation of iris recognition systems. And Kresimir and Mislav (2004) [9] presented an overview of biometric methods and discussed their advantages and limitations. Whereas Mayank, et. al (2004) [12] studied and compared algorithms for iris recognition. And Sheela and Vijaya (2008) [13] discussed techniques and databases related to iris recognition. While Bhalchandra, et. al (2008) [8] presented feature extraction methods for iris recognition. And Lenina and Manesh (2009) [14] presented literature related to iris recognition systems and their architectures. Finally, Sajida and Sheikh (2012) [15] highlighted the current state and issues of iris based biometric authentication systems.
Particle swarm optimization (PSO) is a heuristic, population-based, self-adaptive search optimization technique that is based on swarm intelligence to solve optimization problems in many applications. It comes from the research on the bird and fish flock movement behavior. The algorithm is widely used and rapidly developed for its easy implementation and few particles required to be tuned [16]. PSO was first introduced in 1995 by Kennedy and Eberhart [17] and has been growing rapidly. Many literature researches were focused on developing and enhancing the PSO [18..26]. Many literature researches were used PSO for solving recognizing problems such as: Object recognition [27]; Fingureprint [28]; Handwritten Characters Recognition [29]; Face recognition [30][31]; Palmprint recognition [32..35]; and Gait recognition [36..38]. Iris recognition analyses the iris pattern and these patterns are unique, stable and reliable with age in comparison with other biometric features such as face and fingerprint [8]. Many literature researches related to iris recognition systems were based on different algorithms and approaches. Few literature researches were based on PSO for iris recognition [39][40].

Independent component analysis (ICA) is a method for finding underlying components from multi-dimensional statistical data. ICA differs from other methods is that it looks for components that are statistically independent, and non Gaussian. ICA was used for different applications such as: Medical and Finance data, Audio Processing, Array processing and Coding [41..44]. Many literature studies were used ICA for image recognition problems [45..49].

According to optimization features of PSO algorithm, this research is focused on building an iris recognition system based on PSO algorithm. To increase the performance of this suggested iris recognition system ICA will be used as a feature reduction and extraction algorithm. This paper is organized as follows: section 2 includes description of PSO. Section 3 includes description of ICA and section 4 includes research methodology and section 5 includes results. Finally section 6 concludes this work.

2 Practical Swarm Optimization

PSO algorithm can be implemented easily, converge rapidly, and can be applied on large number of samples. PSO is one of the swarm intelligence methods that explore global optimal solution and based on social behavior of birds flocking. It uses swarm of particles as the individuals in population for searching through solution space. Each solution in PSO is implemented as a particle that represents one individual of a population. A particle can be regarded as a point of N-dimension solution space and has a speed which is N-dimension vector. The swarm is population and it is a set of vectors. Each particle has a fitness function (value) associated with it. Each particle adjusts its position and evaluate their position and move closer to optimal point. Particles also compare themselves to their neighbors and imitate the best of that neighbor. Swarm of particles is flying through the parameter space and searching for optimum. The best value that particle in a local swarm reach it is called Lbest. Fig.1 shows the main steps of PSO [16-26]. The following equation is used to compute new velocity of each particle [16-26]:

$$Vi(t+1)=W\times Vi(t) + C1 \times rand \times (Pbest(t)-Xi(t)) + C2 \times rand \times (Gbest(t)-Xi(t)) \qquad (1)$$

Where,

V[]: particle velocity,
Xi: ith particle of swarm
W: weight (random number between 0 and 1).
C1, C2 : the speeding factors (with value 2).
Pbest: represents the best value of the particle i.
Gbest: the best value that one of the swarm particle reach it.

From Eq.1, the new velocity vi(t+1) is affected by: Pbest, Gbest and Vi(t): earlier velocity of ith particle X in time t.

The following equation is used to compute new fitness value of each particle in swarm:

$$Xi(t+1)=Xi(t) + Vi(t+1) \qquad (2)$$

Particle will change its value according to its new velocity (vi(t+1)).

3 Independent Component Analysis

Many feature extraction algorithms used to extract the main feature of image for recognition process. [41][42][43]. Basic ICA assumed that the numeric multidimensional data series are available. Every individual data series is a mixture of a number of statistically independent source signals. ICA is used to solve the inverse problem of finding unknown sources without knowing the mixing conditions.

Let us assume that we observe the *n*-dimensional vector signal $x(t)$, which is the result of an unknown mixing of m statistically independent source signals $s(t)$ [50][51]:

$$\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{s}(t) + \boldsymbol{n}(t) = \sum_{i=1}^{m} s_i(t)\boldsymbol{a}_i^T + \boldsymbol{n}(t) \qquad (3)$$

where $a^T{}_i$ denotes the i-th row vector of an unknown mixing matrix A. ICA can estimate m unknown sources and $m\times n$-dimensional separation matrix $W(t)$. The m-dimensional vector should become (up to the scale and signal permutation) an estimate of original sources.

$$\boldsymbol{y}(t) = \boldsymbol{W}(t)\boldsymbol{x}(t) \qquad (4)$$

The source signals play the role of base vectors of expected image representation subspace, whereas the rows of matrix A represent the feature vectors. The ICA can estimate the sources based solely on the measurement data. The ICA-based texture description scheme is applied over conventional space to adjust the base functions to given images. The base functions are of general nature set in a heuristic way in conventional approaches. In ICA, the search in samples space for a non-orthogonal basis of a subspace that: retains structure of learning data, the base vectors correspond to interesting directions of sample structure; and also after individual rotations of sub-space vectors both an orthogonal basis and statistical independence of border distributions are achieved [50[51].

4 Research Methodology

An iris recognition system based on PSO and ICA for feature extraction is suggested in this work. The main steps of PSO for training/testing iris recognition system were implemented using MatLab2013.This section describes in details the database, training and testing steps of iris recognition system. Fig.2 shows 10 samples of two persons each with 5 images.

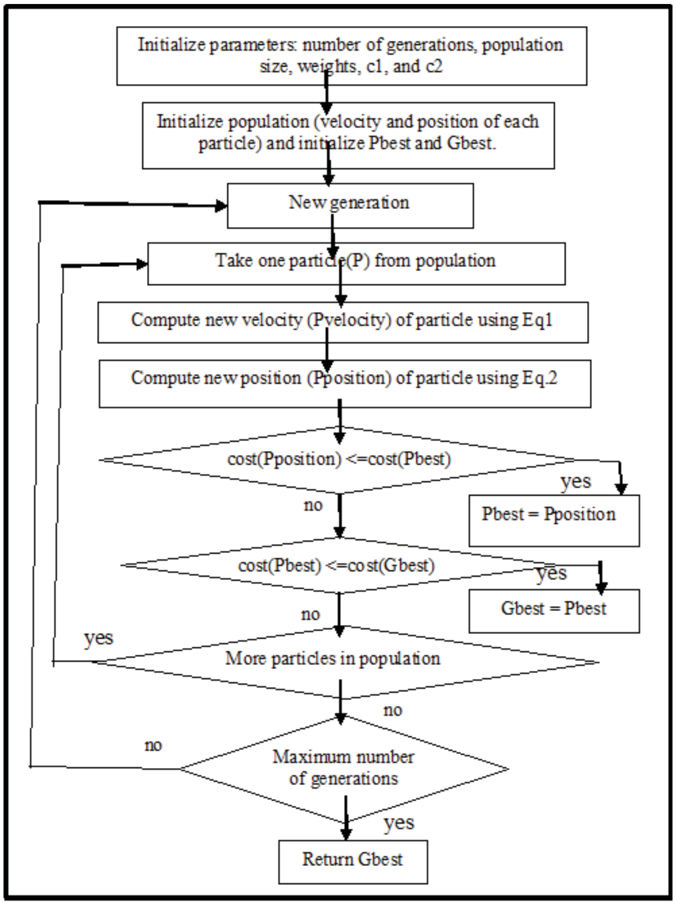


Fig.1 : PSO Algorithm

4.1 Iris Recognition Training Process

The training part of the suggested iris recognition system includes the following steps:

1. Read 100 images (each with 64×64pixels) for 10 persons.
2. Each image is converted from two-dimension array (64×64) to one vector dimension 4096.
3. Add one byte to each image array (4096) of each person to be with size (4097). This byte is used to identify the person with ID and contains the same value for the 10 images of each person. As an example this byte includes the value 1 for the 10 images for person 1. The total size of the array containing the overall 100 images is 4097×100.
4. The person properties will be extracted by applying ICA algorithm for feature extraction. The ICA is computed for each image (with dimension 4097) to produce feature vector with size 64. Many experiments were conducted with different ICA feature vector length (128, 256, 512). At the same time one byte (ID) is added to each feature vector (65, 129, 257, 513) for person identification
5. The classification PSO algorithm is used for each one of the 100 feature vectors (generated using ICA) to produce a classified PSO vector with length equal to swarm size plus one (for person ID) as follows:

Step1: initialize parameters of PSO algorithm as follows: set swarm size (N) equal 80, 40, 20 separately for each experiments, c1 equal 2, c2 equal 2, weight equal 0.7, number of Iterations equal to 100 and 200 for different experiments. Finally set ICA feature vector length equal to 64, 128, 256, 512 separately for each experiments.

Step 2: initialize position and velocity (speed) of each sample. Initialize lbest and gbest

Step3: Calculate fitness function of each sample

Step4: Calculate optimal value of particle swarm (pbest) and optimum value of group (gbest) according to comparison between the current value of particle and the pbest and gbest

Step5: Calculate the new speed of practical according to Eq.1.

Step6: Compute the new position of particle according to Eq.2.

Step7: repeat steps 3 to 6 while there are more iterations to be executed.

Step8: store the features sub set which are represented by vector with 40 values (according to population size) in sub features database: gaitdbf

4.2 Iris Recognition Testing Process

The testing part of the suggested system is described by the following steps:

1. Read one image with 64×64 pixels for any person.
2. Convert this image from two-dimension array (64×64) to one vector dimension 4096.
3. Apply ICA for this image vector (4096) for feature extraction to produce feature vector with size either 64., 128, 256 or 512.
4. Apply PSO algorithm to this feature vector to produce a PSO vector with length equal to swarm size (20, 40, 80).
5. Compare this classified PSO vector with database containing 100 classified PSO feature vectors to determine which vector is most similar to the input vector.
6. Return the ID of the matched PSO vector.

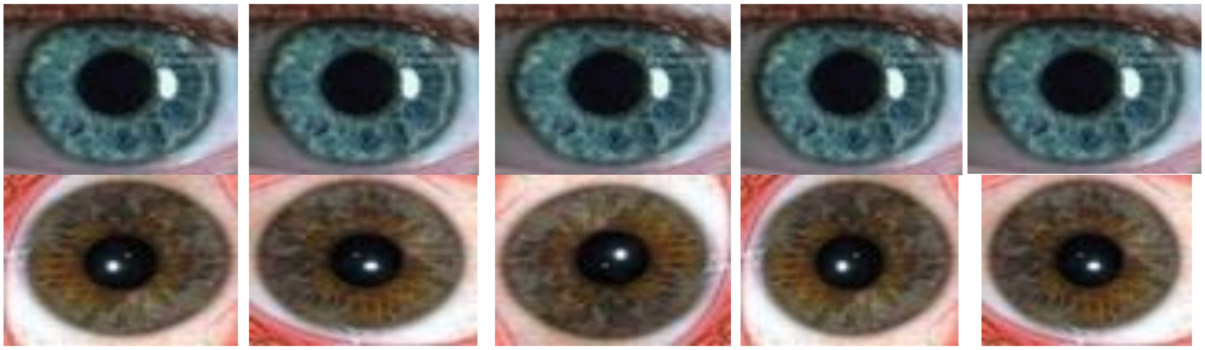


Fig.2: 10 different iris image samples of two persons each with 5 samples

5 Experimental Results

The implementation of PSO and ICA algorithms of the suggested iris recognition system was implemented using MATHLAB 2013. Mean Square Error (MSE), peak signal to noise ratio (PSNR) and recognition rate (RR) were used to evaluate the performance of PSO based ICA model for iris recognition system. Many experiments were conducted for the suggested iris recognition based on the selected 100 images of 10 persons (each with 10 images).

In the first four experiments, the feature extraction process is achieved using ICA with feature vector length equal to 64, 128, 256 and 512 respectively. And PSO is executed with swarm size equal 80 and number of iterations equal 100. Table (1) shows the recognition rate, MSE and PSNR of these four experiments.

Table (1): PSO/ICA results
(swarm size=80 and no. of iteration=100)

ICA feature size	Reco.rate	MSE	PSNR
64	90%	0.0122	30
128	94%	0.0098	32
256	96%	0.0055	34
512	98%	0.0011	38

We can note from Table (1) that best results including high recognition rate, high PSNR and low MSE were obtained when selecting ICA feature vector length equal 512.

Another 4 experiments were conducted for PSO with ICA. The ICA feature extraction process is achieved using different feature vector lengths equal to 64, 128, 256 and 512 respectively. Then PSO is executed with swarm size equal 80 and number of iterations equal 200. Table (2) shows the recognition rate, MSE and PSNR of these four experiments.

Table (2): PSO/ICA results
(Swarm size=80 no. of iterations=200)

ICA feature size	Reco.rate	MSE	PSNR
64	89%	0.0132	29
128	93%	0.0105	31
256	95%	0.0064	33
512	97%	0.0025	37

We can note from Table (2) that best results were obtained when selecting ICA feature vector length equal 512. We can note from Table (1) and Table (2) that increasing number of PSO iterations from 100 to 200 will reduce the performance of iris recognition system (lower recognition rate and lower PSNR).

Another 4 experiments were conducted for PSO with different ICA feature vectors length equal to 64, 128, 256 and 512 respectively. Then PSO is executed with swarm size equal 40 and 100 iterations. Table (3) shows the recognition rate, MSE and PSNR of these four experiments with swarm size equal 40 and 100 iterations.

Table (3): PSO/ICA results
(Swarm size=40 no. of iterations=100)

ICA feature size	Reco.rate	MSE	PSNR
64	88%	0.0154	29
128	92%	0.0123	30
256	93%	0.0087	32
512	95%	0.0034	36

Also, we can note from Table (3) that best results were obtained when selecting ICA feature vector length equal 512.

Another 4 experiments were conducted for PSO with different ICA feature vectors length equal to 64, 128, 256 and 512 respectively. Then PSO is executed with swarm size equal 40 and 200 iterations. Table (4) shows the recognition rate, MSE and PSNR of these four experiments with swarm size equal 40 and 200 iterations.

Table (4): PSO/ICA results
(Swarm size=40 no. of iterations=200)

ICA feature size	Reco.rate	MSE	PSNR
64	87%	0.0165	28
128	91%	0.0123	30
256	92%	0.0089	31
512	94%	0.0064	34

From Table (4), best results were obtained when selecting ICA feature vector length equal 512. Another 4 experiments were conducted for PSO with different ICA feature vectors length equal to 64, 128, 256 and 512 respectively. Then PSO is executed with swarm size equal 20 and 100 iterations. Table (5) shows recognition rate, MSE and PSNR of these four experiments.

Table (5): PSO/ICA results
(Swarm size=20 no. of iterations=100)

ICA feature size	Reco.rate	MSE	PSNR
64	88%	0.0177	28
128	91%	0.0131	30
256	91%	0.0096	31
512	92%	0.0051	35

From Table (5), best results were obtained when selecting ICA feature vector length equal 512. Another 4 experiments were conducted for PSO with different ICA feature vectors length equal to 64, 128, 256 and 512 respectively. Then PSO is executed with swarm size equal 20 and 200 iterations. Table (6) shows the recognition rate, MSE and PSNR of these four experiments.

Table (6): PSO/ICA results
(Swarm size=20 no. of iterations=200)

ICA feature size	Reco.rate	MSE	PSNR
64	87%	0.0171	27
128	90%	0.0132	30
256	91%	0.0093	30
512	92%	0.0078	32

From Table (6), best results were obtained when selecting ICA feature vector length equal 512.

5.1 Effect of Swarm Size on PSNR

We can note from Table (1), Table (3) and Table (5) that, the effect of swarm size (80, 40 and 20) on PSNR can be determined by using constant value of number of iterations (100) and also constant value of ICA feature vector length (512) in the three experiments. Fig.3 shows that increasing swarm size will increase the PSNR of the reconstructed image.

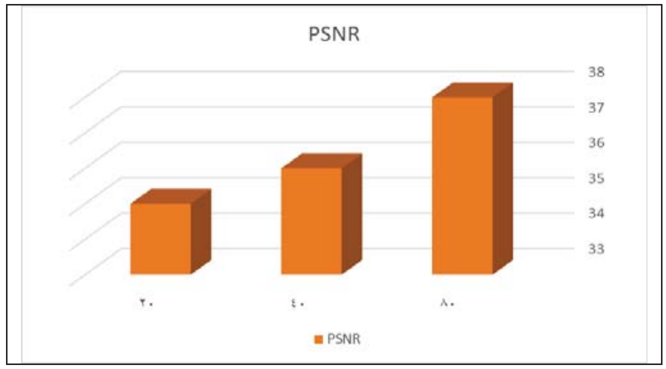


Fig.3: Swarm Size with PSNR

5.2 Effect of Swarm Size on Recognition Rate

We can note from Table (1), Table (3) and Table (5) that, the effect of swarm size (80, 40 and 20) on recognition rate can be determined by using constant value of number of iterations (100) and constant value of ICA feature vector length (512) in the three experiments. Fig.4 shows that increasing swarm size will increase the recognition rate.

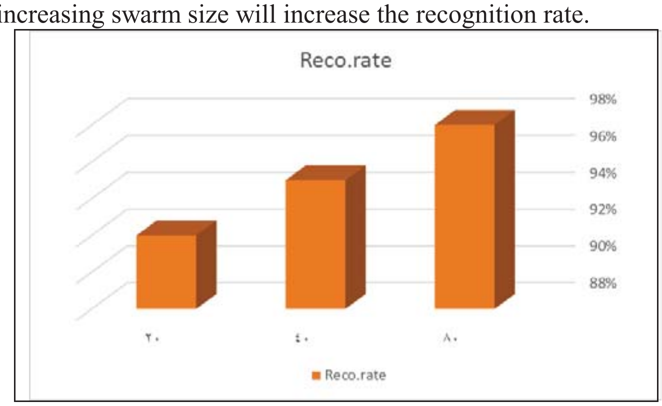


Fig.4: Swarm Size with Recognition rate

5.3 Effect of Swarm Size on MSE

We can note from Table (1), Table (3) and Table (5) that, the effect of swarm size (80, 40 and 20) on MSE can be determined by using constant number of iterations (100) and constant ICA feature vector length (512) in the three experiments. Fig.5 shows that increasing swarm size will decrease the MSE of the reconstructed image.

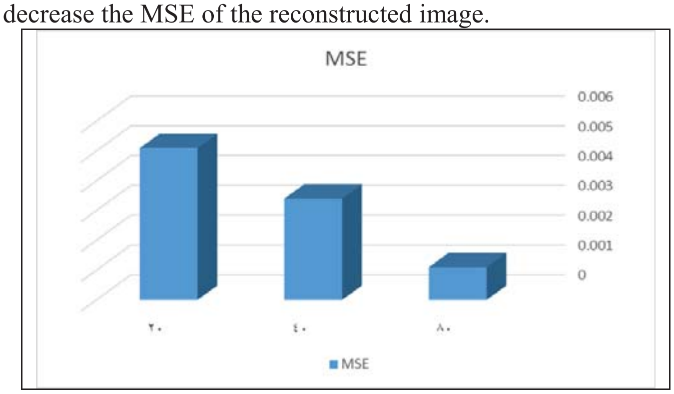


Fig.5: Swarm Size with MSE

5.3 Recognizing un Trained Images

In additional experiments, the performance of the suggested iris recognition system is evaluated also using un trained iris images (i.e. images were not used in training process). The feature extraction process is achieved using ICA with feature vector length equal to 64, 128, 256 and 512 respectively. And PSO is executed with swarm size equal 80 and number of iterations equal 100. Table (7) shows the recognition rate, MSE and PSNR of these experiments.

Table (7): PSO/ICA results /untrained (swarm size=80 and no. of iteration=100)

ICA feature size	Reco.rate	MSE	PSNR
64	47%	0.6725	13
128	48%	0.6235	15
256	51%	0.5874	17
512	53%	0.5673	19

We can note from Table (7) that the best results of recognizing untrained images (highest recognition rate, highest PSNR and lowest MSE) are achieved when using ICA feature size = 512.

6 Conclusions

Iris recognition system is suggested in this work depend on PSO algorithm. ICA algorithm is used as a feature reduction and extraction algorithm. To improve the performance of PSO. This system training and testing programs of the PSO and ICA algorithms were implemented using MATHLAB 2013. The training and testing samples of the suggested iris recognition system was taken from web site. Many experiments were conducted using different: swarm size (20, 40 and 80), number of iterations (100 and 200) and ICA feature vector length (64, 128, 256 and 512).

The results shows that the iris recognition performance (high PSNR, high recognition rate and lower MSE) will be increased when increasing the ICA feature vector length to 512 and also increasing the swarm size to 80 with decreasing the number of iterations to 100. Best obtained recognition rate is 98%, best PSNR is 38 and lower MSE is 0.0011, this is taken when selection 100 PSO iterations, ICA feature size to 512 and swarm size equal 80. The suggested iris recognition system has the ability to recognize un trained iris images but with performance lower than the performance of it when using trained iris images.

As a future work, other feature reduction and extraction algorithms may be used such as Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) and Linear Discriminative Analysis (LDA). These algorithms will be used separately with the PSO algorithm. Comparisons between the results of each algorithm.

7 References

[1] P.Chitte, et al. Iris Recognition System Using ICA, PCA, Daugman's Rubber Sheet Model Together, International Journal of Computer Technology and Electronics Engineering (IJCTEE) Vol.2, Issue.1, pp:16-23.

[2] Kazuyuki Miyazawa, et al., A Phase-Based Iris Recognition Algorithm, D. Zhang and A.K. Jain (Eds.): ICB 2006, LNCS 3832, pp: 356–365, Springer-Verlag Berlin Heidelberg, 2005.

[3] Kshamaraj G. and Sanjay G., Iris Recognition Using Independent Component Analysis, Int. Journal of Emerging Technology and Advanced Eng., Vol.2, Issue.7, pp:433-437, July2012.

[4] S. V. Sheela and P A Vijaya, Iris Recognition Methods – Survey, International Journal of Computer Applications, Vol.3, No.5, pp:19-25, Jun2010.

[5] M. Ezhilarasan et al., "Iris Recognition Based on Its Texture Patterns", International Journal on Computer Science and Engineering (IJCSE), Vol.2, No.9, pp:3071-3074, 2010.

[6] Libor Masek, Recognition of Human Iris Patterns for Biometric Identification, Bachelor of Eng. degree, School of Computer Science and SW Eng., University of Western Australia, 2003.

[7] H. Rahib Abiyev and K. Altunkaya, Personal Iris Recognition Using Neural Network, International Journal of Security and its Applications, Vol.2, No.2, 41-50, April 2008.

[8] A. S. Bhalchandra , et al.,Iris Recognition, Proceedings of World Academy of Science, Eng. and Technology, Vol.36, pp:1073-1078, Dec2008.

[9] D. Kresimir and G. Mislav, A Survey of Biometric Recognition Methods, 46th International Symposium Electronics in Marine, ELMAR, Zadar, Croatia, 16-18Jun2004.

[10] Kevin W. Bowyer, Image Understanding for Iris Biometrics: A Survey, Computer Vision and Image Understanding, Vol.110, No.2, pp:281-307, May 2008.

[11] P. W. Richard, Iris Recognition: An Emerging Biometric Technology, Proceedings of IEEE, Vol.85, No.9, Sep1997.

[12] V. Mayank, et. al, Comparison of iris Recognition Algorithms, IClSIP 2004, pp:354- 358, 2004.

[13] D. W. Patterson. Artificial Neural Networks, Theory and Applications, Singapore: Prentice Hall, 1996.

[14] L. Birgale and M. Kokare, A Survey on Iris Recognition, The IUP Journal of Electrical & Electronics Eng., Vol.II, No.4, 2009.

[15] Sajida K. and Sheikh Z., Iris Recognition: Existing Methods and Open Issues, Patterns, The Fourth International Conferences on Pervasive Patterns and Applications, IARIA, 2012.

[16] Qinghai Bai, Analysis of Particle Swarm Optimization Algorithm, Computer and Info Science, Vol.3, No.1, 2010.

[17] Kennedy, J. and Eberhart, R., Particle swarm optimization, Proceedings of IEEE International Conference on Neural Networks, Perth, WA, 1995.

[18] H. Kuo, J. Chang and C. Liu, Particle Swarm Optimization For Global Optimization Problems, Journal of Marine Science and Technology, Vol. 14, No. 3, pp:170-181, 2006.

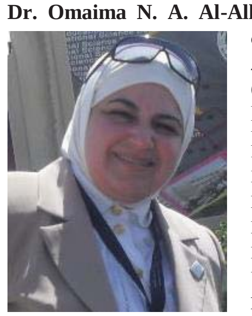
[19] Dian P.R., Siti M.S. and Siti S.Y., Particle Swarm Optimization: Technique, System and Challenges, International Journal of Computer Applications, Vol.14, No.1, pp:19-27, Jan2011.

[20] Satyobroto T., Mathematical Modelling and Applications of Particle Swarm Optimization, Master's Thesis, Mathematical Modelling and Simulation, School of Eng. at Blekinge Institute of Technology, Master of Science, Feb 2011.

[21] Voratas K., Comparison of Three Evolutionary Algorithms: GA, PSO, and DE, Industrial Engineering & Management Systems, Vol.11, No.3, pp.215-223, Sep2012.

[22] Daniel Bratton and James Kennedy, Defining a Standard for Particle Swarm Optimization, Proceedings of the 2007 IEEE Swarm Intelligence Symposium, SIS, 2007.
[23] Davoud S. and Ellips M., Particle Swarm Optimization Methods, Taxonomy and Applications, Int. Journal of Computer Theory and Eng., Vol.1, No.5, pp:486-502, 2009.
[24] M. Peyvandi, M. Zafarani and E. Nasr, Comparison of Particle Swarm Optimization and the Genetic Algorithm in the Improvement of Power System Stability by an SSSC-based Controller, Journal of Electrical Engineering & Technology Vol. 6, No. 2, pp:182-191, 2011.
[25] Bijayalaxmi Panda, Soumya Sahoo, Sovan Kumar Patnaik, A Comparative Study of Hard and Soft Clustering Using Swarm Optimization, International Journal of Scientific & Engineering Research, Vol.4, Issue.10, pp:785-790, Oct2013.
[26] Rania H., Babak C., Olivier D., A Copmarison Of Particle Swarm Optimization and the Genetic Algorithm, American Institute of Aeronautics and Astronautics, 46th Aiaa/Structures, Structural Dynamics And Materials Conference, 2005.
[27] Hugo A. Perlin,, et al. Particle Swarm Optimization for Object Recognition in Computer Vision, N.T. Nguyen et al. (Eds.), pp:11–21, Springer-Verlag Berlin Heidelberg 2008.
[28] Ziyad Al-Ta'i and Omer Abd Al-Hameed, Comparison between PSO and Firefly Algorithms in Fingerprint Authentication, International Journal of Eng. and Innovative Technology (IJEIT) Vol.3, Issue.1, pp:421-425, July 2013.
[29] Majida A. and Hamid A. Alasadi, Simplifying Handwritten Characters Recognition Using a Particle Swarm Optimization Approach, European Academic Research, Vol. I, Issue.5, pp:535-552, August 2013.
[30] R. M. Ramadan and R. F. AbdelKader, Face Recognition Using Particle Swarm Optimization-Based Selected Features, International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol.2, No.2, pp:51-66, Jun2009.
[31] P.V. Shinde, B.L. Gunjal and R. G. Ghule, Face Recognition Using Particle Swarm Optimization, Emerging Trends in Computer Science and Information Technology - (ETCSIT2012) Proceedings published in International Journal of Computer Applications (IJCA), pp:11-13, 2012.
[32] M. Arunkumar and S. Valarmathy, Palmprint And Face Based Multimodal Recognition Using Pso Dependent Feature Level Fusion, Journal of Theoretical and Applied Information Technology,Vol.57, No.3, pp:337-346, 30Nov 2013.
[33] K. Krishneswari and S. Arumugam, Intramodal Feature Fusion Based on PSO For Palmprint Authentication, Int Journal on Image and Video Proc., Vol.2, Issue.4, pp:435-440, May 2012.
[34] Ola M. Aly, et al., A Multimodal Biometric Recognition system using feature fusion based on PSO, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue.11, pp: 4336-4343, Nov2013.
[35] Ola M. Aly, et al., An Adaptive Multimodal Biometrics System using PSO, International Journal of Advanced Computer Science and Applications, Vol. 4, No.7, pp:158-165,2013
[36] S. Ivekovic, E. Trucco and Y. R. Petillot, Human Body Pose Estimation With Particle Swarm Optimisation, Evolutionary Computation, Vol.16, No.4, pp:509-528.
[37] Sruthy Sebastian, Activity Based Person Identification Using Particle Swarm Optimization Algorithm, International Journal of Computer Science and Mobile Computing, Vol.2, Issue.7, pp:1-6, Jul2013.
[38] Payam Saisan and Swarup Medasani and Yuri Owechko, Multi-View Classifier Swarms for Pedestrian Detection and Tracking, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR Workshops, San Diego, CA, USA, 25-25Jun2005.
[39] N.F. Shaikh, el al. Hybrid Optimization Method To Improve Iris Recognition System, International Journal For Technological Research in Eng., Vol.1, Issue.9, pp:675-680, May2014
[40] Nuzhat Faiz Shaikh, and D. D. Doye, Improving the Accuracy of Iris Recognition System using Neural Network and Particle Swarm Optimization, International Journal of Computer Applications, Vol79, No3, pp:1-8, Oct 2013.
[41] T. Hastie, R. Tibshirani, and J. H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction. New York: Springer-Verlag, 2001.
[42] Ewa Snitkowska and Włodzimierz Kasprzak, Independent component analysis in angiography images, Archives of Control Sciences, Vol.16, No. 3, pp:247–260, 2006.
[43] Ganesh R. Naik and Dinesh K Kumar, An Overview of Independent Component Analysis and Its Applications, Informatica, 35, pp:63–81, 2011.
[44] Wikipedia, Independent component analysis, http://en.wikipedia.org/wiki/Independent_component_ analysis
[45] Vivekanand D., Natalia S. and Gamal F., Performance Evaluation of Iris Based Recognition System Implementing PCA and ICA Encoding Techniques, Proc. SPIE 5779, Biometric Technology for Human Identification II, Vol. 5779, pp:1-8, 28 March 2005.
[46] Yoshinori S., Seiichi O. and Manabu K., Feature Extraction Using Supervised Independent Component Analysis By Maximizing Class Distance, ICONIP '02. Proceedings of the 9th International Conference on Neural Information Processing, Vol.5, pp:2502 – 2506, 18-22 Nov2002.
[47] Kyungim B., el al., PCA vs. ICA: A comparison on the FERET data set.
[48] Bruce Draper, et al., Recognizing faces with PCA and ICA, Computer Vision and Image Understanding 91, pp:115–137, 2003.
[49] Chengjun Liu and Harry Wechsler, Comparative Assessment of Independent Component Analysis (ICA) for Face Recognition, the Second International Conference on Audio- and Video-based Biometric Person Authentication, AVBPA,Washington USA, 22-24March 1999.
[50] A. Hyvärinen, Survey on independent component analysis. Neural Computing Surveys, 2, pp|:94-128, 1999.
[51] A. Hyvärinen, J. Karhunen And E. Oja: Independent component analysis. John Wiley & Sons, New York, 2001.

Dr. Omaima N. A. Al-Allaf is an Assistant Professor at Basic Sciences department, Faculty of Sciences and IT, AL-Zaytoonah University of Jordan, Amman, Jordan (from 15 Feb. 2009- to present). She received her B.Sc. in Computer Science from Dept. of CS, Faculty of Sciences, University of Mosul, Mosul, Iraq (1990-1994), and received her M.Sc. from Dept. of CS, Faculty of Computer & Mathematical Sciences, University of Mosul, Mosul, Iraq (1997-1999). She worked as Teacher B in Applied Science University (Private), Amman, Jordan from 2000 to 2004. She received her PhD in CIS from the Department of CIS, AABFS- Amman, Jordan (2004-2008). During 2004-2008, she worked as a part time lecturer in the AABFS. Her research interests are in the fields of: artificial intelligence, artificial neural networks, image processing, web engineering, software process models.



Optimal Design of SVC and Thyristor-Controlled Series Compensation Controller in Power System

O. Abedinia
Electrical Engineering Department
Semnan University
Semnan,
Iran

N. Amjady
Electrical Engineering Department
Semnan University
Semnan,
Iran

H. A. Shayanfar *
Department of Elec. Engineering
College of Tech. and Engineering
South Tehran Branch
Islamic Azad University
Tehran, Iran

oveis.abedinia@gmail.com, n_amjady@yahoo.com, hashayanfar@yahoo.com

Abstract— *In this paper, the optimal location and tuning parameters of Static Var Compensator (SVC), Proportional-Integral-Derivative (PID) stabilizer with low pass filter and Thyristor Controlled Series Compensator (TCSC) controllers using multi objective Honey Bee Mating Optimization (HBMO) to damp small signal oscillations in a multi machine power system has been implemented. Whereas, the performance of Flexible AC Transmission (FACT) devices highly depends upon its parameters and suitable position in the power network which is based on proposed algorithm. To demonstrate the validity of the proposed method, 10-machine 39-bus power system has been considered. Obtained results demonstrate the validity of proposed method.*

Keywords*:* FACTs devices, PID, Small signal stability, HBMO.

I. Introduction

Recently, power demand increases substantially and, on the other hand, the expansion of power generation and transmission is limited due to limited resources and environmental restrictions. So, the existing transmission systems should be utilized effectively by operating them closer to their thermal limits. This aim can be provided by reliable and high-speed Flexible AC Transmission System (FACTS) devices [1-3] such as static VAR compensator (SVC), Thyristor-Controlled Phase Shifter (TCPS), and Thyristor Controlled Series Capacitor (TCSC). FACTS are designed to enhance power system stability by increasing the system damping in addition to their primary functions such as voltage and power flow control.

The stability of power system is the core of power system security protection which is one of the most important problems researched by electrical engineers [1]. The fast-acting static excitation systems, used to improve transient stability limits, contribute strongly to the diminution of low frequency oscillation damping. The conventional lead-lag compensators have been widely used as the Power System Stabilizers (PSSs) [1-5]. However, the problem of PSS parameter tuning is a complex exercise. Beside of new control techniques with different structure, Proportional Integral Derivative (PID) type controller is still widely used for industrial applications [6-7]. Accordingly, it performs well for a wide class of process. Furthermore, they give robust performance for a wide range of operating conditions and easy to implement. Also, FACTs devices are too employed to enhance small signal stability which are based on high-voltage and high-speed power electronics devices [8-10]. This ability increase the controllability of power flows and voltages enhancing the utilization and stability of existing systems.

The optimal placement of FACTS controller in power system networks has been reported in scientific literatures based on different aspects. A method to obtain optimal location of TCSC has been suggested in [11] based on real power performance index and reduction of system VAR loss. In [10] optimal allocation of SVC using Genetic Algorithm (GA) has been investigated to achieve the optimal power flow (OPF) with lowest cost generation in power system. But the optimal allocations of SVC, PID and TCSC controllers using multi objective Honey Bee Mating Optimization (HBMO) have been

* Corresponding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

considered in this paper. Simulations are carried out on a typical multi-machine electric power systems; 10-machine 39-bus. Obtained simulation results confirm the enhances of small signal stability of proposed method in power system.

II. Problem Definition

The system dynamics of the synchronous machine can be expressed as a set of five first order linear differential equations given in Eqs. (1)–(5) [12].

$$\dot{\delta}_i = \omega_b(\omega_i - 1) \quad (1)$$

$$\dot{\omega}_i = (P_{mi} - P_{ei} - D_i(\omega_i - 1))/M_i \quad (2)$$

$$\dot{E}'_{qi} = (E_{fdi} - (x_{di} - x'_{di})i_{di} - E'_{qi})/T'_{doi} \quad (3)$$

$$\dot{E}_{fdi} = (K_{Ai}(v_{refi} - v_i + u_i) - E_{fdi})/T_{Ai} \quad (4)$$

$$T_{ei} = E'_{qi} i_{qi} - (x_{qi} - x'_{di}) i_{di} i_{qi} \quad (5)$$

Where, i_d and i_q are d-q components of armature current. E_{fd}, E'_d and E'_q are voltage proportional to field voltage, damper winding flux and field flux, respectively. Also, T'_{d0} and T'_{q0} are d-axis and q-axis transient time constant, respectively. In this paper, the results obtained with a relatively large power system which is the New England 10 machine 39 bus power system.

A. PID Stabilizer

The operating function of a PID is to produce a proper torque on the rotor of the machine involved in such a way that the phase lag between the exciter input and the machine electrical torque is compensated. The supplementary stabilizing signal considered is one proportional to speed. A widely speed based used PID is considered throughout the study [6]. The transfer function of the ith PID is:

$$U_i = \frac{ST_w}{1+ST_w}.(K_P + \frac{K_I}{S} + \frac{K_D}{1+T_D S})\Delta\omega_i(s) \quad (6)$$

Where $\Delta\omega_i$ is the deviation in speed from the synchronous speed. The value of the time constant, T_w is usually not critical and it can range from 0.5 to 20 s. The stabilizer itself mainly consists of two lead-lag filters as shown in Fig. 1. The parameters of the damping controllers for the purpose of simultaneous coordinated design are obtained using the multi objective HBMO algorithm. Many input signals have been proposed for the FACTS to damp the inter-area mode for this system. Signals which carry invaluable information about the inter-area mode can be considered as the input signals.

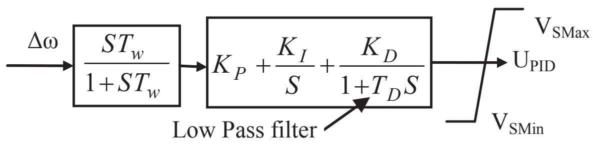


Figure 1. Structure of PID stabilizer

B. TCSC Modeling

The series connection scheme allows the power flow to be influenced through changing the effective admittance linking two buses, and is a method of improving transient stability limits and increasing transfer capabilities [13]. The transfer function pattern of a TCSC controller [14] has been given in Fig. 2.

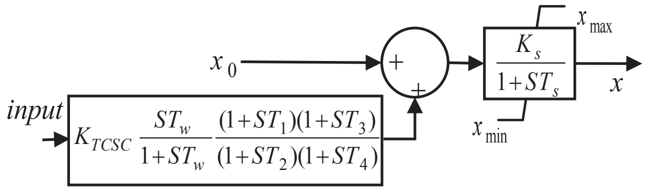


Figure 2. structure of TCSC based controller

This block may be considered as a lead-lag compensator. It comprises gain block, signal-washout block and two stages of lead-lag compensator. Where, X_0 is the impedence reference of TCSC. The X is the output reactance of TCSC. Time T_1 is a measurement time constant and T_w is the washout time constant.

C. SVC Modeling

Figure 3 shows the structure of an SVC model with a lead–lag compensator. The susceptance of the SVC, B, can be defined by:

$$\rho B = \frac{1}{T_s}(K_s(B_{ref} - u_{SVC}) - B) \quad (7)$$

Where, B_{ref}, K_s and T_s are the reference susceptance, gain and time constant for SVC device. As given in Fig. 3, a lead–lag controller is considered in the feedback loop to create the SVC stabilizing signal u_{SVC}.

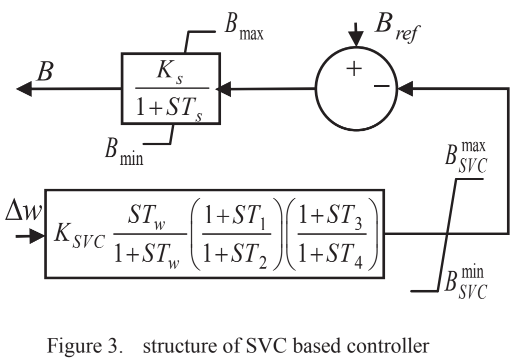


Figure 3. structure of SVC based controller

III. Multi Objective Honey Bee Mating Optimization

The honey bee is a social insect that can survive only as a member of a community, or colony. This means that they tend to live in colonies while all the individuals are the same family. In the more highly organized societies there is a division of labor in which individuals carry out particular duties. In fact, a colony consists of a queen and several hundred drones, 30,000 to 80,000 workers and broods in the active season. Each bee undertakes sequences of actions which unfold according to genetic, ecological and social condition of the colony [15]. The queen is the most important member of the hive because she is the one that keeps the hive going by producing new queen and worker bees and any colony maybe contain one or much queen in it life′s. Drones' role is to mate with the queen. In the marriage process, the queen(s) mate during their mating flights far from the nest [16]. In each mating, sperm reaches the spermatheca and accumulates there to form the genetic pool of the colony. The queen's size of spermatheca number equals to the maximum number of mating of the queen in a single mating flight is determined. When the mate be successful, the genotype of the drone is stored. In start the flight, the queen is initialized with some energy content and returns to her nest when her energy is within some threshold from zero or when her spermatheca is full. A drone's mate probabilistically is [17]:

$$P_{rob}(Q,D) = e^{-(\Delta f)/(S(t))} \quad (8)$$

Where,

Prob (Q, D) = The probability of adding the sperm of drone *D* to the spermatheca of queen *Q*

Δ(f) = The absolute difference between the fitness of *D* and the fitness of *Q (i.e., f (Q))*

S(t) = The speed of the queen at time *t*

After each transition in space, the queen's speed, and energy, decay using the following equations:

$$S(t+1) = \alpha \times S(t)(2), \quad \alpha \in [0,1] \quad (9)$$
$$E(t+1) = E(t) - \gamma$$

γ = The amount of energy reduction after each transition. The flowchart of Classic HBMO is presented in "Fig. 4", [14].

Thus, HBMO algorithm may be constructed with the following five main stages [13]:

- The algorithm starts with the mating–flight, where a queen (best solution) selects drones probabilistically to form the spermatheca (list of drones). A drone is then selected from the list at random for the creation of broods.
- Creation of new broods by crossoverring the drones' genotypes with the queen's.
- Use of workers (heuristics) to conduct local search on broods (trial solutions).
- Adaptation of workers' fitness based on the amount of improvement achieved on broods.
- Replacement of weaker queens by fitter broods.

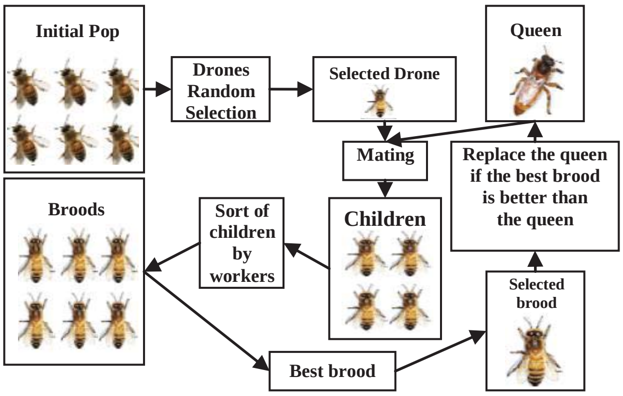


Figure 4. The Classic HBMO technique

A. Fuzzy Decision in Multi Objective HBMO

Usually, a membership function for each of the objective functions is defined by the experiences and intuitive knowledge of the decision maker. In this work, a simple linear membership function was considered for each of the objective functions. The membership function is defined as:

$$FDM_i = \begin{cases} 0 & ,\mu_i \le 0 \\ \dfrac{f_i^{\max} - f_i}{f_i^{\max} - f_i^{\min}} & ,0 < \mu_i < 1 \Rightarrow \mu_i = \dfrac{f_i^{\max} - f_i}{f_i^{\max} - f_i^{\min}} \\ 1 & ,\mu_i \ge 1 \end{cases} \quad (10)$$

Where f_i^{min} and f_i^{max} are the maximum and minimum values of the i^{th} objective function, respectively. For each non-dominated solution k, the normalized membership function FDM^k is calculated as:

$$FDM^k = \left(\sum_{i=1}^{N_{obj}} FDM_i^k \right) \Bigg/ \left(\sum_{j=1}^{M} \sum_{i=1}^{N_{obj}} FDM_i^j \right) \quad (11)$$

Where M is the number of non-dominated solutions, and N_{obj} is the number of objective functions.

IV. Numerical Results

A multi objective problem is formulated to optimize a composite set of objective functions comprising the damping factor, and the damping ratio of the lightly damped electromechanical modes, and the effectiveness of the suggested technique is confirmed through eigenvalue analysis and nonlinear simulation results. The simulation operated with multi objective HBMO algorithm and the objective functions for optimization as follow:

$$J_1 = \sum_{j=1}^{N_p} \sum_{i=1}^{N_g} \int_0^{t_{sim}} t.(|\Delta\omega_{ij}|).dt \qquad (12)$$

$$J_2 = \sum_{j=1}^{N_p} \sum_{i=1}^{N_g} \max_{i,j}[\mathrm{Re}(\lambda_{i,j}) - \min\{-\zeta\,|\,\mathrm{Im}(\lambda_{i,j})\,|, \alpha\}] \qquad (13)$$

Where, N_P, N_g, t_{sim}, λ and ζ are number of operating condition, number of generators, the time of simulation, the i^{th} eigenvalue of the system at an operating point and the desired minimum damping, respectively. The optimal location and tuning parameters problem can be formulated as the following constrained optimization problem, where the constraints are the PID, TCSC and SVC parameters bounds. The optimization Problem can be stated as:

$$\text{Minimize } J \text{ Subject to}:$$
$$K^{\min} \le K_{PID/TCSC/SVC} \le K^{\max} \qquad (14)$$
$$T_i^{\min} \le T_{i\,(PID/TCSC/SVC)} \le T_i^{\max}, i = 1, ..., 4$$

Typical ranges of the optimized parameters are [0.01-20] for $K_{PID/TCSC/SVC}$ and [0.01-1] for T_1 to T_4. Different operating conditions are analyzed for the New England system, as given in Table 1.

TABLE I. OPERATING CONDITIONS.

Conditions	Characteristics
1	Base case (normal operation)
2	Lines out: 1-2
3	Line out: 8-9
4	Increase 20% load to bus 17
5	Lines out: 46-49, Load increase 25% : 20, 21 generation increase 20%: G_9

It should be noted that proposed algorithm is run several times. The initial colony is produced randomly for each drone and is kept within a typical range. Figure 5 shows the trend evaluating process. The optimum parameters are given in Table 2. To demonstrate performance robustness of the proposed method, two performance indices: the Integral of the Time multiplied Absolute value of the Error (ITAE) and Figure of Demerit (FD) based on the system performance characteristics are defined as

$$ITAE = 100 \times \sum_{i=1}^{N_G} \int_0^{t_{sim}} t.(|\Delta\omega_i|).dt \qquad (15)$$

$$FD = \frac{1}{N_G} \sum_{i=1}^{N_G} ((600 \times OS_i)^2 + (8000 \times US_i)^2 + 0.01 \times T_{s,i}^2) \qquad (16)$$

Where, Overshoot (OS), Undershoot (US) and settling time of rotor angle deviation of machine is considered for evaluation of the FD. It is worth mentioning that the lower value of these indices is, the better the system response in terms of time domain characteristics.

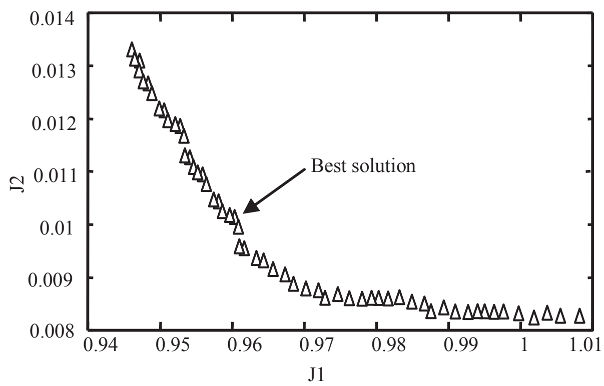


Figure 5. Fitness convergence with proposed algorithm.

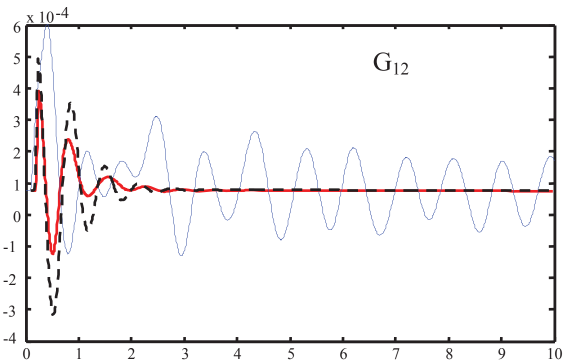
TABLE II. OPTIMAL VALUE FOR SVC, TCSC AND PID.

Type	Loc	K	T_1	T_2	T_3	T_4	
SVC	25	19.12	0.79	0.09	0.78	0.08	
TCSC	26#27	17.43	0.54	0.26	0.48	0.21	
PID	Loc	K_P	K_I	K_D	Loc	K_P	K_I	K_D
	1	13.23	11.43	5.43	9	13.43	12.43	6.54
	3	13.54	14.31	3.22	10	18.34	8.98	3.22
	5	18.32	12.74	3.90	12	13.52	9.54	1.22
	6	18.45	8.33	3.21	13	11.24	11.23	3.23
	7	12.32	13.54	2.12	15	17.87	12.43	1.43
	8	17.33	13.12	3.11	16	19.23	10.32	3.44

The results of the proposed multi objective based designed PID, SVC and TCSC under transient conditions is verified by applying disturbance and fault clearing sequence under different operating conditions based tuned them with mentioned objective functions. The following types of disturbances have been considered.

- *Scenario 1:* the three lines (16#17, 1#2 and 25#26) are out of service, assuming also that the nonlinear time domain simulations were carried out for a three phase-fault, with duration of 100 ms on the line 25#60. The speed deviations of generators under the proposed fault are shown in Fig. 6.

By considering to fig. 6, it can be said that the proposed method could provide low overshoot, undershoot as well as settling time in comparison with other techniques.



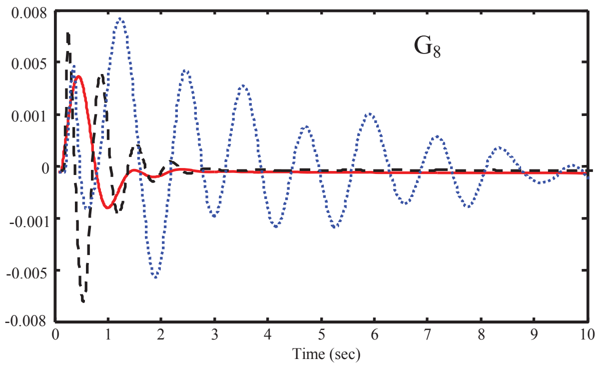


Figure 6. Speed variations of G8 and G12 under scenario I; solid (PID/SVC/TCSC), dashed (only with PID), dotted (with TCSC and SVC)

- *Scenario 2:* the two lines (16#17 and 25#26) are out of service and three-phase fault is applied at the same above mentioned location in scenario 1, assuming also that the variations of +30% in all load levels were used. The speed deviations of generators under the proposed fault are shown in Fig. 7.

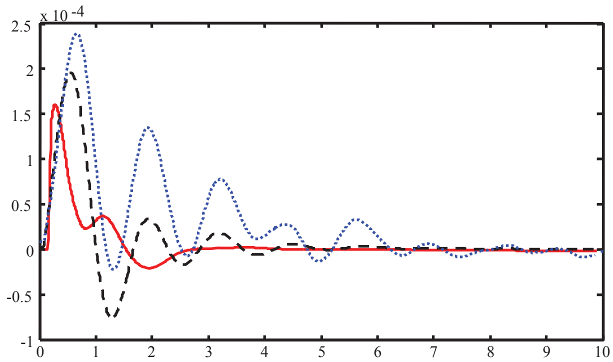


Figure 7. Speed variations of G_{15} under scenario II; solid (PID/SVC/TCSC), dashed (only with PID), dotted (with TCSC and SVC).

Numerical results of the system performance for different loading conditions are shown in Fig 8. It is worth mentioning that the lower the value of these indexes is, the better the system answer in terms of time domain characteristics. It is clear that the values of the power system performances with the proposed strategy are smaller compared when only PID or TCSC/SVC is installed. This shows that the OS, US, settling time and speed deviations of all generators are greatly reduced by applying the proposed multi objective HBMO algorithm based tuned PID, SVC and TCSC. Moreover, the nature of critical (Table 3) eigenvalue and time response analysis reveal that the proposed controller is more superior than the uncoordinated TCSC or SVC damping controller and PID stabilizers to improve the small signal oscillation problem even during critical loading. Also, the convergence of proposed method has been presented in Fig. 9.

TABLE III. CRITICALSWING MODES

SVC and TCSC		PID, TCSC and SVC	
Swing modes	Damping ratio	Swing modes	Damping ratio
Line outage (#13-14)			
-0.645±6.534i	0.982	-0.956±4.736i	0.1978
Load increase (20% more than nominal value)			
-0.756±6.032i	0.124	-0.847±4.021i	0.206

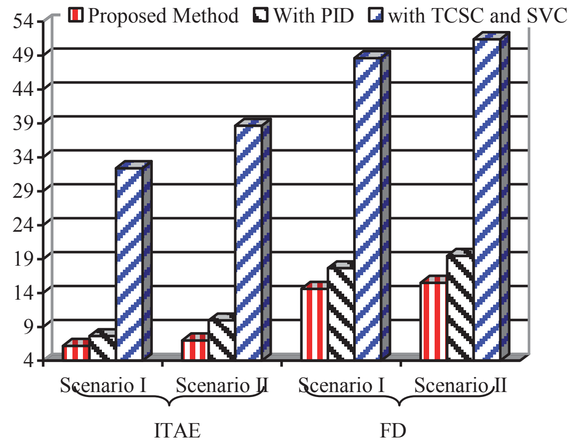


Figure 8. Values of performance indexes.

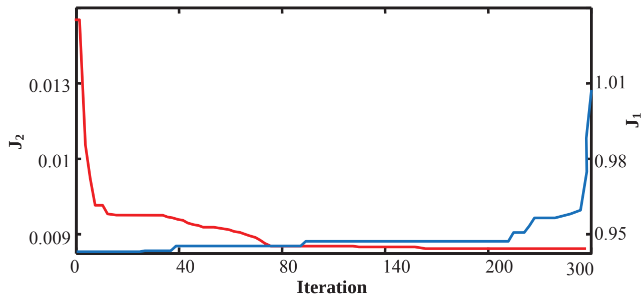


Figure 9. Convergence of objective functions

V. Conclusions

In this paper, the multi objective HBMO is implemented over an optimization problem for finding the best location and the parameters of coordinated PID stabilizers, SVC and TCSC controller simultaneously. Also, fuzzy decision making approach is proposed to obtain the best Pareto optimal location and settings of the FACTS controller among the Pareto optimal solutions. The proposed method has been applied over 10-machine 39 bus power system in different load conditions. Also, to demonstrate the validity of proposed method simulation results compared with only PID and only TCSC/SVC controller. Obtained results proofs the superiority of proposed method.

References

[1] V. A. Preethi, S. Muralidharan, S. Rajasekar, "Application of Genetic Algorithm to Power System Voltage Stability Enhancement Using Facts Devices", International Conference on Recent Advancements in Electrical, Electronics and Control Engineering, pp. 333-338, 2011.

[2] M. Eslami, H. Shareef, M. Khajehzadeh," Optimal design of damping controllers using a new hybrid artificial bee colony algorithm," Int J Electr Power Energy Syst, vol. 52, pp. 42–54, 2013.

[3] M. Eslami, H. Shareef, A. Mohamed, M. Khajehzadeh," Gravitational search algorithm for coordinated design of PSS and TCSC as damping controller," J Cent South Univ Technol, Vol. 19, pp. 923–32, 2012.

[4] P. He, F. Wenc, G. Ledwich, Y. Xue, K. Wang," Effects of various power system stabilizers on improving power system dynamic performance," Int J Electr Power Energy Syst, Vol. 46, pp. 175–183, 2013.

[5] S. K. Wang," A Novel Objective Function and Algorithm for Optimal PSS Parameter Design in a Multi-Machine Power System," IEEE Trans on Power Sys, Vol. 28, No. 1, pp. 522- 531, 2013.

[6] M. Gad, P. Shinde, S. U. Kulkarni, "Optimal Location of TCSC by Sensitivity methods" International Journal of Computational Engineering Research, Vol. 2, No. 6, 2012.

[7] K. Balamurugan, G. Suganya, N. Manojkumar, " Power System Dynamics By Series Connected FACTS Controllers (TCSC)", International Conference on Power, Energy and Control (ICPEC), pp 98-103, 2013.

[8] Y. Lu, A. Abur, "Static Security Enhancement via Optimal Utilization of Thyristor-Controlled Series Capacitor", IEEE Transaction on Power Systems, Vol. 17, No.2, pp. 324-329, 2002.

[9] AD. Rosso, CA. Caizares, VM. Doa," A study of TCSC controller design for power system stability improvement," IEEE Trans Power Syst, Vol. 18, No. 4, pp. 1487–96, 2003.

[10] El MM. Metwally, El AA. Emary, Bendary El FM, Mosaad MI. Optimal allocation of FACTS devices in power system using genetic algorithms. In: IEEE conference, MEPCON, pp. 4-1, 2008.

[11] H. Besharat, SA. Taher," Congestion management by determining optimal location of TCSC in deregulated power systems," Int J Electr Power Energy Syst, vol. 30, no. 10, pp. 563–624, 2008.

[12] N. Acharya, N. Mithulananthan, "Locating Series FACTS devices for Congestion Management in Deregulated Electricity Markets", Electric Power Systems Research, Vol. 77, No.3-4, pp. 352-360, 2007.

[13] A. Jalilvand, M. R. Safari Tirtashi, "Design of Output Feedback Controller for PSS and TCSC by GA to improve the Damping of Power System Oscillations", IEEE International conference on power and energy, pp 178-182, 2010.

[14] N.G.Hingorani, L.Gyugyi "Understanding FACTS: Concept and Technology of Flexible AC Transmission system", IEEE PRESS, 2000. sStandard Publishers.

[15] A. Ghasemi, H.A. Shayanfar, Mohammad. S. Naderi, O. Abedinia, "Optimal placement and tuning of robust multi-machine PSS via HBMO", In: Proceedings of the international conference on artificial intelligence, Las Vegas, Nevada, pp: 1-6, USA, 2011.

[16] HA. Abbass, "Marriage in honey-bee optimization (MBO): a haplometrosis polygynous swarming approach", The Congress on Evolutionary Computation,vol. 1, pp: 207-214, 2001.

[17] O. Abedinia, Mohammad. S. Naderi, A. Ghasemi, "Robust LFC in deregulated environment: fuzzy PID using HBMO", Proceeding of the IEEE International Power & Energy Society Power Systems Conference and Exposition, Italy, Rome (EEEIC), pp: 74-77, 2011.

Biographies

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM'10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

Heidar Ali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical

Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Smart Grids, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 495 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Analysis of the Performance Improvement Obtained by a Genetic Algorithm-based Approach on a Hand Geometry Dataset

A. G. A. Silva[1], **I. A. M. Barbosa**[1], **M. V. P. Nascimento**[1], **T. G. Rego**[1], **L. V. Batista**[1]
[1]Informatics Center, Federal University of Paraíba, João Pessoa, Paraíba, Brazil

Abstract— *Biometric recognition by hand geometry has a large number of measurements that may be used for authentication. The higher number of attributes, the harder is to define the importance of each one. In this paper, we analyze the use of a Genetic Algorithm-based approach in improving Equal Error Rate (EER) performance for biometric authentication by hand geometry. We used an own data set of dorsal and palm images of hand in a controlled environment to validate our approach. As the best results, the genetic algorithm decreased the equal error rate up to 0% in the training set and 0.01% for the test set. Additionally, a relative improvement of 90.91% was achieved by GA in the best case for the test set.*

Keywords: genetic algorithms, hand geometry, palm, optimization, EER

1. Introduction

Reliable identification systems have become a key issue for applications that authenticate users. Traditional methods to establish user's identity include mechanisms based upon knowledge (e.g., passwords) or tokens (e.g., identification cards). However, such mechanisms may be lost, stolen or even manipulated to spoof the system. In such a context, biometrics rises as an alternative. [15].

The biometry allows the identification of individual based on anatomical and behavioral features. There are many examples of biometric traits used to recognize a person, e.g., fingerprint, handprint, hand geometry, hand veins, face, voice, and iris. Those features can be used alone or combined (multi-biometric). Because biometric identifiers cannot be easily misplaced, forged, or shared, they are considered more reliable for person recognition than a traditional token or knowledge-based methods. Thus, biometric recognition systems are increasingly being deployed in many government and civilian application [14].

For human hands, a large number of features may be extracted, such hand print, the pattern of hand veins and hand geometry. Hand geometry refers to features like the shape of the hand, size palm, length and width of the fingers [11]. Such process has some advantages when compared with other methods [16], including easy to use; low cost, requires only an average resolution camera (no special sensors are necessary); and low computational cost, allowing faster results.

In biometric systems, a matching algorithm is used to compare two templates and generates a score value to indicate the degree of similarity between the templates. Such score depends on factors and constant weights are generally assigned to each factor. In the most of cases, these weights are computed empirically or statically. The optimization of these weights can be a hard task for a large N-dimensional features space.

Genetic Algorithms (GA) are an approach to optimization based on the principle of natural selection of Charles Darwin. These algorithms input is an N-dimensional vector that will be optimized according to a fitness function. GAs proved to be quite successful in finding good solutions to such complex problems as the traveling salesman, the knapsack problem, large scheduling problems and others [5].

The most used method to compare biometric systems is Equal Error Rate (EER) [19] [20] [12] [10]. To compute EER, two values are necessary: the False Acceptance Rate (FAR) and the False Rejection Rate (FRR). The FAR is the probability that an impostor is falsely accepted as a genuine use, while the FRR is the probability that a genuine user is falsely rejected by the system. Thus, the EER is the point where FAR and FRR are equal. The lower EER, the better the system.

This work analyzes a Genetic Algorithm [8] approach in order to improve EER performance in a hand geometry data set. To validate our study, an own data set of hand images were used.

2. Related Works

In this section, we list some related works that may be useful to the reader. First, the work of John Holland [9] was the first to describe evolutionary algorithms. Such work provides a good background about Holland's goal of understanding the life adaptation as like it occurs in nature and the ways of developing systems based on these principles.

For a good theoretical foundation for evolutionary algorithms Back et al. [1] provides an overview of the three main branches of evolutionary algorithms (EA): evolution strategies, evolutionary programming, and genetic algorithms. In their work, certain characteristic components of EAs are considered: the representation scheme of object variables, mutation, recombination, and selection operators.

The paper [2] describes a method to optimal feature selection for a speech signal of people with unilateral vocal fold paralysis. The GA is used in order to find an optimal set of features that maximize the recognition rate of support vector machine classifier. The results show that entropy feature, in comparison with energy, demonstrates a more efficient description of such pathological voices and provides a valuable tool for clinical diagnosis of unilateral laryngeal paralysis.

Considering the importance of parameter optimization on biometric systems, the work by Goranin et al. [7] analyzes a GA application in such context. According to it, the use of evolutionary algorithms may ensure a qualitative increase of biometric system parameters, such as speed, error rate, and flexibility.

The work [17] presents an optimization approach for authentication of fingerprint biometric system. The GA described in the paper is used in order to find the set of parameters that optimize the equal error rate. In the best case,their work reached a relative improvement of 40% in the equal error rate.

In a biometric system based on hand geometry, the work of [13] present a fusion approach of palmprint and hand geometry features in a verification system. The data set of hand images is built without pegs and controlled illumination, only using a digital camera. The results show that when the fusion of features is used, the error rates achieve lower values than when the features works alone (each palmprint or hand geometry).

Finally, the closest work in comparison with ours is the work of [6]. Such work describes an approach for biometric recognition based on hand geometry. Different classification and training methods are applied to measure results. The database used in this work is the same of ours. Additionally, their results are competitive when compared to other state-of-the-art methods.

3. Genetic Algorithms

Genetic algorithms were proposed by [8] as a tool to find solutions to optimization problems in poorly understood large spaces. They are based on the genetic processes of biological organisms, especially on the principle of natural selection by Charles Darwin [4]. Although, this slogan seems to be slightly tautological in the natural environment, where fitness is defined as the ability to survive, it makes good sense in the world of optimization problems where fitness of a string is given as the value of the function to be optimized at the argument encoded by the string.

Typically, a genetic algorithm works on a population of individuals. Each individual is represented by one chromosome formed by a set of genes representing the parameters to be optimized. Some operations are realized in order to produce new generations of individuals based on their capability to generate good results: crossover, selection and mutation.

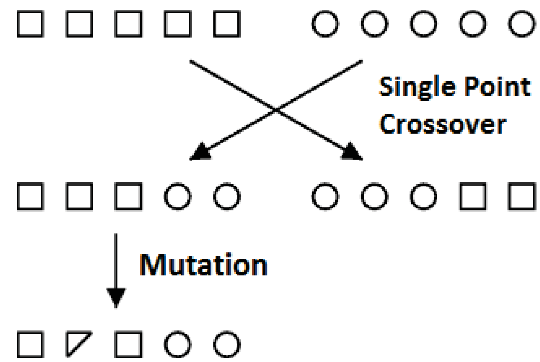


Fig. 1: Example of Crossover and Mutation operators

The crossover is the key operator to generate new individuals in the population. Inspired by the example of nature, crossover is intended to join the genetic material of chromosomes with a high fitness in order to produce even better individuals.

The selection operator is intended to implement the idea of "survival of the fittest". It basically determines which of the chromosomes in the current population is allowed to inherit their genetic material to the next generation.

The mutation operator should allow the GA to find solutions which contain genes values that are non-existent in the initial population. The parameter governing this operator is called mutation probability. Whereas the selection operator reduces the diversity in the population, the mutation operator increases it again. The higher the mutation probability, the smaller is the danger of premature convergence. A high mutation probability, however, transforms a GA into a pure random search algorithm, which is of course not the intention of this.

Let P be a random population of N chromossomes (x_1, x_2, ..., x_n) and *f(x)* a fitness function. The following pseudocode describes the steps of genetic algorithms.

1) Create a random population P of N chromosomes (candidate solutions for the problem).
2) Evaluate *f(x)* of each chromossome *x* in the population.
3) Generate a new population by repeating the following steps until the new population reaches population N:
    a) Select two parent chromosomes from the population, giving preference to highly fit chromosomes (high *f(x)* values). Automatically copy the fittest chromosome to the next generation.
    b) With a given crossover probability, crossover the parent chromosomes to form two new offspring. If no crossover was performed, offspring is an exact copy of parents.
    c) With a given mutation probability, randomly swap two genes in the offspring.
    d) Copy the new offspring into a new population.
4) Copy the newly generated population over the previous

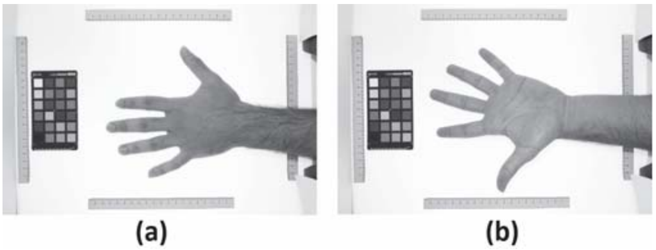


Fig. 2: Image samples of the data set used. (a) dorso (palm down). (b) palm (palm up).

(existing) population.

5) If the loop termination condition is satisfied, then stop and return the best solution in current population. Otherwise, go to Step 2.

4. Proposed Technique

In this paper, we use a GA-based approach to optimizing the EER performance of a hand geometry authentication algorithm. Detailed description of the database used and the methods developed are described in subsections below.

4.1 Database

The database for our work was the same described in [6]. The images were acquired using a device built with a negatoscope, a wooden box, and a DSLR camera. This device has lighting conditions controlled. Each individual places his hand inside the box through a hole in the bottom. The camera capture images 5184x3456 pixels in raw format. The Fig. 2 shows an two images from this data set.

This database is composed of 1200 images divided into 100 different class, where each class represents a person. Each person has 12 hand images, 7 dorsal images, and 5 palm images. Thus, the database has 700 dorsal images and 500 images with palm up.

For each hand image, we use eight measurements from each finger: area, perimeter, length, bottom width, convex area (the area of the smallest convex polygon that contains the finger), eccentricity and the axes of the ellipse that has the same normalized second central moments as the finger image. Of all the fingers, except the thumb, are extracted two angles between three segments that are used to determine the natural inclination of the fingers.

Six measurements are also extracted from hand images: area, perimeter, convex area, eccentricity and the axes of the ellipse (calculated similarly to the fingers). To sum up, there is a total of 54 features which are used as an attributes vector for classification.

For this work, we also extracted 31 more measurements for each palm up image: 6 width information for each finger and hand width. Thus, we use 85 measurements in total for each palm image. Tests were performed to compare the GA performance using 54 and 85 measurements.

4.2 GALib

The GALib library [18] was chosen as a framework to apply the genetic algorithm in this work. It is an open source library written in Java and very easy to use.

In our method, the genetic algorithm was used to compute the weights to optimize our matching algorithm (see section 4.3) concerning to decrease the EER. For this, the initial population of the genetic algorithm is a set of coefficients that represent the importance of each attribute to classification. These coefficients multiply the values of attributes giving importance to each one in the classification. The Equal Error Rate (EER) is used as a fitness function. The Coefficients that generate lower values of EER on classification are more indicated to next generations.

In our work, we have chosen the initial chromosome population of GA equals to 1,000 in order to do an exhaustive search on the search space. Moreover, we also defined 1,000 as the number of generations to be generated by GA and the crossover type was set to *uniform* because preliminary tests have shown this crossover type converges faster than the other two. All the other GALib parameters were left as default.

4.3 The Matching Algorithm

The matching algorithm of this work computes the score between two templates using the score function defined by Eq.(1). A template is defined as the set of measurements of a hand or palm image.

$$d_{1,2} = \sum_{i=0}^{N} w_i . \left| a_i^1 - a_i^2 \right| \quad (1)$$

Where a_i^1 and a_i^2 represent the *i*-th measurement from template 1 and 2, respectively; w_i represents the *i*-th multiplier coefficient; N is the number of measurements; and $d_{1,2}$ is the distance between templates 1 and 2. The lower the distance, the more similar templates 1 and 2.

The main goal of genetic algorithm in this work is to find all the weights $\{w_1, w_2, w_3, ..., w_N\}$ that minimize the EER performance for training set.

4.4 Validation

To validate our method, a cross-validation was performed in the database used. For hand templates, 4 templates of each class were used for training and 3 templates were used for test. For palm templates, 3 templates of each class were used for training and the 2 remaining templates were used for test. The relative performance improved by GA is also analyzed. Table 1 summarize the tests applied.

Each combination was performed three times in training and the best result was stored. The use of GA for test cases of both hand and palm templates means the matching algorithm was applied using the weights computed by GA in corresponding training set.

Table 1: Combination parameters for hand templates

Combination	Templates (Training)	Templates (Test)
1	1st,2nd,3rd,4th	5th,6th,7th
2	1st,2nd,3rd,5th	4th,6th,7th
3	1st,2nd,3rd,6th	4th,5th,7th
4	1st,2nd,3rd,7th	4th,5th,6th
5	1st,2nd,4th,5th	3rd,6th,7th
6	1st,2nd,4th,6th	3rd,5th,7th
7	1st,2nd,4th,7th	3rd,5th,6th
8	1st,2nd,5th,6th	3rd,4th,7th
9	1st,2nd,5th,7th	3rd,4th,6th
10	1st,2nd,6th,7th	3rd,4th,5th
11	1st,3rd,4th,5th	2nd,6th,7th
12	1st,3rd,4th,6th	2nd,5th,7th
13	1st,3rd,4th,7th	2nd,5th,6th
14	1st,3rd,5th,6th	2nd,4th,7th
15	1st,3rd,6th,7th	2nd,4th,5th
16	1st,3rd,5th,7th	2nd,4th,6th
17	1st,4th,5th,6th	2nd,3rd,7th
18	1st,4th,5th,7th	2nd,3rd,6th
19	1st,4th,6th,7th	2nd,3rd,5th
20	1st,5th,6th,7th	2nd,3rd,4th
21	2nd,3rd,4th,5th	1st,6th,7th
22	2nd,3rd,4th,6th	1st,5th,7th
23	2nd,3rd,4th,7th	1st,5th,6th
24	2nd,3rd,5th,6th	1st,4th,7th
25	2nd,3rd,5th,7th	1st,4th,6th
26	2nd,3rd,6th,7th	1st,4th,5th
27	2nd,4th,5th,6th	1st,3rd,7th
28	2nd,4th,5th,7th	1st,3rd,6th
29	2nd,4th,6th,7th	1st,3rd,5th
30	2nd,5th,6th,7th	1st,3rd,4th
31	3rd,4th,5th,6th	1st,2nd,7th
32	3rd,4th,5th,7th	1st,2nd,6th
33	3rd,4th,6th,7th	1st,2nd,5th
34	3rd,5th,6th,7th	1st,2nd,4th
35	4th,5th,6th,7th	1st,2nd,3rd

Table 2: Combination parameters for palm templates

Combination	Templates (Training)	Templates (Test)
1	1st,2nd,3rd	4th,5th
2	1st,2nd,4th	3rd,5th
3	1st,2nd,5th	3rd,4th
4	1st,3rd,4th	2nd,5th
5	1st,3rd,5th	2nd,4th
6	1st,4th,5th	2nd,3rd
7	2nd,3rd,4th	1st,5th
8	2nd,3rd,5th	1st,4th
9	2nd,4th,5th	1st,3rd
10	3rd,4th,5th	1st,2nd

5. Results and Discussions

To evaluate our method, 45 combinations of tests were performed on 1200 images (700 hands and 500 palms) divided into 100 classes. Furthermore, for each hand and palm template, 54 and 85 measures were extracted respectively. Subsets of each class are used for training and the remaining are used for test. Genetic algorithms are applied to analyze the EER improvement.

Figure 3 shows the results of the application of GA to all combinations performed in the training set of hand templates. As it can be clearly seen, the genetic algorithm improved the EER for all combinations. As depicted in Table 3, such improvement is at least 86.2694% (combination 32). In some cases (combinations 6, 16, 18, 19, 28, and 31), the improvement obtained by GA acquires 100.0%.

For test set of hand templates, the relative improvement of EER ranged from 25.2396% (combination 11) to 84.4985% (combination 2). Such results can be noticed in Table 3 and observed graphically in Fig. 4. Overall, the use of GA improved EER performance for all combinations of tests where GA is not applied.

Similar to hand templates, the application of GA achieved a significant improvement in EER performance on training sets of palm templates (see Fig. 5 and Table 4). In this respect, the performance improvement ranged from 33.99% (combination 5) to 98.51% (combination 8).

In Figure 6 and in Table 4 is possible to see the results accomplished by genetic algorithms in all test sets of palm templates. The best performance improvement were around 90.91% (combination 2). However, only in these tests there was a worsening of the EER performance in some cases (combinations, 1, 7, and 9). Figure 6 shows this worsening is lower than 1%, in practice, though.

6. Conclusion

This paper analyzes the EER performance improvement obtained by a GA-based approach. A cross-validation was performed on a database with high-quality images of palm down and palm up hands to evaluate your method. The genetic algorithm was used to optimize weights present in the matching algorithm in order to improve EER performance, thus, improving the authentication of system.

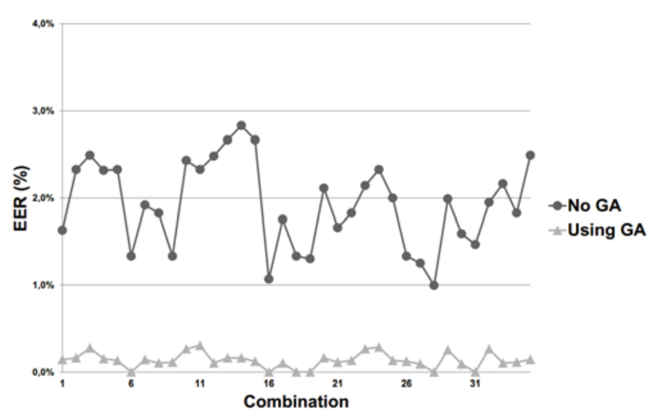


Fig. 3: Results of application of GA for all combinations performed on training set of hand templates.

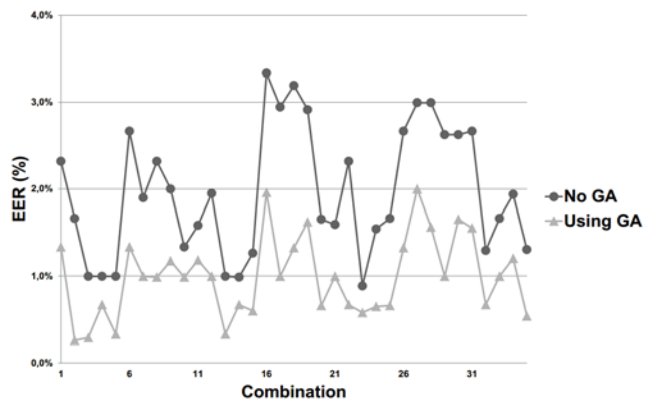


Fig. 4: Results of application of GA for all combinations performed on test set of hand templates.

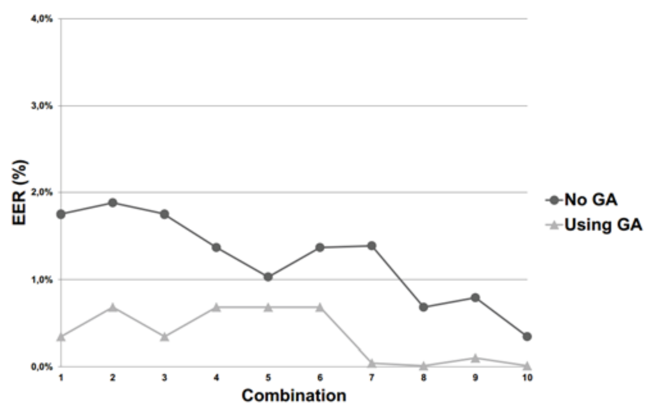


Fig. 5: Results of application of GA for all combinations performed on training set of palm templates.

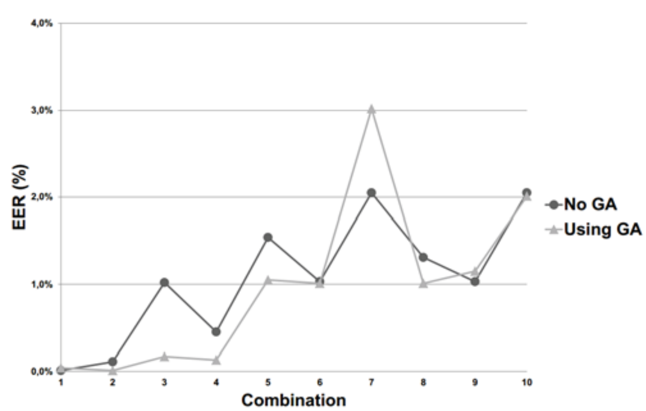


Fig. 6: Results of application of GA for all combinations performed on test set of palm templates.

Table 3: Relative improvement for training and test sets of hand templates.

Combination	Rel. Improv. (Training)	Rel. Improv. (Test)
1	91.176471%	42.483660%
2	92.950108%	84.498480%
3	88.855117%	71.065990%
4	93.355120%	32.994924%
5	94.251627%	67.005076%
6	100.000000%	50.000000%
7	92.500000%	47.606383%
8	94.329184%	57.516340%
9	91.477273%	41.414141%
10	88.992731%	26.136364%
11	86.767896%	25.239617%
12	95.829095%	48.963731%
13	93.844697%	67.005076%
14	94.201606%	32.307692%
15	95.359848%	52.800000%
16	100.000000%	41.212121%
17	94.100719%	66.209262%
18	100.000000%	58.544304%
19	100.000000%	44.367418%
20	92.234170%	60.244648%
21	93.161094%	37.460317%
22	92.669433%	71.241830%
23	87.514723%	62.857143%
24	87.635575%	58.032787%
25	93.308081%	60.486322%
26	90.719697%	50.378788%
27	92.540323%	33.220911%
28	100.000000%	47.892074%
29	87.055838%	62.115385%
30	94.126984%	37.115385%
31	100.000000%	41.856061%
32	86.269430%	48.437500%
33	95.215869%	40.121581%
34	93.775934%	38.020833%
35	94.224924%	58.914729%

The results show that our approach produces a significant improvement when GA is used. For the training set, all the 45 combinations had an improvement in EER performance. In the most of cases, the relative improvement was above 80%. For the test set, i.e. samples not used for GA training, the relative improvement has occurred in 42 of 45 cases.

For future works, we intend to test our approach with other databases to verify whether GA also improve their results. Furthermore, we also expect to test other evolutionary algorithms and optimization metaheuristics, as ant colony, particle swarm, and greedy randomized adaptive search procedure, comparing their results with the results presented in this paper.

Although hand geometry recognition is not usual yet, the experiments performed in this work can indicate evolutionary algorithms as a tool to improve the equal error rate and the quality of biometric systems.

References

[1] Thomas Bäck and Hans-Paul Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary computation*,

Table 4: Relative improvement for training and test sets of palm templates.

Combination	Rel. Improv. (Training)	Rel. Improv. (Test)
1	80.69%	-300.00%
2	64.08%	90.91%
3	80.69%	83.09%
4	50.74%	71.11%
5	33.99%	32.13%
6	50.74%	1.97%
7	97.10%	-47.04%
8	98.51%	23.17%
9	87.18%	-11.82%
10	97.02%	1.97%

1(1):1–23, 1993.

[2] Roozbeh Behroozmand and Farshad Almasganj. Optimal selection of wavelet-packet-based features using genetic algorithm in pathological assessment of patients' speech signal with unilateral vocal fold paralysis. *Computers in Biology and Medicine*, 37(4):474–485, 2007.

[3] José Bins and Bruce A Draper. Feature selection from huge feature sets. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 159–165. IEEE, 2001.

[4] Charles Darwin and Gillian Beer. *The origin of species*. Oxford University Press Oxford, 1951.

[5] Herbert Dawid. Genetic algorithms. In *Adaptive Learning by Genetic Algorithms*, volume 441 of *Lecture Notes in Economics and Mathematical Systems*, pages 37–60. Springer Berlin Heidelberg, 1996.

[6] Marcia V. P. do Nascimento, Leonardo V. Batista, Nicomedes Cavalcanti, et al. Comparative study of learning algorithms for recognition by hand geometry. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pages 423–428. IEEE, 2014.

[7] N Goranin and A Cenys. Evolutionary algorithms application analysis in biometric systems. *Journal of Engineering Science and Technology Review*, 3(1):70–79, 2010.

[8] John H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, 1975. second edition, 1992.

[9] John Holland. Outline for a logical theory of adaptive systems. *Journal of the Association of Computing Machinery*, 3, 1962.

[10] Anil Jain, Lin Hong, and Sharath Pankanti. Biometric identification. *Communications of the ACM*, 43(2):90–98, 2000.

[11] Anil K Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(1):4–20, 2004.

[12] Alisher Kholmatov and Berrin Yanikoglu. Identity authentication using improved online signature verification method. *Pattern recognition letters*, 26(15):2400–2408, 2005.

[13] Ajay Kumar, David CM Wong, Helen C Shen, and Anil K Jain. Personal verification using palmprint and hand geometry biometric. In *Audio-and Video-Based Biometric Person Authentication*, pages 668–678. Springer, 2003.

[14] Davide Maltoni, Dario Maio, Anil K. Jain, and Salil Prabhakar. *Handbook of Fingerprint Recognition*. Springer Publishing Company, Incorporated, 2nd edition, 2009.

[15] Arun A. Ross, Karthik Nandakumar, and Anil K. Jain. *Handbook of Multibiometrics (International Series on Biometrics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[16] Raul Sanchez-Reillo, Carmen Sanchez-Avila, and Ana Gonzalez-Marcos. Biometric identification through hand geometry measurements. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1168–1171, 2000.

[17] Tobias Scheidat, Andreas Engel, and Claus Vielhauer. Parameter optimization for biometric fingerprint recognition using genetic algorithms. In *Proceedings of the 8th Workshop on Multimedia and Security*, pages 130–134. ACM, 2006.

[18] Jeff Smith. Genetic algorithm library, 2003.

[19] Robert Snelick, Umut Uludag, Alan Mink, Mike Indovina, and Anil Jain. Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(3):450–455, 2005.

[20] Pim Tuyls, Anton HM Akkermans, Tom AM Kevenaar, Geert-Jan Schrijen, Asker M Bazen, and Raimond NJ Veldhuis. Practical biometric authentication with template protection. In *Audio-and Video-Based Biometric Person Authentication*, pages 436–446. Springer, 2005.

Reconfiguration of Radial Distribution Networks by Applying a Multi-objective Technique

R. Tapia-Juárez, E. Espinosa-Juárez
Faculty of Electrical Engineering, Universidad Michoacana de San Nicolás de Hidalgo
Morelia, México

Abstract -*In this paper, the reconfiguration of distribution networks in order to optimize several aspects of its operation is presented. The proposed methodology based on genetic algorithms finds the optimal reconfiguration for networks which have a radial topology, taking in consideration the expected occurrence of voltage sags reduction and also the real power losses. The developed functions of the genetic algorithm are described, and case studies are presented for the IEEE 33-bus test network and the IEEE 118-bus test system demonstrating the effectiveness of the implemented methodology.*

Keywords: distribution networks, multi-objective reconfiguration; genetic algorithms; voltage sags.

1 Introduction

The distribution systems require to supply electricity every day with more reliability, and with the quality parameters that require consumers. Furthermore, in the context of smart grids, the distribution network plays an important role together with the information systems, communications and control, in order to optimize the operation of the entire electrical system [1]-[4].

The reconfiguration of electrical systems is a technique that has been proposed for several authors in order to improve different aspects of the planning and operation of the electrical systems, and also, different methods have been applied to solve this optimization problem [5]-[8].

In this paper a methodology is presented for the reconfiguration of the distribution network for the purpose of improving an important aspect of power quality, the voltage sags, and simultaneously reduce losses, in order to make more efficient the operation of the distribution network. The proposed multi-objective methodology, previously reported in [9] is based on the application of genetic algorithms which include novel genetic operators developed to reduce the search space. In order to test the proposal, studies in the IEEE 33-bus and the IEEE 118-bus test systems are presented.

2 Formulation of the multi-objective problem

Next, the methodology proposed in [9] is presented in order to explain its application to the case studies in this paper.

The multi-objective optimization consists of optimizing a problem with two or more simultaneous objectives and finding the set of compromised solutions (trade-offs) which best solves the problem, considering specified constraints [10]-[12]. In this paper, the network reconfiguration problem consist on determining the state (on/off) of the switches of the electrical system which leads to a radial topology, with two objectives: to minimize the expected number of voltage sags with respect to specified values of voltage sags at system buses, and to minimize real power losses.

In order to formulate the reconfiguration problem, the following vectors are defined:

$$V_{ref} = \begin{bmatrix} v_{ref1} & v_{ref2} & v_{ref3} & \cdots & v_{refn} \end{bmatrix}$$
$$V_{reconf} = \begin{bmatrix} v_{reconf1} & v_{reconf2} & v_{reconf3} & \cdots & v_{reconf\ldots n} \end{bmatrix}$$
$$V_B = \begin{bmatrix} v_{B1} & v_{B2} & v_{B3} & \cdots & v_{B\ldots n} \end{bmatrix}$$

where V_{ref}, is the vector of reference values of voltage sags at buses, V_{reconf} is the vector of voltage sags at buses after reconfiguration, V_B is a binary vector where v_{Bi} is 1 if $v_{reconfi}>v_{ref\,i}$, otherwise, v_{Bi} is 0, and n is the number of system buses.

According to this, the multi-objective optimization problem is formulated as follows:

$$\min F(x) = (f_1(x), f_2(x)) \qquad (1)$$

Subject to:

$$\begin{gathered} M = n - n_f \\ V_{\min k} \le V_k \le V_{\max k} \\ P_{Gk} - P_{lk} - P_k(V,\theta) = 0 \\ Q_{Gk} - Q_{lk} - P_k(V,\theta) = 0 \end{gathered} \qquad (2)$$

where:

n_f number of feeders or sources
M number of radial net branches.
$f_1(x) = \sum_{k=1}^{n} V_B$
$f_2(x) = \sum_{km}^{n} [V_k^2 + V_m^2 - 2V_m^2 V_m^2 \cos(\theta_k - \theta_m)]$
V_k voltage magnitude at bus k
θ_k voltage phase angle at bus k
$V_{min\,k}$ minimum voltage at bus k
$V_{max\,k}$ maximum voltage at bus k
$P_{G\,k}$ active power injected by generator at bus k
$Q_{G\,k}$ reactive power injected by generator at bus k
$P_{l\,k}$ active load power at bus k
$Q_{l\,k}$ reactive load power at bus k
$P_k(V_k, \theta)$ active power injected at bus k
$Q_k(V_k, \theta)$ reactive power injected at bus k

In this work, the number of voltage sags is determined using the fault position method [13][14]. It can be observed that the radiality constraint is partially considered by (2); however, there are situations where even when (2) is satisfied, the system is not radial, so this constraint is further verified with a review of the system graph.

Ordinarily, the connection/disconnection of a transmission line is represented by using a single switch operation. According to this, the number of elements that can change their state is equal to the considered number of switches SN. For a generic system of N buses, with SN switches that can operate (open/close) without leaving any system areas de-energized, a vector S that contains only the number of disconnected lines is defined. For example, for network in Fig. 2 the correspondent vector is S=[15 21 26].

3 Implementation of the methodology by using genetic algorithms

In this paper, the solutions for optimizing the problem are restricted to solutions with radial topologies, and since (2) is a necessary constraint but not sufficient, this restriction is implicitly included in the genetic algorithm. The proposed GA uses crossover and mutation operators, designed to generate radial individuals. When an individual violates the radiality constraint, this potential solution is dismissed or applied to another round of genetic operations until the radiality restriction is met. This has the advantage of retaining the genetic material and using it again to create a new individual. Furthermore, this ensures that only feasible individuals will be evaluated on the fitness function.

In order to explain the proposed methodology, the basic distribution system shown in Fig. 1 is used. Data of this system is provided by [15].

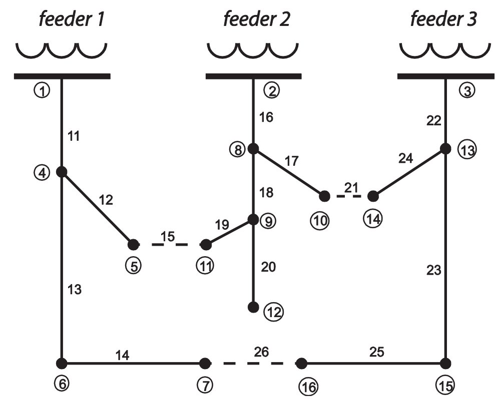


Fig 1. Distribution system.

3.1 Basic loops

In Fig. 2 the feeders are represented into a single bus with the purpose of avoiding a cycle to be formed between them. After a successful reconfiguration, feeders will return to their original position. It can be seen how each feeder supplied energy to each corresponding load.

For a generic system, the number PL of basic loops is defined b

$$PL = n_l - n - 1 \qquad (3)$$

where n_l is the total number of system lines.

Note that (3) also indicates the number of lines that must be out of operation in order to assure that the system has a radial configuration. In Fig. 2 the basic loops are shown and also the tie-lines which are normally open (looping branches).

Note that in a distribution system, in order to change the operating state of a tie-line from normally open to closed, the state of one line belonging to the corresponding basic loops must be changed to normally open, in order to maintain the radiality of the system.

For the system in Fig. 2, the basic loops are

Loop 1 = [11 12 16 18 19 15]

Loop 2 = [16 17 22 24 21]

Loop 3 = [11 13 14 22 23 25 26]

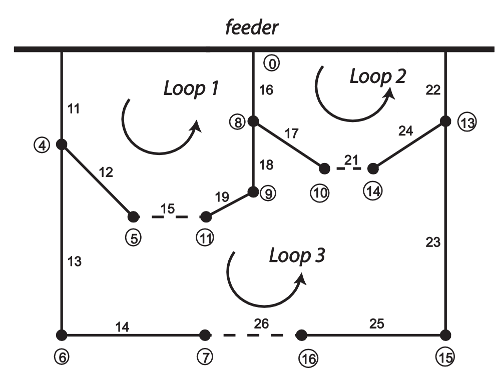


Fig 2. Basic distribution system.

Note that line 20 is not part of any basic loop, and its status change causes bus 12 to be left without energy or disconnected. Because of this, in Fig. 2 is not taken into account, but when the reconfiguration is performed, line 20 is incorporated into the generated topology.

3.2 Chromosome encoding

The reconfiguration encoding has been extensively studied in [16] where the most common encodings are mentioned. The encoding proposal in this paper is a combination of the proposals in [16] and [17].

For the example, the vector S = [15 21 26] is encoded as

S = [1 1 1 1 0 1 0 1 0 1 0 1 0 1 1]

It can be observed that the least significant bit is the first line for each line.

3.3 Initial population

There exist several algorithms to generate radial networks. In [18] the authors generated radial topologies implementing Prim's algorithm and Kruskal. In this paper the implemented method consists of the following steps: 1) Generate a random vector for loop selection. 2) For each basic loop one line is chosen randomly. 3) The line is deleted from the following sets of basic loops with the main purpose of avoiding its selection. 4) Repeat steps 2 and 3 for all remaining loops. 5) When the solution vector S is full, its radiality is checked.

The process is repeated if the vector does not fulfill the radiality condition. This will generate radial topologies, which always respect the positions of the loop in vector S.

3.4 Crossover operation

The crossover operation used is a uniform crossover implemented by using a binary mask. That is, if the i-th element of the mask is 1, the gene located in that position of parent 1 passes to the descendant, similarly, if the element is 0 the gene is copied from parent 2 to the descendant in its corresponding position. This is shown in Fig. 3.

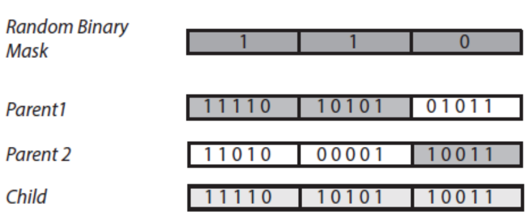


Fig. 3. Crosses of individuals.

After performing the crossover, the new individual is checked to see if it maintains the radiality of the system. In the case this new individual does not represent a radial system, the individual is discarded and generates a new binary mask, in order to obtain an individual who meets the restriction of radiality.

3.5 Mutation operator

To perform this operation, a normally open line is randomly selected to change its status. This is described in the following steps: 1) Select one line of the chromosome randomly. 2) Identify the basic loop to which it belongs and remove the assembly line. 3) Select a random line of the loop assembly and replace the main line into the chromosome. 4) Check the system radiality.

If the system does not meet the radiality condition, the process is repeated. In this way, the mutation operator ensures that the algorithm does not stagnate in a local minimum and performs a search in a larger space of solutions.

To illustrate this process, line 3 was the randomly selected chromosome at line 26, which is removed from the main loop three. To replace this line, one line of the set of the main loop 3 is randomly selected, line 23. Fig. 4 shows the individual obtained by the mutation operation.

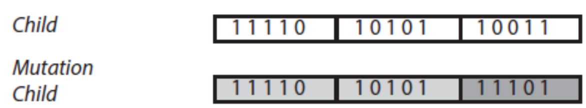


Fig. 4. Mutation of individuals.

In a multi-objective optimization problem, two different stages must be taken into account. The first one is related to the optimization concept, which seeks to find the Pareto optimal set. Since the optimal solutions are considered as equals, a decision must be made. In the next stage, this decision is made according to the particular problem to be solved.

3.6 Fitness function evaluation

In order to evaluate the fitness function, the system voltage sags and power losses must be evaluated; the estimation of voltage sags is performed by using the fault positions method [1][14], and the power loses are obtained by using the power flow program MATPOWER [19].

The procedure for application of the proposed reconfiguration method is described next:

1) Read the electrical system's parameter data and data of reference values of voltage sags V_{ref}. for a considered voltage threshold (t_{spec}) at system buses,
2) Identify the base system and provide it as an individual in the initial population.
3) Generate the population as described in subsection 3.3.
4) Carry out the genetic operations, described above, to generate new individuals.
5) Identify the lines for each solution vector that are out of operation and change the status of system parameters.

6) Evaluate pre-fault voltages, V_{pre}, by performing a power flow study and obtain the total active power losses.
7) Evaluate the voltage sags in order to obtain the vector V_B. considering the current operating status of the system.
8) If results do not meet the criteria of stop, return to step 4 and re-evaluate possible solutions.
9) Select a solution of the Pareto optimal set, prioritizing systems having the lowest number of nodes with voltage depression problems, provided that the losses are less than the base system.

In this work, the proposed reconfiguration method has been implemented in *Matlab*® by using the functions described in this section. In addition, functions *@gamultiobj, @fitscalingprop* and *@selectournament* from the *toolbox Global Optimization Toolbox* [20] are employed.

4 Case studies

In order to show the performance of the proposed methodology, two case studies are presented for IEEE 33-bus and IEEE-118 bus distribution test systems. In both, it is assumed that the voltage sags are caused by three-phase balanced faults.

4.1 Studies for the IEEE-33 bus distribution system.

The data of the IEEE 33-bus test system can be found in [21], which is a 12.66 KV radial distribution system. This system consists of one substation and 33 buses interconnected by 32 lines, where 5 of them are tie lines. The total active and reactive load is 3715 KW and 2300 KVA, respectively. Active power losses of the initial system are 202.67 kW and the minimum voltage is 0.913 p.u. at node 18.

The fault position method was used in order to carry out the stochastic estimation of voltage sags, considering 10 fault positions for each line, a uniform distribution of fault probability at lines, the original configuration of the distribution system, and a voltage sags threshold of 0.8 p.u.

Fig. 5 shows the IEEE 33-bus distribution system. As shown, five switches through normally open conditions meet the radiality of the system. In Table 1 system buses having problems of voltage sags are indicated, and also the lines that are usually open are shown.

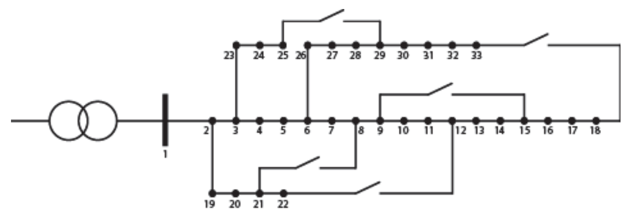


Fig. 5. IEEE 33-bus distribution system.

Table 1. Actual status of the IEEE 33-bus test sytem.

Voltage sags threshold 0.8 p.u.	
Buses with voltage-sags values above the reference values	4,5,6, ,10,12,13,15,16,17,18,21, 26,27,29, 30,31,32,33
Active power losses	202.67 kW
Minimum voltage	0.9131 p.u. at bus 18
Open lines	33, 34, 35, 36, 37

For this specific case there is only one non-dominated solution, in other words this solution is the best for both objective functions. After the reconfiguration was applied, the new configuration has only four buses with values above reference voltage sags values and active power losses were reduced on 29.56%. Table 2 shows the results with the new reconfiguration and, also the disconnect lines are presented. This configuration satisfies the radiality criteria.

Table 2. Results for the IEEE 33-bus radial sytem after reconfiguracion

Voltage sags threshold 0.8 p.u.	
Buses with voltage-sags values above the reference values	4 20 21 22
Active power losses	142.75 kW
Minimum voltage	0.9378 p.u. at bus 32
Open lines	37, 7, 32, 11, 34

4.2 Studies for the IEEE 118-bus distribution system

The multi-objective reconfiguration has been tested with the IEEE 118-bus distribution system [22], in order to show the capacity of the proposed methodology to handle large-scale problems. This test system have 132 lines and 122 of them can change its status. This system has moderate size, however because the large number of lines that can be switched, result in a complicated combinatorial problem. Figure 6 shows the IEEE118-bus system base case.

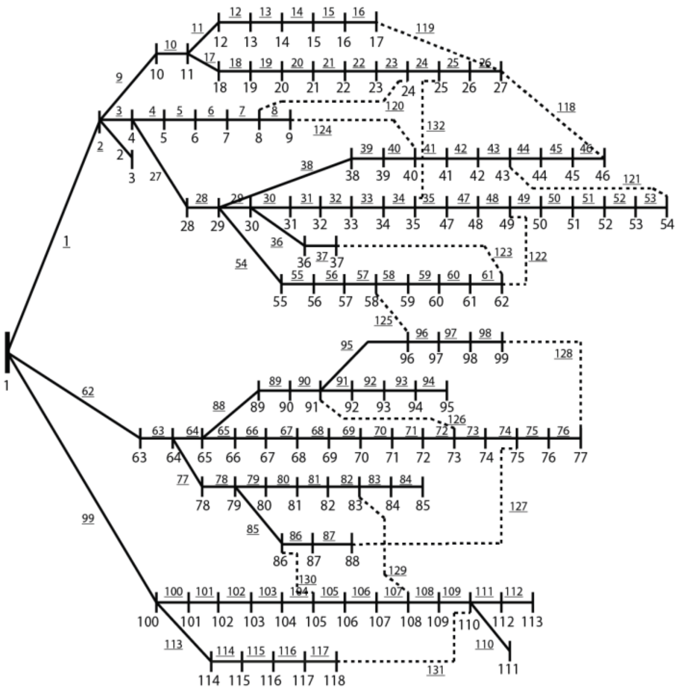


Fig. 6. 118-bus distribution test system.

Table 3 shows data for the base case of the IEEE 118-bus test system, it can be observed that the system has 77 buses with voltage-sags values above the reference values, and active power losses are 1298.09 kW.

The parameters used in the genetic algorithm to solve the multi-objective problem of reconfiguration are shown in Table 4.

Table 4. Simulation parameters

Number of generations	Population size	Crossover Rate	Mutation Rate	Simulation time
36	15	1	0.3	38,935 s

Table 4. Actual status of the IEEE 118-bus distribution test sytem (before reconfiguracion)

Voltage sags threshold 0.8 p.u.	
Buses with voltage-sags values above the reference values	5 7 8 11 12 14 16 17 18 20 21 24 25 26 28 29 30 33 34 35 36 37 38 39 43 44 45 47 48 49 50 51 52 53 54 55 56 57 60 62 64 65 66 67 68 69 70 71 72 73 74 75 76 77 79 80 82 84 85 89 90 91 92 93 94 95 96 97 98 99 101 104 105 107 109 112 116
Active power losses	1298.09 kW
Minimum voltage	0.8687 p.u. at bus 77
Open lines	118 119 120 121 122 123 124 125 126 127 128 129 130 131 132

Solving the optimization problem yields a set of Pareto Optimal (non-dominated) solutions and the Pareto Front which are shown in Fig. 7 and Table 5. There are 7 solutions that meet not only radiality but also active power loss and voltage sags reduction.

All the found solutions are equally optimal, but only one of them must be selected. In this paper, the selected solution corresponds to the system that has fewer nodes with voltage depression problems.

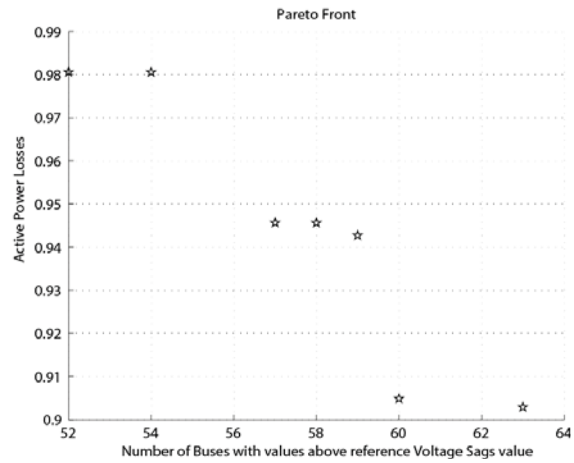


Fig. 7. Pareto Front for the 118-Bus distribution test System.

Table 5. Pareto optimal set

Pareto optimal set	Pareto front	
Disconnected lines	Number of buses with values above reference value	Active power losses
43 14 21 53 46 60 38 95 126 73 128 82 130 109 34	59	942.66 kW
42 26 21 53 46 60 37 95 70 73 128 82 130 109 34	60	904.38 kW
43 14 21 53 46 60 37 95 126 73 98 82 130 109 34	58	945.60 kW
43 14 21 53 46 60 37 95 126 73 128 82 130 109 34	57	945.62 kW
42 26 4 53 46 60 124 95 70 73 128 82 130 109 33	54	980.51 kW
42 26 4 53 46 60 124 95 70 73 98 82 130 109 33	52	980.54 kW
42 26 21 53 46 60 38 95 70 73 128 82 130 109 34	63	902.28 kW

The reconfigured system is shown in Fig. 8, and the results of this reconfiguration are summarized in Table 6. As shown, initially the system had 77 buses that did not meet the specified value, and at 25 of them, the voltage sags have been reduced to appropriate values, leaving 52 nodes that do not fulfill the specifications.

The new configuration reduces both objective functions, and also improves the voltage profile. The minimum voltage is presented at bus 111, with a value of 0.9321 p.u. The first objective function, which corresponds to the nodes with voltage depression problems, has a reduction of 32.46%, while active power losses are reduced by 24.46% respect to the base case.

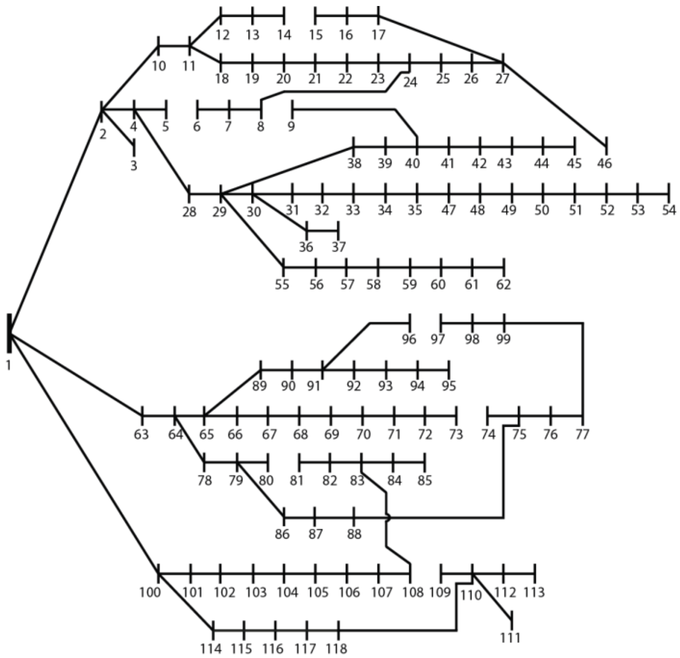


Fig. 8. IEEE 118-bus distribution test system after reconfiguration, considering the 0.8 p.u. voltage threshold.

Table 6. Results for 118-bus distribution test sytem after reconfiguracion

Voltage sags threshold 0.8 p.u.	
Buses with voltage-sags values above the reference values	5 6 7 8 9 11 12 14 16 17 18 20 21 23 24 25 26 34 35 37 43 44 45 49 50 55 60 61 62 64 65 66 73 79 80 82 85 87 88 92 93 94 95 96 101 104 105 107 109 112 116 117
Active power losses	980.54 kW
Minimum voltage	0.9321 p.u. at bus 111
Open lines	42 26 4 53 46 60 124 95 70 73 98 82 130 109 33

In Fig. 9 the values of reference voltage sags, the initial sags (SagsBase) and those obtained after the system reconfiguration for the 118-bus distribution system are graphed, considering a voltage threshold of 0.8 p.u. Clearly, after performing the reconfiguration, voltage sags decrease in several nodes, but in other buses the values of voltage sags increase. This is due to the modification of network topology; however, the system operates correctly, and with a reduction in voltage sags with respect to the initial topology of the system.

Based on the conducted studies considering the same objective function with a limited number of switches in the same case studies, it was found that the search space and consequently the computational cost, have been reduced, reflecting the proper functioning of the proposed methodology.

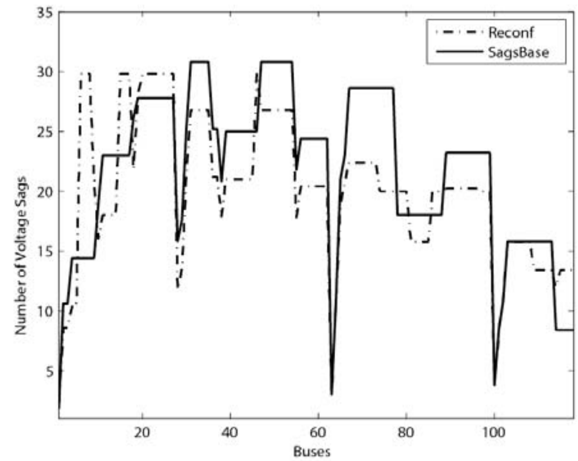


Fig. 9. Voltage sags/year, initial and after reconfiguration values, considering a 0.8 p.u. voltage threshold.

5 Conclusions

In this paper a methodology for the reconfiguration of radial distribution electrical systems in order to reduce voltage sags and active power losses, using a multi-objective optimization technique was described. The methodology based on genetic algorithms, generates and reproduces individuals representing radial topologies, given that the genetic operators were designed for that purpose, in order to reduce of the solutions search space, improving the computational cost and the solution time.

Two case studies were presented, and results show the efficiency of the proposed method to solve the multi-objective problem leading to solutions that improve the operation of the distribution network.

Acknowledgment

This work has been partially supported by the Consejo Nacional de Ciencia y Tecnología, México, under grant no. 388080.

References

[1] M. Kezunovic, "Smart fault location for smart grids", IEEE Transaction on Smart Grid, Vol. 2, No. 1, pp. 11-22, March 2011.

[2] J. A. Momoh, "Smart grid design for efficient and flexible power networks operation and control", IEEE/PES PSCE'09 Power Systems Conference and Exposition, Seattle, USA, March 2009.

[3] N. Barrera, M. J. Pavas, F. Meléndez, "Power quality assessment of the Bogotá distribution network focused on voltage sag analysis", IEEE PES, Innovative Smart Grid

Technologies, Europe Conference 2011, Manchester, UK, Dec. 2011.

[4] M. H. J. Bollen, J. Zhong, F. Zavoda, J. Meyer, A. McEachern, F. Córcoles, "Power quality aspects of smart grids", International Conference on Renewable Energies and Power Quality, March 2010.

[5] E. M. Carreno, R. Romero, A. Padilha-Feltrin, “An efficient codification to solve distribution network reconfiguration for loss reduction problem”, IEEE Trans. on Power Systems, vol. 23, no. 4, pp. 1542- 1551, Nov. 2008.

[6] S García-Martínez, E Espinosa-Juárez, “Optimal reconfiguration of electrical networks by applying tabu search to decrease voltage sag indices”, Electric Power Components and Systems, vol. 41, no. 10, pp. 943-959, 2013.

[7] R Tapia-Juarez, E Espinosa-Juárez, M Graff, “Optimal reconfiguration of radial distribution networks for reducing voltage sags”, Electrical Engineering, Computing Science and Automatic Control (CCE), 2013.

[8] L. Tang; F. Yang ; J. Ma, “A survey on distribution system feeder reconfiguration: Objectives and solutions”, Innovative Smart Grid Technologies - Asia (ISGT Asia), May 2014.

[9] R. Tapia-Juarez, E. Espinosa-Juarez, "Multi-objective reconfiguration of radial distribution networks considering voltage sags", IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), 2013, pp.1,6, Nov. 2013.

[10] C.A. Coello Coello, D.A. Van Veldhuizen, G.B. Lamont, “Evolutionary algorithms for solving multi-objective problems”, New York: Kluwer Academic Publishers; 2002.

[11] T.P. Bagchi, “Multiobjective scheduling by genetic algorithms”, Boston: Kluwer Academic Publishers; 1999.

[12] K. Y. Lee and M. A. El-Sharkawi, “Modern Heuristic Optimization Techniques”, Edit. The Institute of Electrical and Electronics Engineers, Inc. 2008.

[13] M. R. Qader, M. H. J. Bollen, “Stochastic prediction of voltage sags in a large transmission system”, IEEE Trans. on Industry Applications, vol. 35, no. 1, pp. 152-162, Jan. 1999.

[14] M. H. J. Bollen, “Understanding power quality problems. Voltage sags and interruptions”, IEEE PRESS Series on Power Engineering, 2000

[15] S. Civanlar, J. Grainger, H. Yin, S. Lee, “Distribution feeder reconfiguration for loss reduction”, IEEE Trans. on Power Del., vol. 3, no. 3, pp. 1217–1223, Jul. 1988.

[16] M. Lavorato, J. F. Franco, M. J. Rider, and R. Romero "Imposing radiality constraints in distribution system optimization problems" IEEE Trans. on PowerSystems, vol. 27, no. 1, pp 172-180, Feb. 2012.

[17] J. Z. Zhu, “Optimal reconfiguration of electrical distribution network using the refined genetic algorithm,” Elect. Power System Research, vol. 62, pp. 37–42, 2002.

[18] J. C. Cebrian and N. Kagan "Evolutionary algorithms for the reconfiguration of radial distribution networks considering long interruption and voltage sag disruption costs", 13th International Conference on Harmonics and Quality of Power, 2008. Sept 2008.

[19] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning and Analysis Tools for Power Systems Research and Education," Power Systems, IEEE Transactions on, vol. 26, no. 1, pp. 12-19, Feb. 2011.

[20] Global Optimization Toolbox. User's Guide, The MathWorks, Inc., 2012

[21] M. E. Baran, F. F. Wu, “Network reconfiguration in distribution systems for loss reduction and load balancing”, IEEE Trans .on Power Del., vol. 4, no. 2, pp. 1401–1407, Apr. 1989.

[22] D. Zhang, F. Zhengcai, Z. Liuchun, "An improved TS algorithm for loss-minimum reconfiguration in large-scale distribution systems", Electric Power Systems Research, vol. 77, no. 5-6, pp. 685-694, 2007.

SESSION

FUZZY LOGIC AND SETS, FUZZY CONTROL AND APPLICATIONS

Chair(s)

TBA

Optimizing Cost with Intelligent Energy Management System based on Fuzzy Logic

Musbah Abdulgader[1], Cheng Yang [1], and Devinder Kaur [1]
[1] Department of EECS, University of Toledo, Toledo, OH 43607 USA

Abstract - *In this paper an Intelligent Energy Management System based on fuzzy logic is proposed. The objective of the proposed strategy is to optimize the cost saving for the end consumer by determining the share of the power drawn from the grid or the Energy Storage Unit for each hour of the day, because the price of the energy varies each hour depending on the demand and is determined a day before. The strategy is based on Mamdani's Fuzzy Inference System. The proposed design will also avoid overloading the grid during the periods of high demand by shifting the load to the ESU.*

Keywords— Intelligent Energy Management System; optimization; fuzzy logic

1. Introduction

Energy consumption is increasing worldwide [1]; therefore it makes good sense to lower the consumption from the grid generated using traditional methodology to reduce mankind's footprint. The electricity bill of a large enterprise can be minimized by managing dynamic energy pricing [2] [3]. The wholesale electrical companies determine the cost of energy for each hour of the day at midnight before the day begins [4]. Consumers can use this information to optimize their energy consumption. The idea is to shift the load from a high price to a low price. In order to achieve this some basic infrastructure is assumed, such as a smart meter connected to the grid for two way communication of price [5] and load demand. Also an Electric Storage Unit to store the energy; when the price is relatively cheaper and release it when it is high and two switches one for drawing energy from the grid and the other for using the ESU for meeting the demand [6] [7]. Thus, one of the main ideas of smart grids is encouraging the consumer to participate in making decisions about energy consumption in an efficient way. In this paper we have come up with an Energy Management Strategy based on simple rules which mimic human thinking. This strategy provides us the ratio of the power drawn from the grid and the ESU at any particular hour of the day, to minimize the cost to the end consumer and also reduce the load on the grid during peak hours by not drawing from the grid [9] [10].

Fig. 1 illustrates the infrastructure of Large Enterprise using intelligent EMS.

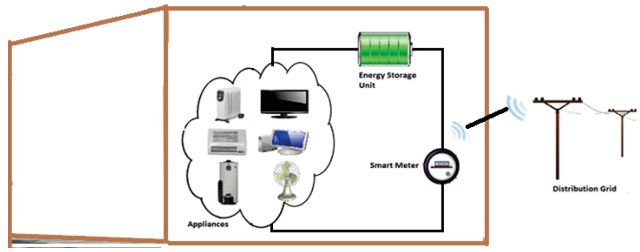


Fig. 1. Infrastructure of Large Enterprise for Intelligent EMS

Fig. 2 illustrates the hourly power price curve of a typical day. The lowest hourly price was at 15:00 and the highest hourly price was at 18:00 PM. Fig. 3 illustrates the hourly grid load curve of a typical day. Hourly prices were significantly affected by the hourly grid load. Therefore, the highest grid load also occurred at 18:00 PM [8].

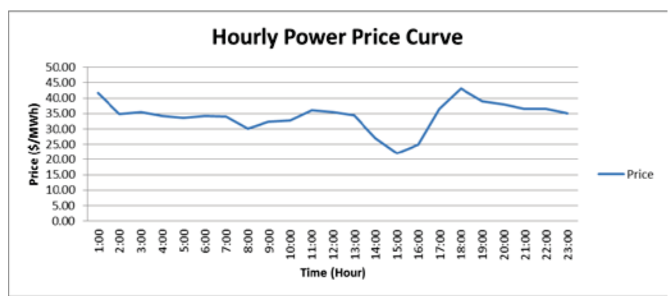


Fig. 2. Power price curve on Jan/1/2011[8]

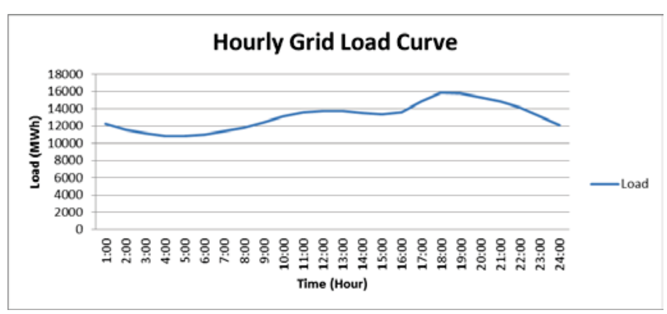


Fig. 3. Grid load curve on Jan/1/2011[8]

Fig. 4 illustrates the hourly cost of power consumption curve of a typical day computed by Equation.

$$Cost = Price * Load \quad (1)$$

In Fig. 4, the lowest hourly cost was at 15:00, when people are usually at work. The highest hourly cost was at 18:00, when people came home and used the energy for domestic needs [8].

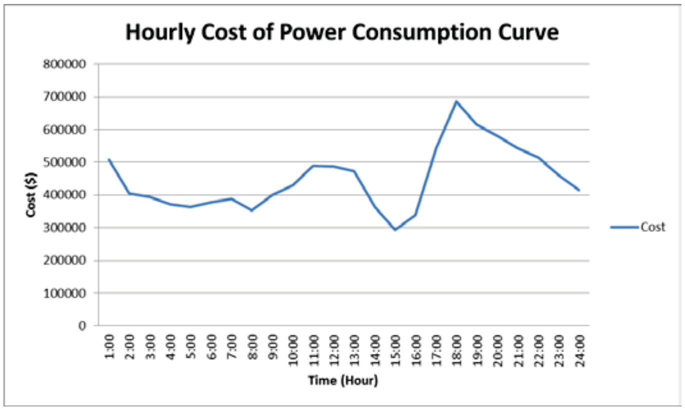


Fig. 4. Cost of power consumption curve on Jan/1/2011

The paper is organized as follows: Section II describes the basic concepts of Fuzzy Logic; In Section III, an Intelligent Energy Management System based on Mamdani's Fuzzy Inference System is proposed; In Section IV implementation of Mamdani's Fuzzy Inference System is discussed; Section V discusses the results and the cost savings obtained by using this Fuzzy Logic based approach.

2. Fuzzy Logic

The purpose of Fuzzy Logic is to map an input space to an output space with simple if-then-else rules which mimic human-thinking. All rules are equally evaluated, and the order of the rules does not affect the results. In general, fuzzy inference is a method that interprets the values in the input space and, based on some set of rules, assigns values to the output space [11][15]. Fuzzy Logic starts with the concept of a fuzzy set. Compared with a classical set, a fuzzy set is a set without crisp and clearly defined boundaries [12] [15].

In a classical set theory, the universe is black and white. Either an element belongs to the set with 100% membership, or it does not belong to the set at all. In fuzzy set theory an element can belong to more than one fuzzy set with a membership degree ranging anywhere from 0 to 100%. For example, Fig 5 illustrates classical sets. Fig 6 illustrates fuzzy sets. An element, say a price of 23, belongs to both the fuzzy sets Cheap and Medium. However, in classical sets price 23 will either belong to one set which is Cheap, and it can never belong to more than one set at the same time [13].

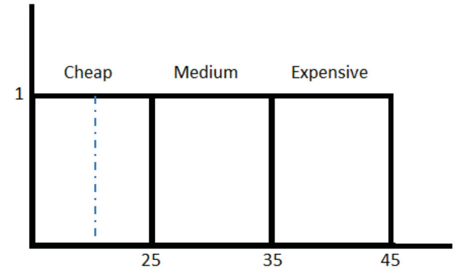


Fig. 5. Classical Sets

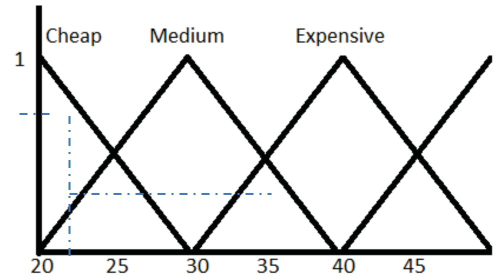


Fig. 6. Fuzzy Sets

If X is the universe of discourse, and its elements are denoted by x, then a fuzzy set A in X is defined as a set of ordered pairs. It is shown in Equation 2 [11] [12].

$$A = \{ x, \mu_A(x) \mid x \in X \} \quad (2)$$

$\mu_A(x)$ is the membership degree of x in A [14]. The membership degree maps each element of X to a value between 0 and 1[11].

There are two most widely used fuzzy inference systems: Mamdani's Fuzzy Inference System and TSK Fuzzy Inference System. Fuzzy inference systems have been successfully applied to the domains of automatic control, data classification, decision analysis, expert systems, and computer vision [14]. In TSK model the output is expressed as a mathematical equation.

Mamdani's Fuzzy Inference System is preferred when no relation between the input and output parameters is known and the relationship can express in the form of if-then-else rules. It was proposed in 1975 by Ebrahim Mamdani [12].

3. Mamdani's FIS Strategy

Fig. 7 illustrates the framework for Intelligent Energy Management System [13]. The integrated Mamdani's Fuzzy Inference System will process the input of price and load, and generate output which will be sent as a control signal to operate Grid Switch and Storage Switch. The control signal sent to Grid Switch will determine how Smart Meter directs electricity from the Distribution Grid to the Energy Storage Unit and other appliances [14]. The control signal sent to Storage Switch will operate the Energy Storage Unit and it will decide whether the Energy Storage Unit is charging or releasing [13].

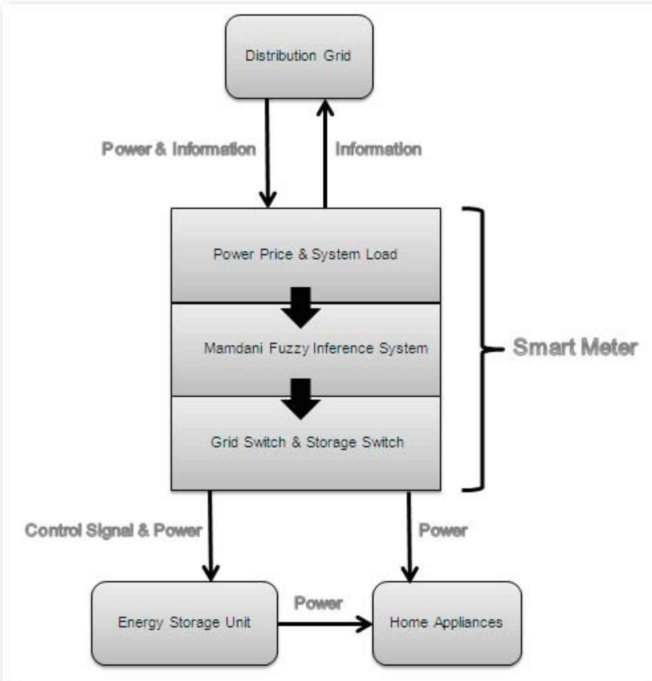


Fig. 7. Framework of Mamdani' FIS Strategy [8]

Fig. 8 illustrates Mamdani's FIS for Energy Management. The crisp inputs of price and load are presented to fuzzy inference system, which goes through the process of fuzzification, rule evaluation, and defuzzification to generate a crisp output to control the state of ESU which is either charging or discharging. All the six rules are listed below:

1. If price is cheap and load is low, then ESU is charging at 100% of the Load at that hour.
2. If price is cheap and load is high, then ESU is charging at 60% of the Load at that hour.
3. If price is medium and load is low, then ESU is charging at 20% of the Load at that hour.
4. If price is medium and load is high, then ESU is releasing at 20% of the Load at that hour.
5. If price is expensive and load is low, then ESU is releasing at 60% of the Load at that hour.
6. If price is expensive and load is high, then ESU is releasing at 100% of the Load at that hour.

The first rule is "if price is cheap and load is low, then ESU is charging at 100%." 100% charging means that ESU is charging to the capacity of the load at that hour, e.g. if the load at a particular hour is 2 KW, then ESU is charging to 2KW.

Similarly, if ESU is releasing at 20%, this means 20% of the total demand at that particular hour is being met by the storage unit, e.g. if the load at that hour is 10KW, then 2KW is being met by the storage unit and rest of the 80% which is 8KW is being drawn from the grid.

The following four trends emerge from the rules.

Trend 1: When power price is cheaper, ESU is charging more.
Trend 2: When power price is more expensive, ESU is releasing more
Trend 3: When grid load is lower, ESU is charging more
Trend 4: When grid load is higher, ESU is releasing more [8]

These four trends precisely simulate the process of decision making in a human mind. Trend 1 and Trend 2 are focused on cost reduction; while Trend 3 and Trend 4 are focused on avoiding overloading of the grid.

Mamdani FIS for Energy Management Unit is implemented with the following steps:

Step 1: Fuzzification of Inputs

In this step, input of power price and input of grid load are being fuzzified through membership function. Before the fuzzification, inputs of power price ranged from 19.94 to 558.55 ($/MWh), while inputs of grid load were from 8294 to 27707 (MWh) [4]. After the fuzzification, input parameter price was represented with three fuzzy sets: "expensive", "medium" and "cheap." Input parameter load was represented with two fuzzy sets: "high" and "low [8]."

Step 2: Evaluation of Rule Strength

Three fuzzified values for power price and two fuzzified values for grid load were generated from Step 1 and ranged from 0 to 1. In Step 2, fuzzy operators apply to these values through six rules. For example, Rule 1 is "if price is cheap and load is low, the ESU is charging at 100%." When 0.6 is the fuzzified value of cheap and 0.4 is the fuzzified value of low, then the rule strength is the fuzzy AND operation of two sets Cheap and Low which is 0.4.

Step 3: Truncation of Output Fuzzy Set Associated with the Rule

The consequence is truncated with the rule strength obtained in step 2. In this instance, the output ESU charging at 100% will be truncated at 0.4. This will be done for all six rules [8].

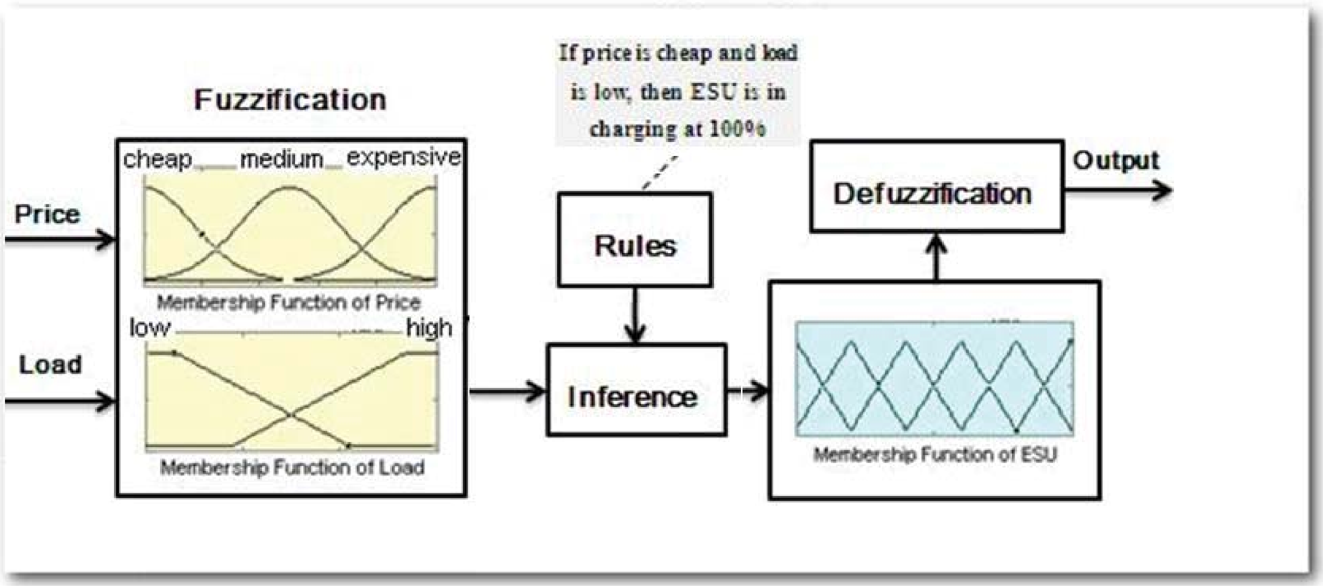


Fig. 8. Mandani's FIS for Energy Management [8]

Step 4: Aggregation of Output of Rules

The rules must be combined in some manner in order to make the decision. Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set [15].

Step 5: Defuzzification

In this step, all the aggregate outputs from step 4 are combined together to give one crisp output which shows the state of ESU charging or releasing. The most popular defuzzification method is the centroid calculation, which returns the center of the area under the curve. Centroid calculation has been applied in the proposed design.

4. Methodology and Implementation

The Energy Management Unit is implemented in MATLAB using Fuzzy Logic Toolbox. Fuzzy Logic Toolbox is an integrated development tool of Matlab that provides Matlab functions, graphical tools, and a Simulink block for analyzing, designing, and simulating systems based on fuzzy logic. Fuzzy Logic Toolbox consists of the following five GUI tools to build, edit and view fuzzy inference systems [15]:

1. Fuzzy Inference System (FIS) Editor
2. Membership Function Editor
3. Rule Editor
4. Rule Viewer
5. ***Surface Viewer***

These five GUI tools are dynamically linked. Any changes made to the fuzzy inference system through one GUI tool will affect other GUI tools.

Fig. 9 shows the FIS editor displaying Mamdani's model for Energy Management Unit. Inputs are price of the power and load on the grid. Output is the state of Charge of ESU. "And method" computes the minimum of fuzzy sets. "Or method" computes the max of fuzzy sets. "Implication" is set to be min. "Aggregation" is set to be max. "Defuzzification function" is set to centroid.

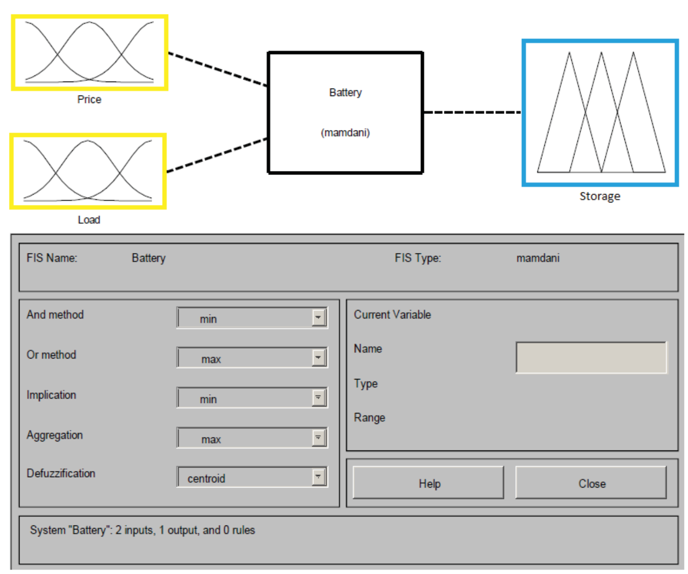


Fig. 9. Mamdani's model in FIS Editor [8]

Fig. 10 shows three membership functions for price: cheap, medium, and expensive. All three membership functions are set to be Gaussian type. Fig.11 shows two membership functions of Load: low and high. Both membership functions are set to be trapezoid type [8].

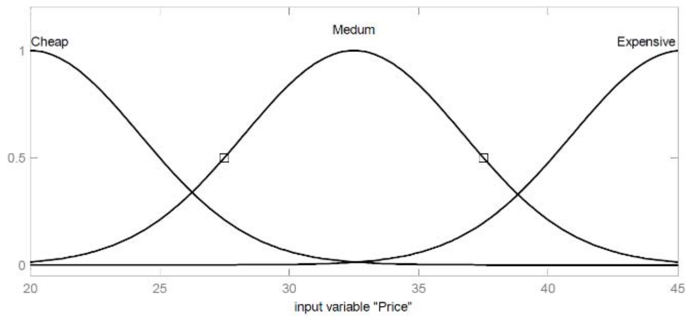


Fig. 10. Membership Functions of Price

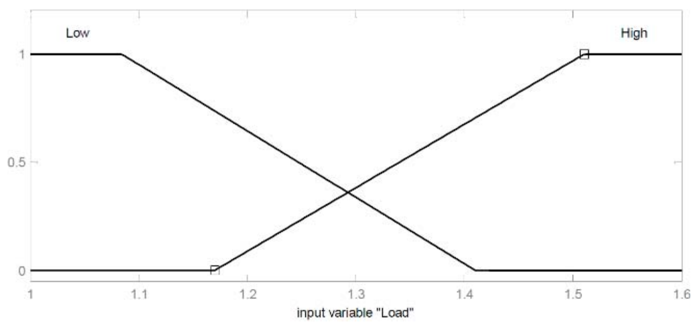


Fig. 11. Membership Functions of Load

Fig.12 shows six membership functions for ESU: releasing high, releasing medium, releasing low, charging low, charging medium, and charging high. All membership functions are set to be triangle type.

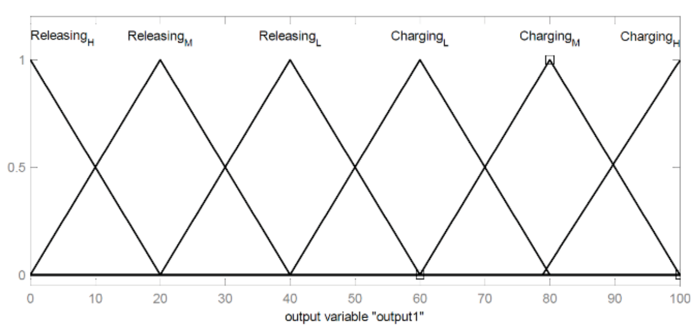


Fig. 12. Fuzzy Membership functions for output variables

Fig 13 shows a snapshot of the rules which fire for a given input of price and grid load. The output computed after defuzzification is 42.9 for proposed FIS [8].

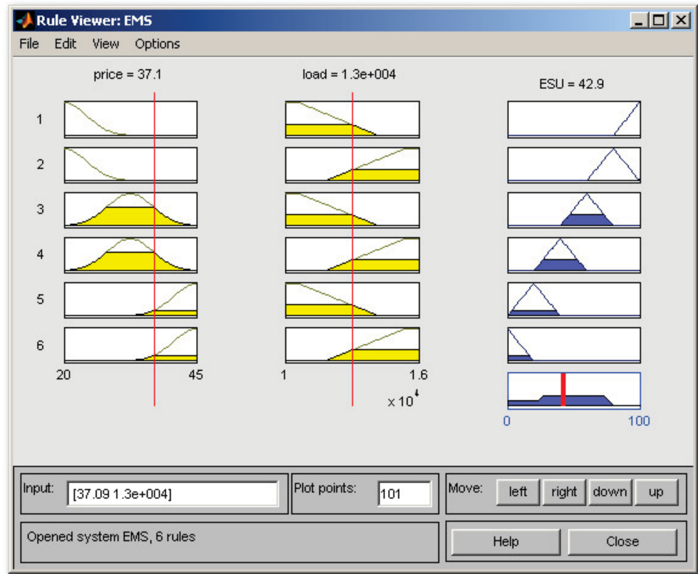


Fig. 13. Rule Viewer for proposed Mamdani's FIS based model

5. Performance Evaluation

Fig. 14 illustrates the hourly load curve of Mamdani's FIS strategy in a typical day and shows that the load is shifted from high price to the low price. By using Mamdani's FIS strategy it is clear that when the grid load is high, the Energy Management unit releases power and does not draw power from the grid, thus not contributing to increase the load on the grid and making the grid more stable.

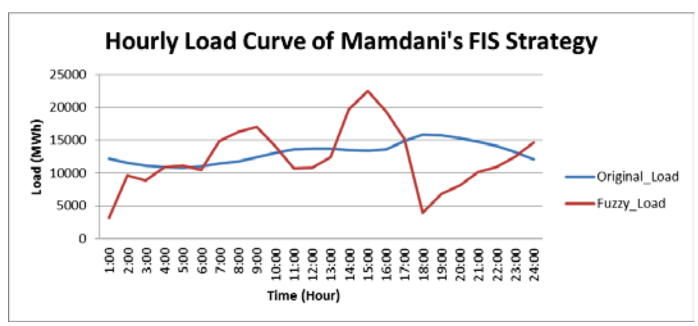


Fig. 14. Load curve of Mamdani's FIS on Jan/1/2011[8].

Fig 15 shows hourly cost of power consumption of Mamdani's FIS strategy in a typical day. When the cost of the power is high then the Energy Management unit releases the stored energy for the need of the enterprise thus saving cost for the consumer.

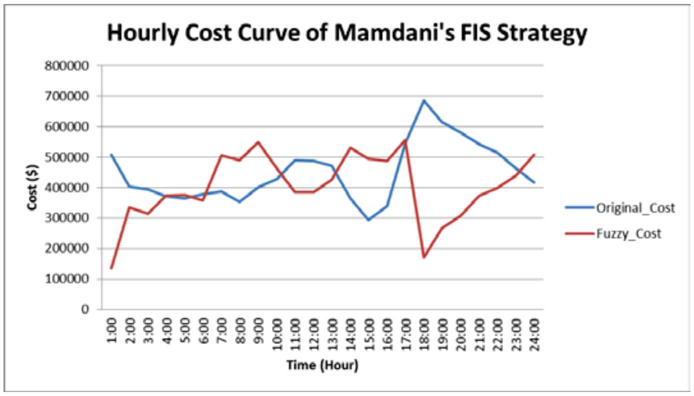


Fig. 15. Hourly cost of Mamdani's FIS on Jan/1/2011[8]

Table I shows the quarterly saving rates of Mamdani's FIS strategy [8]. The saving rates are from 11.4% to 30.1

TABLE I
QUARTERLY SAVING RATES OF MAMDANI'S FIS STRATEGY

Months	Original Cost ($)	Mamdani's FIS Cost ($)	Mamdani's FIS Saving
Jan-Mar	571.1	483.85	15.3
Apr-Jun	467.18	396	15.2
Jul-Sep	662.91	463.3	30.1
Oct-Dec	409.68	363.03	11.4

6. Conclusion

In this paper an Energy Management Unit (EMU) has been proposed for a large enterprise with consumption ranging in several MW. Mamdani's FIS approach is designed to reduce cost for the end consumers and avoid grid overload for the power supplier. The saving rates vary from 11.4% to 30.1%. Therefore, it has been demonstrated that Mamdani's FIS based approach resulted in cost saving as compared to the original cost without using any technique. This same approach can be applied to smaller dwelling units if the price of the energy varies every hour.

7. References

[1] Enerdata, World Energy Use in 2010: Over 5% Growth, World Energy Statistics: Enerdata Yearbook, 2011.

[2] S. Hatami, M. Pedram, "Minimizing the Electricity Bill of Cooperative Users under a Quasi-Dynamic Pricing Model, First IEEE International Conference on," Smart Grid Communications (SmartGridComm), pp.421-426, 4-6 Oct. 2010.

[3] N. Kumaraguruparan, H. Sivaramakrishnan, S.S. Sapatnekar, "Residential task scheduling under dynamic pricing using the multiple knapsack method", proceedings of IEEE PES conference on Innovative Smart Grid Technologies (ISGT), pp.1-6, 16-20 Jan. 2012.

[4] ISO - New England Inc., http://www.iso-ne.com/aboutiso/index.html.

[5] P. Baronti, P. Pillai, V.W.C. Chook, S. Chessa , A. Gotta , Y.F. Hu "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards", Computer Communications, 30 (7), pp. 1655-1695, 2007.

[6] P. Ruetschi, "Low loss battery," US Patent 2,994,626, 1961.

[7] Sandia National Laboratories, "Energy Storage Systems Program (ESS)," http://www.sandia.gov/ess/index.html.

[8] C. Yang, "Development of Intelligent Energy Management System using Natural Computing", Master's Thesis submitted to University of Toledo, 2012.

[9] M. Elsied, A. Oukaour, H. Gualous, R. Hassan, A. Amin, "An advanced energy management of microgrid system based on genetic algorithm", IEEE 23rd International Symposium on Industrial Electronics (ISIE), pp. 2541-2547, 1-4 June 2014.

[10] R. E. Brown, Impact of Smart Grid on Distribution System Design, Proceeding IEEE Power and Energy Society General Meeting, 2008; pp. 1-4.

[11] E. H. Mamdani, and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," International Journal of Man-Machine Studies, vol. 7, no. 1, pp. 1-13, 1975.

[12] M. Sugeno, "Industrial applications of fuzzy control," Elsevier Science Publication Co., 1985.

[13] D. Shahgoshtasbi,; M.M. Jamshidi, "A New Intelligent Neuro–Fuzzy Paradigm for Energy-Efficient Homes," IEEE Transaction on Systems , vol.8, no.2, pp. 664-673, June 2014.

[14] J. S. R. Jang, and C. T. Sun, "Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence," Prentice Hall, 1997.

[15] J.-S. Roger Jang Ned Gulley "MATLAB fuzzy Logic Toolbox http://www.manualslib.com/manual/392793/Matlab-Fuzzy-Logic-Toolbox-2.html

Fuzzy Methodologies for Multi-Sensor Information Fusion with Applications to Precision PNT

James A. Crowder and John N. Carbone,
Raytheon Intelligence, Information, and Services, Aurora, CO, USA
Raytheon Intelligence, Information, and Services, Garland, TX, USA

***Abstract*—** Providing processing for intelligence operations requires collecting, sorting, and fusing data from a variety of sources to produce coherent and correlated intelligence "information" (Multi-Sensor Data Fusion). Multi-sensor data fusion is an evolving technology, concerning the problem of how to fuse data from multiple sensors in order to make a more accurate assessment of a given situation or environment. Applications of data fusion cross a wide spectrum, including automatic target detection and tracking, battlefield surveillance, remote sensing, etc. They are usually time-critical, cover a large geographical area, and require reliable delivery of accurate information for their completion. One of the big problems with multi-sensor fusion is the Level 0 processing: A new approach to fusion is the joint Mutual Information between the features and the class labels. It can be shown that Mutual Information minimizes the lower bound of the classification error. However, according to Shannon's definition this is computationally expensive.

Evaluation of the joint Mutual Information of a number of variables is plausible through histograms, but only for a few variables. If we look toward a different definition of Mutual Information we find a different result. Using Renyi's entropy instead of Shannon's, combined with Parzen density estimation, leads to expression of Mutual Information with significant computational savings. Here, we will extended Renyi's method for Mutual Information to multiple continuous variables and discrete class labels to learn linear dimension-reducing linear feature transforms for data fusion and parameter estimation utilizing competing parameter measures.

Keywords—Fuzzy Logic, Mutual Information, Information Fusion

1. Introduction: Multi-sensor fusion

Multi-sensor data fusion/integration is an evolving technology, concerning the problem of how to fuse data from multiple sensors in order to make a more accurate measurement of the environment. Applications of data fusion cross a wide spectrum, including automatic target detection and tracking, battlefield surveillance, remote sensing, etc. They are usually time-critical, cover a large geographical area, and require reliable delivery of accurate information for their completion [9]. According to the Office of Naval Research:

"Sensor integration is concerned with the synergistic use of multiple sources of information. In warfare, no one piece of information can be accepted as complete truth. The combination of information from every possible source is of primary importance."

Sensor fusion/integration is divided into three classes: complimentary sensors, competitive sensors, and cooperative sensors [8]:

Complimentary sensors do not depend on each other directly but can be merged to form a more complete picture of the environment, for example, a set of radar stations covering non-overlapping geographic regions. Complementary fusion is easily implemented since no conflicting information is present.

Competitive sensors each provide equivalent information about the environment. A typical competitive sensing configuration is a form of N-modular redundancy. For example, a configuration with three identical radar units can tolerate the failure of one unit. This is a general problem that is challenging, since it involves interpreting conflicting readings.

Cooperative sensors work together to drive information that neither sensor alone could provide. An example of cooperative sensing would be using two video cameras in stereo for 3D vision. This type of fusion is dependent on details of the physical devices involved and cannot be approached as a general problem.

Here we attack the problem of real-time, distributed, competitive sensor fusion for time-critical sensor readings. Figures 1 and 2 depict sensor fusion scenarios for this study. It is assumed that each sensor platform has some local intelligence and memory [3]. We also assume that every sensor has limited accuracy and that a limited number of readings may be arbitrarily faulty, each m_i uses possibly different logic to deduce the position, velocity, and parametric measurements of the object under surveillance." Once the information is transmitted to the central processing system, fusion and situational awareness software are used to provide an overall picture of the battlefield to the war fighter. Figure 3 below illustrates a block diagram of an overall fusion and situational awareness processing system. One of the big problems with multi-sensor fusion is the Level 0 processing shown below. This involves putting the various sensors into a classification system where the sensors can be evaluated against each other. i.e., how to eliminate the differences between the sensors so the information content of each can be fused into intelligent information with error bounds consistent with the various information sources [13].

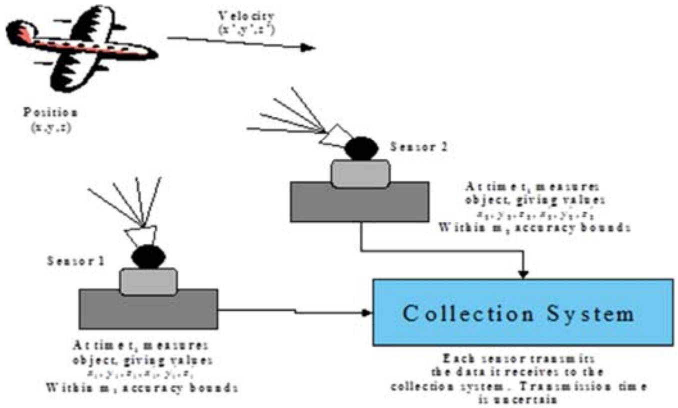


Figure 1 - Competitive Ground Sensors

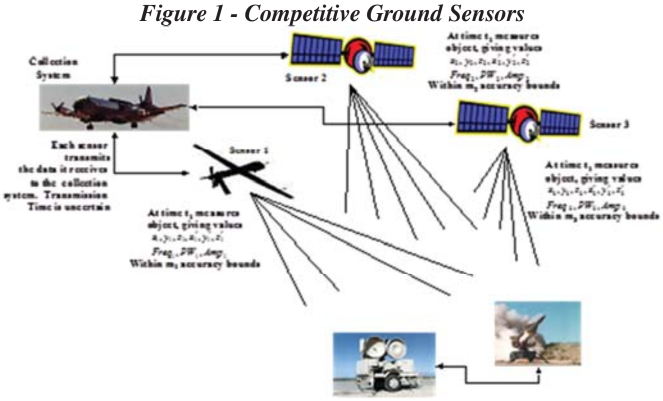


Figure 2 - Competitive Air and Space Sensors

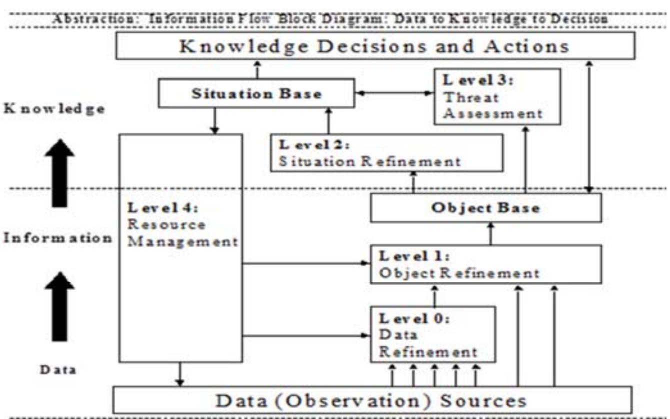


Figure 3 - Data/information flow for situational awareness

The process of data fusion requires a large number of disciplines including signal and image processing, control theory, database design, networks, data standards, as well as human computer interface. Military research applicable to data fusion is in the areas of Intelligence Surveillance and Reconnaissance (IRS) sensors, Command and Control (C2), Communications (C), and Computers, which collectively make up C4ISR. This paper is concerned with Level 0 (Data Refinement) for Competitive Sensors; how to provide a normalization environment to fuse various sensor information so that the overall intelligence and situational awareness processing system can ingest, process, and report on large volumes of disparate intelligence information.

2. JOINT ENTROPY

Since the information coming into a given processing system may have from 1-to-n inputs at any given time, and each sensor that provides input will have random errors associated with any given measurement. The data streams coming into the processing systems may be seen as systems of random data, or each data stream represents a random variable to the system. We will start with a random pair of variables, ***(X,Y)***. Another way of thinking of this is as a vector of random variables.

Definition 1: If ***X*** and ***Y*** are jointly distributed according to ***p(X, Y)***, then the Joint Entropy ***H(X, Y)*** is (**EQ 1**):

$$H(X,Y) = -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log p(x,y)$$

or

$$H(X,Y) = -E \log p(x,y)$$

Definition 2: if ***(X, Y) ~ p(x, y)***, then the conditional entropy ***H(Y|X)*** is (**EQ 2**):

$$H(Y \mid X) = -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log p(y\mid x) = -E_{p(x,y)} \log p(y \mid x)$$

This can also be written in the following equivalent ways:

$$H(Y \mid X) = -\sum_{x\in X} p(x)\sum_{y\in Y} p(y\mid x)\log p(y\mid x) = -\sum_{x\in X} p(x) H(Y \mid X = x)$$

Theorem 1: (Chain Rule)

$$H(X \mid Y) = H(X) + H(Y \mid X)$$

The uncertainty (entropy) about both ***X*** and ***Y*** is equal to the uncertainty (entropy) we have about ***X***, plus whatever we have about ***Y***, given that we know X. This can also be done in the following streamlined manner: Write (**EQ 3** Joint Entropy)

$$\log p(X,Y) = \log p(X) + \log p(Y \mid X)$$

and take the expectation of both sides. We can also have a joint entropy with a conditioning on it, as how below:

Corollary 1:

$$H(X,Y \mid Z) = H(X \mid Z) + H(Y \mid X,Z)$$

2.1 Relative Entorpy and Mutual Informaiton

Suppose there is a random variable with true distribution *p*. Then (as we will see) we could represent that random variable with a code that has average length *H*(*p*). However, due to incomplete information we do not know *p*; instead we assume that the distribution of the random variable is *q*. Then (as we will see) the code would need more bits to represent the random variable. The difference in the number of bits is denoted as *D*(*p*|q). The quantity *D*(*p*|q) comes up often enough that it has a name: it is known as the **relative entropy**.

Definition 3:The relative entropy or **Kullback-Leibler** distance between two probability mass functions ***p(x)*** and ***q(x)*** is defined as (**EQ 4**):

$$D(p \parallel q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)} = E_p \log \frac{p(X)}{q(X)}$$

It should be noted that this is **not** symmetric, and that ***q*** (second argument) appears only in the denominator.

Another important concept is Mutual Information [2]. This describes how much information on random variable tells about another one. This is, perhaps, the central idea in information theory. When we look at the output of a sensor, we see the information as a random variable. What we want to know is what was sent, and the only information we have is what cam out of the sensor. Or, we have two sensors that are each random variables, both providing information about the same thing. We know that the information should be the same, but it is not because of a host of errors that the sensors and the transport medium introduced into the measurements. What we want is to extract the exact information from the sensor readings. Or, in other words, we want to find the mutual information between the sensor two different sensor readings.

Definition 4: Let ***X*** and ***Y*** be random variables with joint distribution ***p(X,Y)*** and marginal distributions ***p(x)*** and ***p(y)***. The **Mutual Information** ***I(X;Y)*** is the relative entropy between the joint distribution and the product distribution (**EQ 5**):

$$I(X;Y) = D(p(x,y) \parallel p(x)p(y))$$
$$= \sum_{x} \sum_{y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

Note that if ***X*** and ***Y*** are truly independent then ***p(x,y) = p(x)p(y)*** so ***I(X;Y) = 0***. However, if they are sensing the same thing, then they should not truly be independent. An important interpretation from Mutual Information comes from the following theorem:

Theorem 2:

$$I(X;Y) = H(X) - H(X \mid Y)$$

The interpretation of this is that the information that ***Y*** tells us about ***X*** is the reduction in uncertainty about ***X*** due to the knowledge of ***Y***. The information ***X*** tells about ***Y*** is the uncertainty in ***X*** plus the uncertainty about ***Y*** minus the uncertainty in both ***X*** and ***Y***. We can summarize a bunch of statements about entropy as follows [10]:

$$I(X;Y) = H(X) - H(X \mid Y)$$
$$I(X;Y) = H(Y) - H(Y \mid X)$$
$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$
$$I(X;Y) = I(Y;X)$$
$$I(X;X) = H(X)$$

3. Fuzzy Data Normalization and Fusion

Figure 4 represents a processing flow for intelligence information. The process involves two main layers, the deductive process and the investigative process. The deductive process goes after assembling information that has been previously known while the inductive process (data mining) looks for patterns and associations that have not been seen before. The model illustrated in Figure 3-1 is the deductive process used to detect previously known patterns in many sources of data by searching for specific information signatures and templates in data streams to understand the state of the intelligence knowledge [16]. As the systems continues to evolve in complexity, the number of objects, situations, threats, sensors and data streams dramatically increase, presenting a very complex challenge for advanced fusion system designers. In order to keep the system "on-top" of its data environment is to have data mining operations going on in the background at all times, finding new associations and evolving the templates and information correlations [6].

Data Mining is an off-line knowledge creating process where large sets of previously collected data is filtered, transformed, and organized into information sets. This information is used to discover hidden but previously undetected intrusion patterns. Data mining is called knowledge/pattern discovery and is distinguished from the data fusion process by two important characteristics, inference method and temporal perspective [7]. Data fusion uses known templates and pattern recognition. Data mining processes search for hidden patterns based on previously undetected intrusions to help develop new detection templates. In addition, data fusion focuses on the current state of information and knowledge; data mining focuses on new or hidden patterns in old data to create previously unknown knowledge, illustrated in Figure 4 [11].

In both data mining and data fusion, feature selection or feature transforms are important aspects of any system. Optimal feature selection coupled with pattern recognition leads to a combinatorial problem since all combinations of available features must be evaluated before deciding how to fuse the information available.

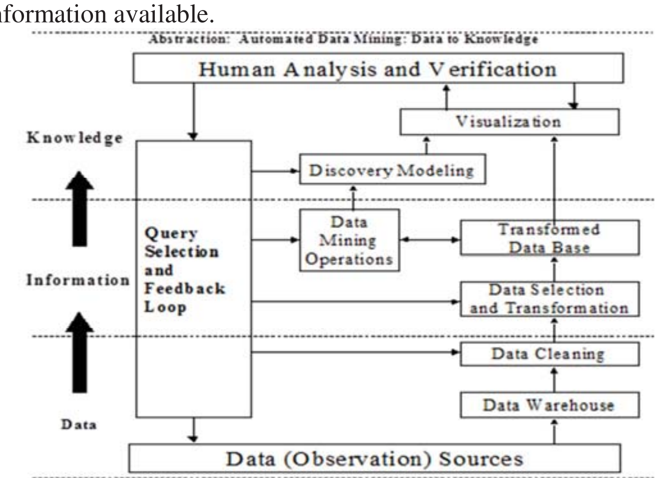


Figure 4 - Data/Information Flow for Data Mining Operations

Another such criterion is the joint ***Mutual Information*** between the features and the class labels. It can be shown that Mutual Information minimizes the lower bound of the classification error. However, according to Shannon's definition this is computationally expensive. Evaluation of the joint Mutual Information of a number of variables is plausible through histograms, but only for a few variables. If we look toward a different definition of Mutual Information we find a different result. Using Renyi's entropy instead of Shannon's, combined with Parzen density estimation, leads to expression of Mutual Information with significant computational savings. As a part of this study, we extended Renyi's method for Mutual Information to multiple continuous variables and discrete class labels to learn linear dimension-reducing linear feature transforms for data fusion and parameter estimation utilizing competing parameter measures [5].

We applied Renyi's entropy-based Mutual Information measure to create fuzzy membership functions that can be used to rapidly asses the Mutual Information content between multiple measurements of a given parameter from different sensors [1]. We introduce the Mutual Information measure based on Renyi's entropy, and describe its application to Fuzzy Membership Functions that were used transform multiple parameter measures and error estimates into a single parameter and error bound estimate for the parameter.

3.1 Shannon's Definition of Mutual Informaiton

We denote labeled samples of continuous-valued random variable ***Y*** as pairs $\{y_i, c_i\}$, where $y_i \in R^d$, and class labels are samples of a discrete-valued random variable *C*, $c_i \in \{1,2,\ldots,N_c\}, i \in [1, N]$. If we draw one sample of ***Y*** at random, the entropy or uncertainty of the class label, making use of Shannon's definition, is defined in terms of class prior probabilities (**EQ 6 – Shannon's Entropy Theory**) [4]:

$$H(C) = -\sum_c P(c)\log(P(c))$$

After having observed the feature vector y, our uncertainty of the class identity is the conditional entropy is (**EQ 7**):

$$H(C) = \int_y p(y)\left(\sum_c p\left(c/y\right)\log\left(p\left(c/y\right)\right)\right)dy$$

The amount by which the class uncertainty is reduced after having observed the feature vector *y* is called the Mutual Information, $I(C,Y) = H(C) - H(C/Y)$, which can be written as (**EQ 8**):

$$I(C/Y) = \sum_c \int_y p(c,y)\log\frac{p(c,y)}{P(c)p(c)}dy$$

Mutual information also measures independence between two or more variables, in this case between ***C*** and *Y*. It equals zero when $p(c,y) = P(C)p(y)$, i.e., when the joint density of ***C*** and ***Y*** factors (the condition for independence). Mutual Information can thus be also viewed as the divergence between the joint densities of the variables, and the product of the marginal densities. Connection between Mutual Information and Data Fusion is given by Fanno's inequality.

This result, originally from digital communications, determines a lower bound to the probability of error when estimating a discrete random variable ***C*** from another variable *Y* (**EQ 9**):

$$\Pr(c \neq \hat{c}) \geq \frac{H(C/Y)-1}{\log(N_c)} = \frac{H(C)-I(C/Y)-1}{\log(N_c)}$$

where $\hat{c}$ is the estimate of ***C*** after observing a sample of *Y*, which can be a scalar or multivariate. Thus the lower bound on error probability is minimized when Mutual Information between ***C*** and ***Y*** is maximized, or, finding such features achieves the lowers possible bound to the error of the classifier. Whether this bound can be reached or not, depends of the goodness of the classifier.

3.2 A Definition Based on Renyi's Entropy

Instead of Shannon's entropy, we apply Renyi's quadratic entropy because of its computational advantages. For a continuous variable ***Y***, Renyi's quadratic entropy is defined as (**EQ 10**):

$$H_R(Y) = -\log\int_y p(y)^2 dy$$

It turns out that Renyi's measure, combined with the Parzen density estimation method using Gaussian kernels, provides significant computational savings, because a convolution of two Gaussians is still a Gaussian.

If the density $p(y)$ is estimated as a sum of symmetric Gaussians, each centered at a sample y_i as:

$$p(y) = \frac{1}{N}\sum_{i=1}^{N} G(y - y_i, \sigma I)$$

then it follows that the integral above equals (**EQ 11**):

$$\int_y p(y)^2 dy =$$

$$= \frac{1}{N^2}\int_y\left(\sum_{k=1}^{N}\sum_{j=1}^{N} G(y - y_k, \sigma I)G(y - y_j, \sigma)\right)dy$$

$$= \frac{1}{N^2}\sum_{k=1}^{N}\sum_{j=1}^{N} G(y_k - y_j, 2\sigma I)$$

Thus, Renyi's quadratic entropy can be computed as a sum of local interactions as defined by the kernel, over all pairs of samples. In order to make use of this convenient property, we make use of fuzzy membership functions and the natural way they demonstrate local interactions to find a function which maximized Mutual Information among sensor measurements.

3.3 A Maximizing Mutual Information

In any real-time system, data arrives at the input as a random variable, since it cannot be known a' priori what data may or may not be received at any given time. This is particularly true of the type of system radar environment represent. In fact, each measurement is a random variable *X*, with a σ bound determined by the system dynamics. Since each of the sensors acts as a random variable we are looking to maximize Mutual Information in order to find the normalization that produces the best overall result.

If we assume each sensor measurement to be a random variable $\boldsymbol{x}$, with its associated s bound, we can form a fuzzy membership distribution function around each measurement using the measurement as the mean (best guess with the information given) and the given error bound as the membership function bounds.In order to maximize Mutual Information across the measurements, we need to find the random variable $\boldsymbol{x_i}$, which minimizes the distance between random variables. We accomplish this by mapping the each measurement value onto each of the fuzzy membership functions [12]. We compute fuzzy membership curves for each sensor, based on measurement and □□ populate each with measurements from all sensors. Figure 5 illustrates 12 sensor measurements on the same system, each reporting a slightly different RF. Each system would have its own measurement error, as shown in Figure 6.

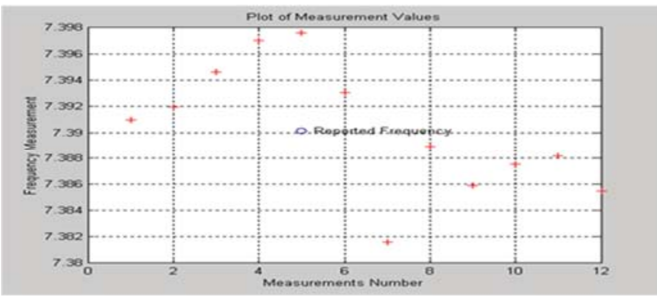


Figure 5 - Example of Multiple Sensor Measures of the Same Frequency

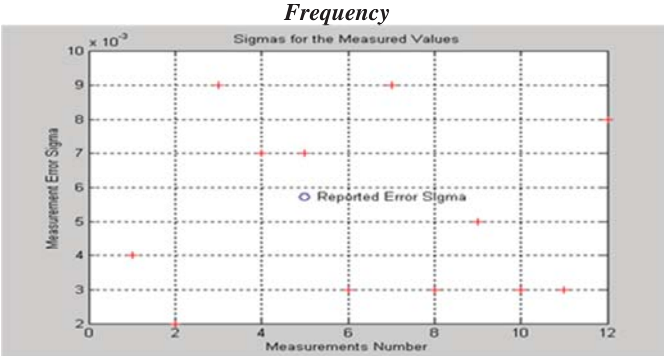


Figure 6 - Examples of Multiple Sensors, each with their own Error Bounds

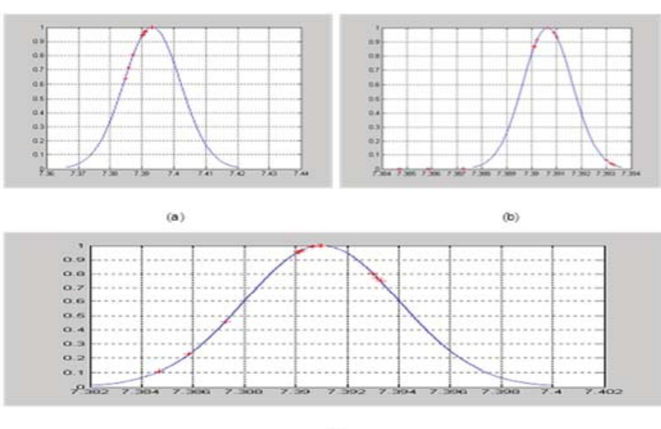


Figure 6 - Examples of Fuzzy Membership Functions for Sensor Measurements, Populated with all the Sensor Measurements

For each sensor, a fuzzy membership normalization function is formed and then each sensor measurement is mapped onto each membership normalization function (**EQ 12**):

$$Y_i = E_{j=1}^{n}\left(e^{\frac{-(M_i - M_j)^2}{2*\sigma_i^2}} \right)$$

Figure 7 a-c illustrates this process

Once all of the curves have been populated, we compute the mean fuzzy membership value for each curve (**EQ 14**):

$$Y_i = E_{j=1}^{n}\left(e^{\frac{-(M_i - M_j)^2}{2*\sigma_i^2}} \right)$$

$$Y_{\max} = \max_{i=1,n} (Y_i)$$

The normalization function with the highest mean membership represents the normalization mapping with the highest Mutual Information and is therefore given the highest weighting in determining the measurement value to report. The weighting factors are then determined for rolling up the measurements and error bounds into a single parametric estimation (EQ 15):

$$W_i = Y_i \Big/ \sum_{j=1}^{n} Y_j$$

where the $\boldsymbol{W_i s}$ are the weighting factors. Figure 7 illustrates the process.

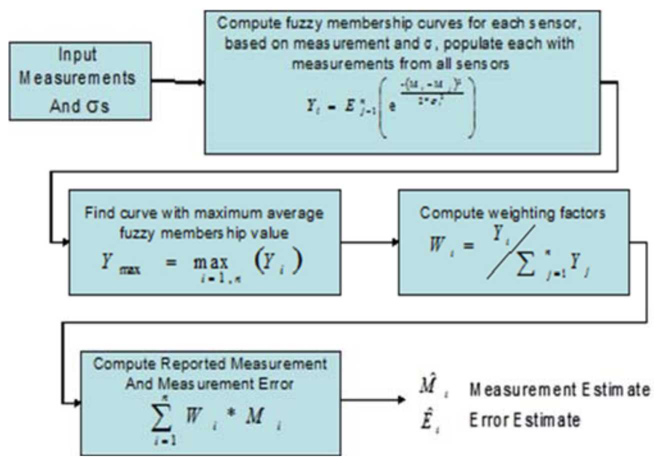


Figure 7 - Weighted Fuzzy Parameter Estimation Process

4. CONCLUSIONS AND DISCUSSION

Renyi's theory of information is extremely important in intelligence work, much more so than its use in cryptography would indicate. The theory can be applied by intelligence agencies to keep classified information secret, and to discover as much information as possible about an adversary. His fundamental theorems lead us to believe it is much more difficult to keep secrets than it might first appear. In general it is not possible to stop the leakage of classified information, only to slow it.

Furthermore, the more people that have access to the information, and the more those people have to work with and belabor that information, the greater the redundancy of that information becomes. It is extremely hard to contain the flow of information that has such a high redundancy. This inevitable leakage of classified information is due to the psychological fact that what people know does influence their behavior somewhat, however subtle that influence might be.

The premier example of the application of information theory to covert signaling is the design of the Global Positioning System signal encoding. The system uses a pseudorandom encoding that places the radio signal below the noise floor. Thus, an unsuspecting radio listener would *not even be aware that there was a signal present*, as it would be drowned out by atmospheric and antenna noise [15]. However, if one integrates the signal over long periods of time, using the "secret" (but known to the listener) pseudorandom sequence; one can eventually detect a signal, and then discern modulations of that signal. In GPS, the C/A signal has been publicly disclosed to be a 1023-bit sequence, but the pseudorandom sequence used in the P(Y) signal remains a secret. The same technique can be used to transmit and receive covert intelligence from short-range, extremely low power systems, without the enemy even being aware of the existence of a radio signal. We believe the use of Renyi's information theory might enhance these capabilities over the original work by Shannon [14].

The discussion below illustrates one possible use of Renyi's Information theory and the use of Fuzzy Filters to implement this theory for asynchronous data fusion for PNT estimations for Unmanned Air and Underwater vehicles (UAVs and AUVs). The fuzzy estimator performs asynchronous data fusion of all sensor measurements based on their relative confidence levels, and then nonlinearly combines the fused information with the INS estimates via fuzzy implementation of Renyi's mutual information theory. The basis and implementation of the estimator is described, and navigation results are presented based on the fuzzy estimator. We believe a fuzzy normalization procedure similar to the one outlined here provides the best automated, and dynamic way to roll-up sensor measurements into a single reported measurement and error bound for intelligence reporting and analysis. Section B presents the results of a number of tests for this process.

4.1 Fuzzy Data Fusion for UAV/AUV PNT Estimates

Figure 15 illustrates the results of utilizing the Stochastic Derivative algorithms on the pulse environments described above. The Higher-Order moments of the 1st 8 Stochastic Derivatives were generated and plotted for each of the signal environments. As can be seen from 15, even 15 pulses from a pseudorandom-driven, multiply-agile radar signal caused a jump in the Stochastic Derivative moments. And 300 pulses out of 10,000 cased a major jump in the Higher Order Stochastic Derivative moments. Clearly the algorithms can detect the presence on non-stochastic signals in the environment.

With rapid progress in COTS sensors and electronics technology, miniaturized Autonomous Underwater Vehicles (AUV) and Unmanned Air Vehicles (UAV) have reached an acceptable level of maturity and reliability that can be capitalized on their use for commercial and military applications. Examples include spatio-temporal surveys during clandestine oceanographic and air-surveillance environments. Without requiring any tethering support, the dynamic stability and data sampling quality can be much improved. In addition, multiple small AUVs or UAVs can be deployed simultaneously to traverse in different regions without necessitating one-to-one support contact and allows higher data sampling efficiency. To truly characterize four-dimensional ocean or air dynamics autonomously, high precision underwater and air navigation is a technically challenging issue because vehicle localization must be done onboard. While differential GPS sensor technology provides a good solution for surface navigation, underwater and complex air navigation still remains a challenging problem especially for autonomous vehicles. The main goal here is to present a novel fuzzy data fusion that collates different and independent asynchronous position sensors together, and nonlinearly gain schedules them with the onboard INS system. One important objective here is to evaluate the possible use of the fuzzy implementation of mutual information theory for autonomous navigation and look at the effectiveness of the fuzzy sensor-based fusion approach with respect to steady-state and convergence performance of bias estimation.

4.2 Fuzzy PNT Fusion Architecture

Figure 8 shows a general architecture for setting up a navigational. In this architecture, there are arbiters created for position sensors, attitude sensors and motion sensors. This set up is desirable because many existing AUVs and UAVs incorporate multiple sensors performing same functions, and it is thus beneficial to fuse all information to obtain the best navigation estimates. In cases of sensor failure, these arbiters will reconfigure in order to complete time-critical missions. Inputs to the position sensor arbiter are absolute position measurements, which can be based on (D)GPS and various forms of baseline sensors. These measurements are particularly valuable because of their drift-free properties over a longer time scale, as compared to the dead-reckoning position estimate. However, over a shorter time scale, the DGPS measurements introduce undesirable position error for compensation. It is thus important to carefully sample these so that the signal-to-noise ratio is maximized. In addition, measurements from all sensors are generally unavailable at every time sampling instant, and a strategy is thus needed to combine these asynchronous measurements before routing the result to the position estimator. Since the number of sensors may not be constant one data point to the next, the real-time fuzzy fusion techniques discussed here are applicable to this problem.

To combine the position measurements, each sensor is assigned a confidence value that characterizes its expected variance in error about the true value. For an example, typical DGPS can have 1-5m range error (σ) together with horizontal dilution of precision (HDOP) uncertainty due to satellite geometry. The total position error introduced in terms of root-mean-square value is HDOP *σ. It should be noted that while the range error is generally hard to quantify, the HDOP profile can be readily obtained from any receiver, and thus it should be accounted for in the position estimator otherwise position error might be compromised. Once a suitable time sampling interval is chosen, the output of the position sensor arbiter is given as

$X = \sum Z_i C_i$ where X is the arbiter output, $\boldsymbol{Z_i}$ is the ***ith*** sensor output, and $\boldsymbol{C_i}$ is its confidence value which has accounted for both the range error and baseline geometry of satellites or sonar beacons.

Note that a constraint of $\sum_{i=1}^{N} C_i \equiv 1$ is imposed on the confidence values and C_i is then loosely interpreted as the probability that the ith sensor is correct.

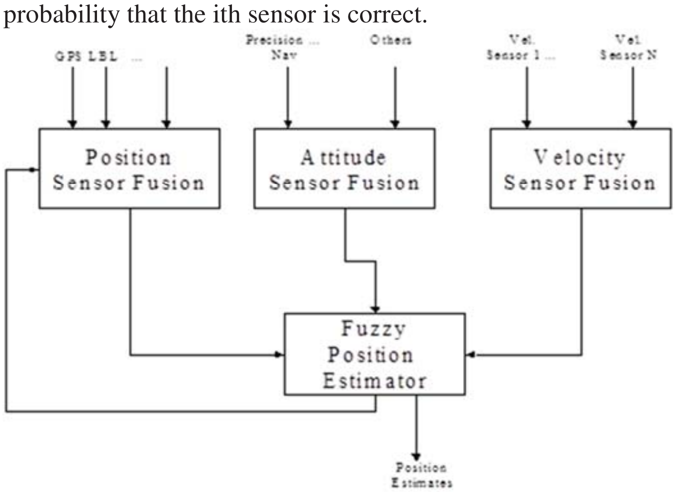


Figure 8 - High-Level Fusion Architecture for PNT Estimation

The use of the fuzzy fusion algorithm discussed here provides a practical estimation algorithm that is not computational intensive but yet provides theoretically sound approach in performing data fusion. Figure 9 below illustrates the GPS X-Y error evaluation for the test.

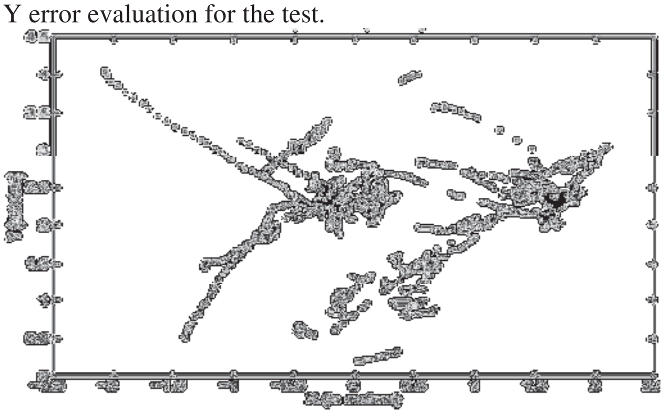


Figure 9 – GPS Error Evaluation

Figure 10 compares the INS way-point navigation performance based on 1) doppler returns (dash line) and 2) GPS + doppler (solid line). In these figures, `X' and `O' represent a differential and regular GPS fix respectively, and in both missions the AUV was started at the origin. During these missions, the AUV was underwater most of the time, and commanded to surface during some specified cornerings in order to obtain fixes, and thus position drift due to Doppler and attitude sensors can be easily observed, as compared to the DGPS fixes as the only source of reference. At the end of the last eastward leg in Figure B3, there was a significant discrepancy between the position estimator and DGPS measurement (approximately 1.5% error based on 50 meters after 3300 meters transect). Figure B3 presents results of a 3-hr mission covering 15km transect. Sporadic fixes can be seen in the figure which corresponds to the AUV surfacing maneuvers. Among these fixes, maximum discrepancy between the position estimator and GPS fixes was found at location [-50 east, 150 north], and also it can be seen that the position estimator responds less to these GPS fixes, but much more to the DGPS fixes obtained immediately afterwards. By observation, the discrepancy was approximately 100m since the last update (6 legs of transect away _ 3300m), and thus the error was approximately 3%. It should be noted that 100m is within the limit of the GPS error deviation, and the result suggests that accurate navigation does not necessarily require frequent surfacing. The results here demonstrate the usefulness of the fuzzy – mutual information algorithms for use in real-time sensor fusion. We believe further investigation is required to determine the complete usefulness of this approach. In Figure 10, the solid line represents the fuzzy position estimator output and the dashed line represents the dead-reckoned output. 'X' and 'O' represent differential and regular GPS fix, respectively.

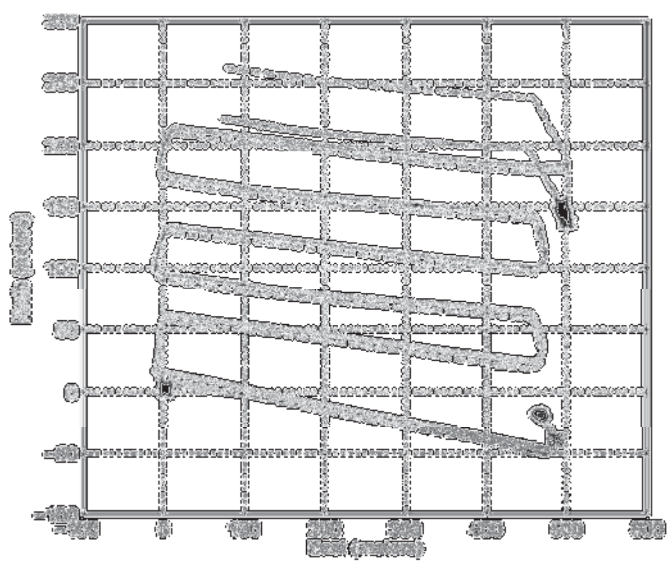
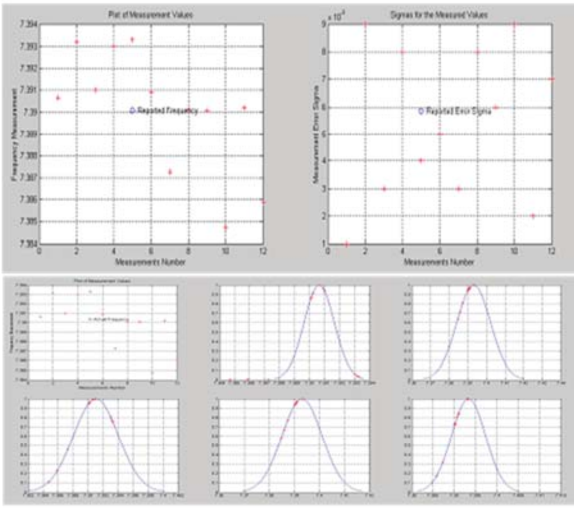
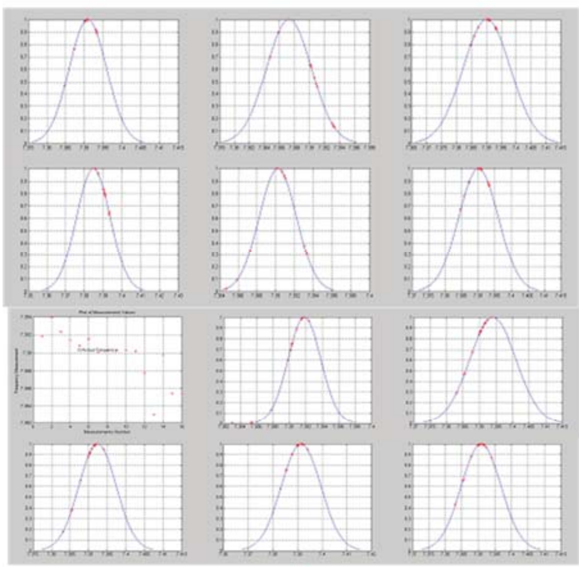
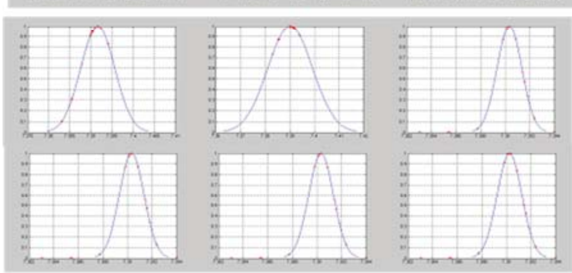


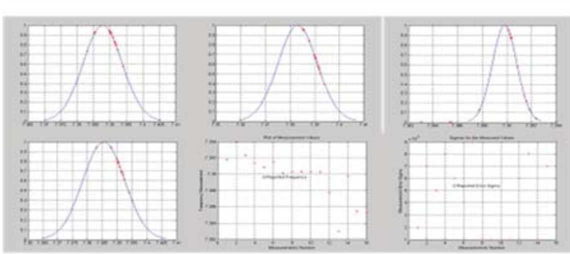
Figure 10 – Fuzzy PNT Estimation and GPS Outputs

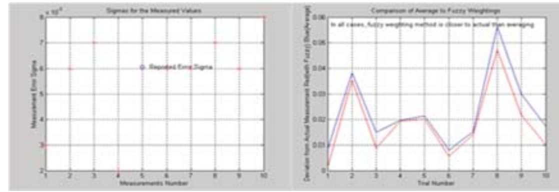
4.3 Results for the Mutual Information Fusion Process











5. REFERENCES

1. Bonarini, A. 1997. Anytime Learning and Adaptation of Structured Fuzzy Behaviors. Adaptive Behavior Volume 5. Cambridge MA: The MIT Press.
2. Brillouin, L. 2004. Science and information theory. Dover.
3. Crowder, J. 2002. Derivation of Fuzzy Classification Rules from Multidimensional Data. NSA Technical Paper CON_0013_2002_001.
4. Crowder, J. A. 2002. Machine Learning: Intuition (Concept Learning) in Hybrid Neural Systems, NSA Technical Paper-CON-SP-0014-2002-06 Fort Meade, MD.
5. Crowder, J. 2004. Multi-Sensor Fusion Utilizing Dynamic Entropy and Fuzzy Systems. Proceedings of the Processing Systems Technology Conference, Tucson, AZ.
6. Crowder, J. 2005. Cognitive Systems for Data Fusion. Proceedings of the 2005 PSTN Processing Technology Conference, Ft. Wayne, Indiana.
7. Crowder, J. A. 2010a. The Continuously Recombinant Genetic, Neural Fiber Network. Proceedings of the AIAA Infotech@Aerospace-2010, Atlanta, GA.
8. Crowder, J. A., 2010c. Flexible Object Architectures for Hybrid Neural Processing Systems. Proceedings of the 11th International Conference on Artificial Intelligence, Las Vegas, NV.
9. J. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White. 2004. Revisiting the jdl data fusion model ii.
10. Crowder, J. A. and Crowder, J. A., and Carbone, J. N. 2011a. Recombinant Knowledge Relativity Threads for Contextual Knowledge Storage. Proceedings of the 12th International Conference on Artificial Intelligence, Las Vegas, NV.
11. Carbone, J. N. and Crowder, J. 2011b. Transdisciplinary Synthesis and Cognition Frameworks. Proceedings of the Society for Design and Process Science Conference 2011, Jeju Island, South Korea.
12. Kosko, G.. 2002. Fuzzy Cognitive Maps. International Journal of Man-Machine Studies,
13. Michalski, R. 1993. Inferential Theory of Learning as a Conceptual Basis for Multi-Strategy Learning. Machine Learning, 11:111-151.
14. Liao, S., He, J., and Tang, T. 2004. A framework for context information management. Journal of Information Science, vol. 30, pp. 528-539.
15. Ramamurti, V., and Ghosh, J. 1996. Advances in using Hierarchical Mixture of Experts for Signal Classification. Proceedings of ICASSP.
16. Stylios, C. and Groumpos, P. 2000. Fuzzy Cognitive Maps in Modeling Supervisory Control Systems. Journal of Intelligent & Fuzzy Systems 8, 2:83-98.

An Adaptive PID Tuning For LFC System Using Neuro-Fuzzy Inference System

Heidarali Shayanfar*, Ghasem Derakhshan
Center of Excellence for Power System Automation and Operation, College of Electrical Engineering
Iran University of Science and Technology, Iran
Email: Hashayanfar@yahoo.com, Gh_derakhshan@iust.ac.ir

Mohammadreza Alimohammadi, Amin Khodabakhshian
Electrical Engineering, University of Isfahan, Iran,
Email: Aminkh@eng.ui.ac.ir

Abstract— **A new control scheme, based on Artificial Neuro-Fuzzy Inference System (ANFIS) is used to design a robust Proportional Integral Derivative (PID) controller for Load Frequency Control (LFC). The controller algorithm is trained by the results of off-line studies obtained by using particle swarm optimization. The controller gains are optimized and updated in real-time according to load and parameters variations. Simulation results of this method on a multi-machine system in comparison with conventional fuzzy controller show the satisfactory results, especially where the parameters of the system change.**

Index Terms— **Load Frequency Control, Fuzzy Control, PID Controller, Particle Swarm Optimization**

I. Introduction

RECENTLY due to dependence of load efficiency to network frequency and being as one of the power quality indices, frequency control is one of the main issues in power systems. However, area load changes and abnormal conditions, such as outages of generation lead to have changes in frequency and scheduled power interchanges between areas [1]. Automatic Generation Control (AGC) (also named, load frequency control) is then widely used to balance between generated power and load demands in each control area in order to maintain the system frequency at nominal value and the power exchange between areas at its scheduled value. The concept of AGC in vertically integrated power systems is well discussed in reference [2]. In this paper different control techniques are presented. Among them PI (Proportional Integra) and PID controllers are widely used. Different techniques such as pole placement and bode diagrams are utilized to set their parameters [3].

Because of the inherent characteristics of the changing loads, the operating point of a power system changes continuously during a daily cycle. Thus, a fixed controller may no longer be suitable in all operating conditions. Therefore, a lot of approaches have been reported to solve this problem such as adaptive control [4], robust control [5], evolutionary algorithms based control [6], fuzzy logic control [7], neural network based control [8].

Intelligent algorithms have, in general, advantageous and disadvantageous when applied to power systems [7, 9, and 10]. Therefore, it is common to combine these techniques to overcome the main problems. In reference [7] membership function of a fuzzy PID controller is optimized by Particle Swarm Optimization (PSO) algorithm. However, this controller is designed at nominal operating condition and fails to provide best control performance over a wide range of operating conditions. In reference [9] genetic algorithm is used to optimize the PI gains for a number of operation conditions of power system. The obtained gains are used to train the ANFIS to provide optimal control gains. In reference [9] the frequency bias factor and synchronizing torque coefficients are used to train the ANFIS algorithm but load variation which is a main parameter has not been considered. In reference [10] PSO algorithm is used to generate the training sets by considering only load variation but ignoring others to train the ANFIS network.

In this paper PSO algorithm is used to optimize the gains of a PID controller for a number of operation conditions of power system. These optimal gains which are obtained offline will train the ANFIS by using Hybrid Learning Algorithm. Then, ANFIS provides a general mapping between the operation conditions and the optimal gains. In fact, ANFIS will tune the PID controller parameters online to cope with the changing power system conditions. The comparative results of the proposed PID-ANFIS with PI-ANFIS and conventional fuzzy controllers show that a much better dynamic performance is achieved.

II. System Model

Frequency changes occur because system load varies randomly throughout the day so that an exact forecast of real power demand cannot be assured. The imbalance between real power generation and load demand (plus losses) throughout the daily load cycle causes kinetic energy of rotation to be either added to or taken from the on-line generating units, and frequency throughout the interconnected system varies as a result. Each control area has a central facility called the energy control center, which monitors the system frequency and the actual power flows on its tie lines to neighboring areas.

* Corresponding Author. E-Mail: hashayanfar@gmail.com (H. A. Shayanfar)

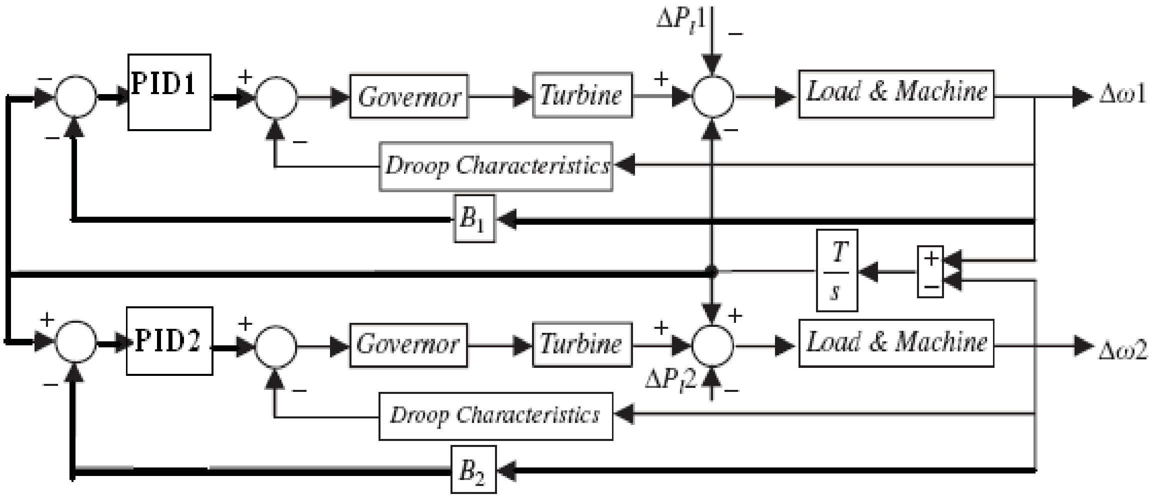


Fig. 1. Model of two area system.

The deviation between desired and actual system frequency is then combined with the deviation from the scheduled net interchange to form a composite measure called the area control error, or simply.

$$AGC_i = \beta\Delta f_i + \Delta P_{tie} \tag{1}$$

In general, for satisfactory operation of power units running in parallel it is most desirable to have the frequency and tie-line power fixed on their nominal and scheduled values even when the load alters and, therefore, to remove area control error ($AGC_i=0$).To help understand the control actions at the power plants for LFC, let us consider the boiler–turbine–generator combination of a thermal generating unit.

Since all the movements are small the frequency–power relation for turbine–governor control can be studied by a linearized block diagram [1]. However, the computer simulation will be carried out using the actual nonlinear system. The linear model is shown in Fig. 1 for a two-machine power system where the blocks are [1]:

$$Governer = \frac{1}{T_g s + 1}$$

$$Turbine = \frac{1}{T_h s + 1} \tag{2}$$

$$Generator = \frac{1}{Ms + D}$$

The uncontrolled system is shown with the continuous lines and the control loops are indicated by the bold lines. The parameters of the test system are given in Appendix A.

III. Fuzzy Based Controller Design

Fuzzy set theory and fuzzy logic establish the rules of a nonlinear mapping. Fuzzy control is based on a logical system called fuzzy logic which is much closer in spirit to human thinking and natural language than classical controllers. Because of the complexity and multi-variable conditions of the power system, conventional control methods may not give satisfactory solutions. The application of fuzzy logic to PID control design can be a simple solution and is classified in two major categories according to the way of their construction [7]:

1. A typical LFC is constructed as a set of heuristic control rules, and the control signal is directly deduced from the knowledge base.

2. The gains of the conventional PID controller are tuned on-line in terms of the knowledge based and fuzzy inference, and then, the conventional PID controller generates the control signal.

Fig. 2 shows the block diagram of fuzzy type controller to solve the LFC problem for each control area. The membership function sets for ACE_i, ΔACE_i, K_{Ii}, K_{di}, K_{pi}, and The appropriate rules for the proposed control strategy are given in Appendix B.

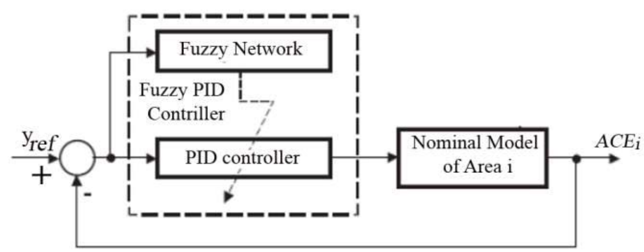


Fig. 2. The FPID controller design problem

IV. Sugeno Fuzzy Model

Unlike Mamdani model, Sugeno output membership functions are either linear or constant [11]. If a fuzzy system has two inputs *x* and *y* and one output *f*, then for a first order Sugeno fuzzy model, a common rule set with two fuzzy if then rules is as follows:

Rule 1: If x is A1 and y is B1, then f1 = p1x q1y+r1

Rule 1: If x is A2 and y is B2, then f2 = p2x q2y+r2

For a zero-order Sugeno model, the output level is a constant (*pi=qi=0*). The output level *fi* of each rule is weighted by the firing strength *wi* of the rule. For example, for an AND rule with *Input 1* = *x* and *Input 2* = *y*, the firing strength is:

wi = AND method *(A1(x), B1(y)).*

Where, *A1* and *B1* are the membership functions form *Input 1* and *Input 2,* respectively. The final output of the system is the weighted average of all rule outputs, computed as:

$$Final\ Output = \frac{\sum_{i=1}^{n} w_i f_i}{\sum_{i=1}^{n} w_i} \tag{3}$$

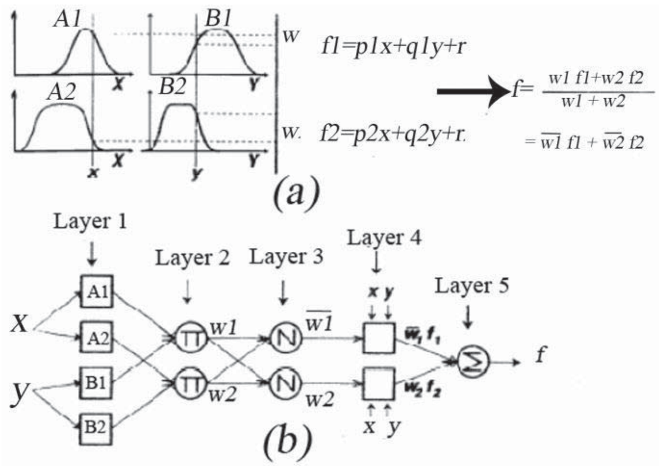


Fig. 3. (a) A two-input first-order Sugeno fuzzy model with two rules; (b) Equivalent ANFIS architecture.

V. ANFIS ARCHITECTURE

Fig.3 (a) illustrates the reasoning mechanism for the Sugeno model discussed above while the corresponding ANFIS architecture is shown in Fig. 3 (b), where nodes of the same layer have similar functions. The output of *ith* node in layer l is denoted as O^l_i [11].

Layer 1: Every node *i* in this layer is an adaptive node with a node function:

$$O_i^1 = M_{A_i}(x) \qquad i = 1,\ 2,\ 3, \dots \tag{4}$$

Where, *x* (or *y*) is the input to node *i* and A_i (or B_i) is a linguistic label "small" (or "large") associated with the node. Here the membership function for *A* (or *B*) can be any parameterized membership function. In this paper, generalized Gaussian membership function is taken as follows:

$$M_{A_i}(x) = e^{-(\frac{x-ci}{ai})^2} \tag{5}$$

Where, c_i and a_i are the parameters set. These are called premise parameters.

Layer 2: Every node in this layer is a fixed node labeled Π, whose output is the product of all the incoming signals.

$$O_i^2 = W_i = M_{A_i}(x) \times M_{B_i}(x) \qquad i = 1,\ 2, \dots \tag{6}$$

Layer 3: Here, the ith node calculates the ratio of the ith rule's firing strength to the sum of all rule's firing strengths.

$$O_i^3 = \overline{W}_1 = \frac{w_i}{w_1 + w_2} \qquad i = 1,\ 2 \tag{7}$$

Layer 4: Every node *i* in this layer is an adaptive node with a node function:

$$O_i^4 = \overline{w}_i f_i = \overline{w}_i (x p_i + y q_i + r_i) \tag{8}$$

Where, w_i is a normalized firing strength from layer 3 and $\{p_i,\ q_i,$ and $r_i\}$ is the parameter set of the node. These parameters are referred to as consequent parameters.

Layer 5: The single node in this layer is a fixed node labeled Σ, which computes the overall output as the summation of all incoming signals:

$$O_i^5 = Overall\ Output = \sum_i \overline{w}_i f_i = \frac{\sum w_i f_i}{\sum w_i} \tag{9}$$

In this paper, the hybrid learning algorithm is used for training of ANSIF networks. This algorithm is a combination of least square and back propagation method. Details of this algorithm have been published in reference [11].

VI. TRAINING PATTERNS

As mentioned earlier, fixed gain controllers are designed at nominal operating conditions and fail to provide best control performance over a wide range of operating conditions. It is desirable to keep system performance near its optimum. Therefore, the presented method tracks different system operating conditions and updates the gains of PID controllers according to loads and parameters variation. In this paper, the PSO is used for preparing the training patterns. Details of this algorithm have published in reference [12]. Thus, the objective function of the optimization of the performance of the system is defined as:

$$Fitness\ Function = \int_0^\infty (\alpha P_{tie} + \beta(\Delta f_1 + \Delta f_2))dt \tag{10}$$

The coefficients *α*, *β* and *γ* are assumed to be 1. Now, the optimal values of PID controller gains should be calculated for area load variation and synchronizing torque coefficients in order to minimize Equation (10) by using PSO technique.

The fitness function is evaluated and the population is updated iteratively until the stopping criterion is obtained. The first step in using ANFIS is preparing training patterns to train it. In this paper, the training data set consists of several disturbances. These disturbances are combinations of the load variations in the range of −0.1 to 0.1 pu and synchronizing torque coefficient variation in the range 0f 1.5 to 3. Therefore, for each load disturbance (ΔP_{l1} , ΔP_{l2}), eleven values in range of -0.1 to 0.1 with 0.02 step are considered. Four values of 1.5, 2, 2.5, and 3 are also considered for Torque Coefficient variation. Finally, the training set consists of 484 elements.

Then, the optimal gains of PID controllers are computed for each of these disturbances using PSO method and the objective function given in Equation (10). After gathering these data, it is possible to train ANFIS. For each gain, one neuro-fuzzy network is trained. After computing the optimal gains, two matrices with 484 rows and 6 columns (three columns for input variation and the three columns representing the optimal gains), and one matrix for each controller, is fed to ANFIS for training.

VII. Proposed Control Approach

When the training is done, the two trained neuro-fuzzy networks can estimate the optimal gains for any disturbances taken place in the power system and online control can be implemented. It should be noted that the efficiency of ANFIS can be enhanced by choosing a bigger training set. The proposed method can be applied to small and intermediate power systems easily. However, although all of the time-consuming computations are carried out offline, for large and very large power systems, computation time will be somehow exorbitant and more powerful computers or advanced computing techniques will be necessary to be used.

The test system shown in Fig. 4 is used to illustrate the behavior of the proposed method. For ANFIS, the settings to be used were Gaussian membership function, four membership functions for load variation, three membership functions for parameter variation, hybrid optimization method and epoch = 15. PSO settings were as follows: population size = 50, $c_1 = c_2 = 2$, $\omega_{max} = 0.9$, $\omega_{min} = 0.4$, *maximum speed* = 2. The proposed controller is a two-level controller. The first level is ANFIS network and the second one is PID controller.

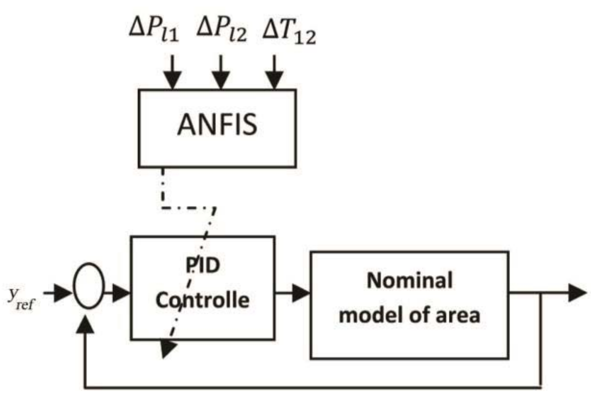


Fig. 4. The proposed ANFISPID controller for each area.

The structure of the ANFISPID controller is shown in Fig. 4, where the PID controller gains are tuned online for each of the control areas.

Therefore, u_i is a control signal that applies to governor set point in each area. By taking ACE_i as the system output, the control vector for a conventional PID controller is given by:

$$u_i = Kp_i \ ACE_i(t) + Ki_i \int_0^t ACE_i(t)dt + Kd_i \ ACE_i(t) \quad (11)$$

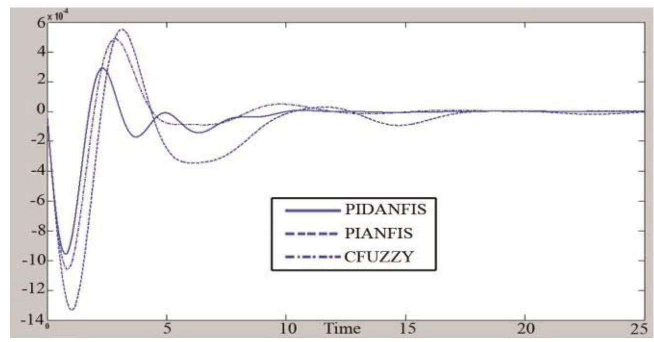
In this strategy, the gains Kp_i, Ki_i, and Kd_i in Equation (11) are tuned on-line based on load variation and Synchronizing Torque Coefficient variation. The PIANFIS is similar to PIDANFIS but $Kd_i = 0$.

VIII. Simulation Results

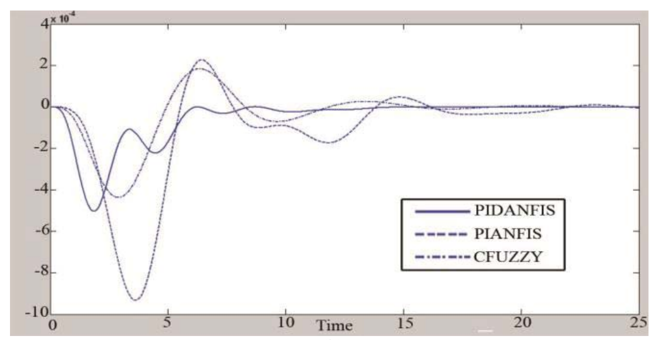
In order to validate the proposed controller for load frequency control three different cases are considered and are given as follows;

A. Case 1:

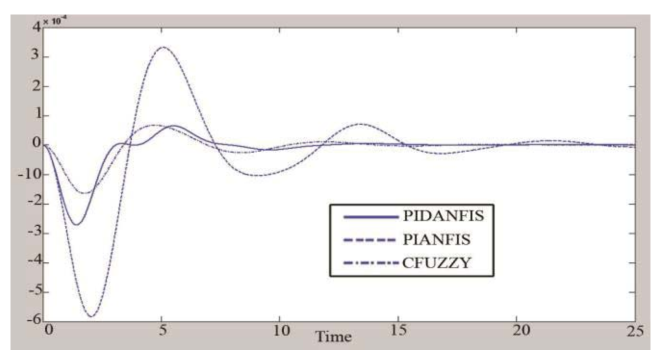
In this case, the following load variations are considered: $\Delta P_{Li}=0.02$, $\Delta T_{L2}=0.02$ and no disturbances occur in area II. From Fig. 5 (a) and Fig. 5 (b) can been seeing that PIDANFIS according to nominal operation point, improves the dynamic performance.



(a)



(b)

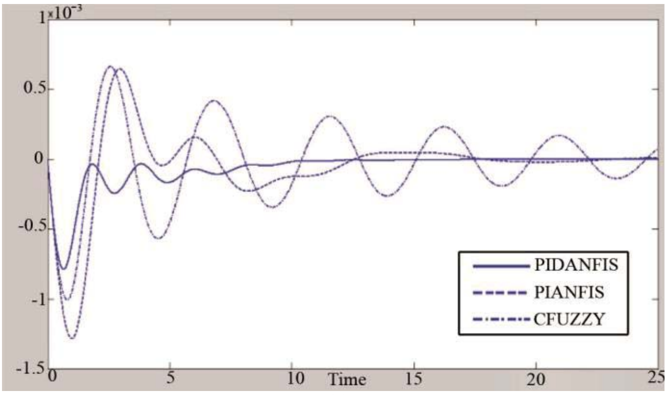


(c)

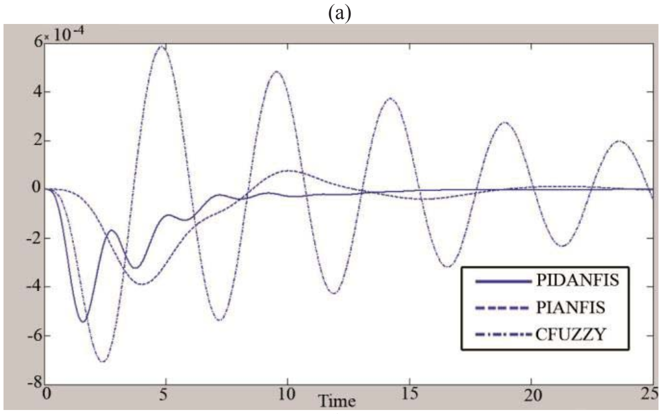
Fig. 5. System response for case 1. (a) $\Delta\omega_1$. (b) $\Delta\omega_2$. (c) ΔPtie.
(Solid: PIDANFIS, Dashed: PIANFIS, Doted: Conventional fuzzy)

B. *Case 2:*

In this case, the synchronizing torque coefficient changes to 2.5, but the demand load is similar to the case 1. The simulation results depicted in Fig. 6 show that system with conventional fuzzy controller fails to provide a desirable performance when parameters vary, because the conventional fuzzy controller was designed based on T_{l2}=2.



(a)



(b)

Fig. 6. System response for case 2. (a) $\Delta\omega_1$. (b) $\Delta\omega_2$.
(Solid: PIDANFIS, Dashed: PIANFIS, Doted: Conventional fuzzy)

C. *Case 3:*

In this case, a bounded random step load change shown in Fig. 7 appears in each control area. The purpose of this case is to test the robustness of the proposed controller against random load disturbances. The frequency deviations of both systems for nominal operating condition (case 1) are shown in Fig. 8 (a) and (b).

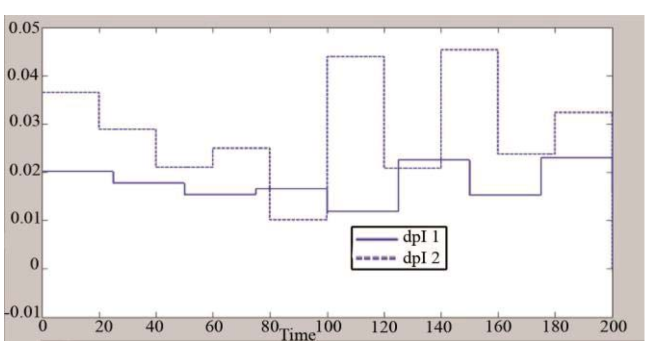
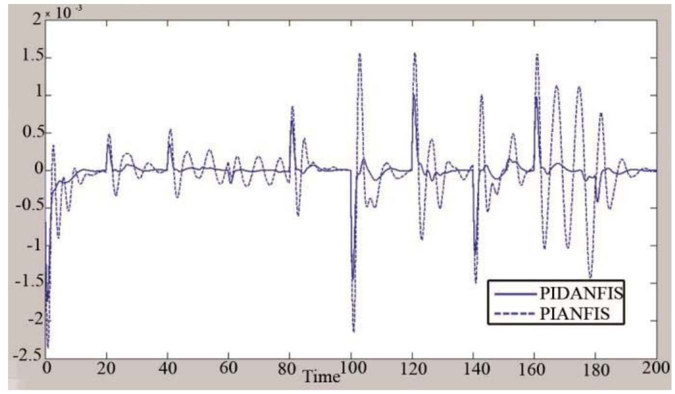
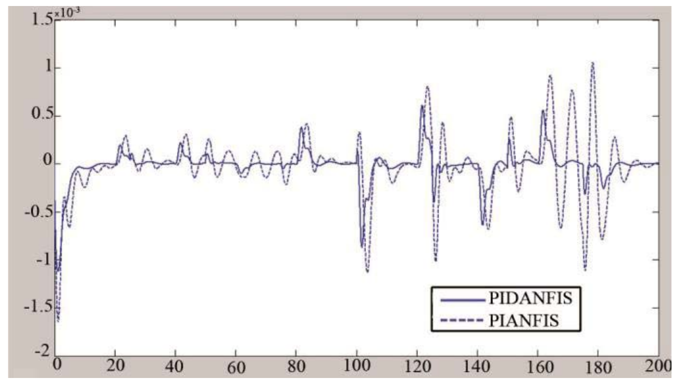


Fig. 7. Load disturbances in two areas.



(a)



(b)

Fig. 8. Frequency deviation of two areas.
(Solid: PIDANFIS, Dashed: PIANSIS)

From Fig. 8 (a) and Fig. 8 (b), it can be seen that the ANFISPID controller tracks the load fluctuations and for a wide range of operating conditions.

IX. CONCLUSION

Optimization of the PID gains for a two-area load frequency controller using PSO algorithms has been proposed. Such optimization technique has the advantage of being systematic and weakly dependent on the model. But, the time consumed for computing optimal gains using PSO directly is too much for real-time control and is not practical. However, the trained ANFIS response time is reasonable and practical.

The gains for 484 operating conditions were used to train an adaptive neuro-fuzzy inference system in order to obtain a general mapping between the operating condition and optimal PID gains. A comparison between the proposed schemes, PI controller based on ANFIS and conventional fuzzy controller revealed that the system performance can be improved.

APPENDICES

- *Appendices A:*

The system parameters are as follows (frequency = 60Hz, base = 1000 MVA):
System #1:
H =5; D = 0.8; Tg = 0.2; Th = 0.5; R = 0.05; β=20.8
System #2:
H = 4; D = 0.9; Tg = 0.3; T_h = 0.6; R = 0.0625; β=16.9

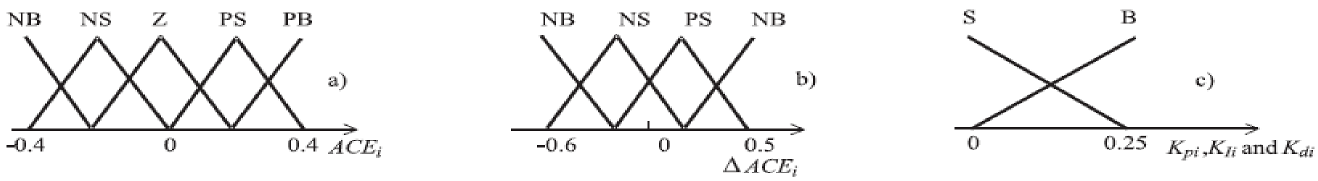


Fig. 9. a) Membership for ACEi , b) Membership for ΔACEi , c) Membership for K_{Ii} , K_{Pi} and K_{di}

- *Appendices B:*

The membership function sets for ACE_i, ΔACE_i, K_{Ii}, K_{di} and K_{pi} are shown in Fig. 9. The appropriate rules for the proposed control strategy are given in TABELS. I and II.

TABLE I
Rule table for K_{Ii}

ΔACE	NB	NS	PS	PB
NB	S	S	S	S
NS	S	B	B	S
Z	B	B	B	B
PS	S	B	B	S
PB	S	S	S	S

TABLE II
Rule table for K_{di} and K_{pi}

ΔACE	NB	NS	PS	PB
NB	B	B	B	B
NS	B	S	S	B
Z	S	S	S	S
PS	B	S	S	B
PB	B	B	B	B

References

[1] P. Kundur, "*Power System Stability and Control*". New York: McGraw-Hill, 1994.

[2] O. I. Elgerd and C. Fosha, *"Optimum Megawatt-Frequency Control of Multi Area Electric Energy Systems", IEEE Trans. Power App. Syst., V*ol. 89 (4), pp. 556–563, 1970.

[3] K. A. Sattar*, "An Improved Automatic Load Frequency Controller for Damping Low Power Oscillations",* Universities Power Engineering Conference, Vol. 2, pp. 496 - 499, 2006.

[4] J. Kanniah, S. C. Tripathy, O. P. Malik, and G. S. Hope, *"Microprocessor Based Adaptive Load-Frequency Control"*, IEE Proceedings, Generation, Transmission and Distribution, Vol. 131, No. 4, July 1984.

[5] K. Lim, Y. Wang, and R. Zhou, *"Robust Decentralized Load Frequency Control of Multi-Area Power Systems",* IEE Proceeding, Generation, Transmission and Distribution, Vol. 143, No. 5, September 1996.

[6] Z. Al-Hamouz and H. Al-Duwaish , *"A New Load Frequency Variable Structure Controller Using Genetic Algorithms,"* Electr Power Syst Res, pp. 1–6, 2000.

[7] H. Shayeghi , A. Jalili , and H. A. Shayanfar, *"Multi-Stage Fuzzy Load Frequency Control Using PSO"* , Energy Conversion and Management, 2008.

[8] D. K. Chaturvedi, P.S. Satsangi, P.K. Kalra, "*Load Frequency Control: A Generalized Neural Network Approach"*, Electrical Power Energy Syst. 21, 405–415, 1999.

[9] A. Abdennour, *"Adaptive Optimal Gain Scheduling for The Load Frequency Control Problem",* Electric Power Components and Systems, Vol. 30, pp. 45-56, 2002.

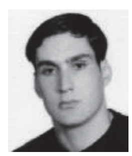
[10] S. H. Hosseini and A. H. Etemadi, *"Adaptive Neuro-Fuzzy Inference System Based Automatic Generation Control"*, Electric Power Systems Research, pp. 1230–1239, 2008.

[11] J. S. Jang, *"ANFIS: Adaptive-Network-Based Fuzzy Inference System"*, IEEE Trans. Syst. Man Cybern, Vol. 23 (3), pp. 665–685, 1993.

[12] Y. Shi and R.C. Eberhart, *"A Modified Particle Swarm Optimizer",* IEEE Press. Internationnal Conferance on Evolutionary Computation, pp. 69-73, May 1998.

Biographies

Heidar Ali Shayanfar received his B.S. and M.S. degrees in Electrical Engineering in 1973 and 1979 and the Ph.D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, He is a Full Professor in Electrical Engineering Department of IUST, Tehran, Iran. His research interests are in the application of artificial intelligence to power system control design, area of smart grid, dynamic load modeling, power system observability studies and voltage collapse. He has published more than 212 Journal papers and 285 conference papers. He is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.



Ghasem Derakhshan received his B.S. and M.S. degrees in Electrical Engineering in 2006 and 2009 from Power and Water Institute of Technology (PWIT) and Iran University of Science and Technology (IUST), respectively and currently he is Ph.D. student at IUST. His research interests are in the area of smart grid and power system problems.

Mohammadreza Alimohammadi received his B.S. and M.S. degrees in Electrical Engineering in 2005 and 2010 from Azad University and University of Isfahan. His area of research is digital control systems applied to power system problems.

Amin Khodabakhshian was born in Isfahan. He obtained his B.Sc. degree in 1984 and his M.Sc. in 1987 from the University of Tehran (Iran) and the Ph. D. degree in the Department of EE at the University of Wollongong, Australia. Currently, He is a assistant of professor in Electrical Engineering Department of university of Isfahan, Iran. His area of research is digital control systems applied to power system problems.

The logic Fuzzy in the operation of the knowledge of the interaction human-computer tasks in complex machines: Learning in significance sets.

E. Martins[1], **I. Martins**[2]
[1]Department of Design Pos-graduation, Federal University of Pernambuco, Recife, Pernambuco, Brazil
[2]E-learning Department, University Estácio de Sá, Rio de Janeiro, RJ, Brazil

Abstract - *The Two machines: The Human and the Computer, one biological and other mechanical/ electronics that use and implement their natural computational logic in sharing the tasks for controlling complex machines. Both components (Human and Computer) in charge of the control and operation of complex machines need to learn their tasks each in its own way. But questions arise: The format of the training, learning objects and the training process are correct? The operationalization of the knowledge showed satisfactory? This article presents a critical analysis of learning processes and the operation for the joint work of these two components, where supposedly the symbiosis in the distribution of tasks is proposed to extract the greatest advantages of each process. The great advantage of man associated with the computer appears as a large capacity for recognition and processing qualitative information, that is often not covered in computer systems with their great ability to process a large amount of data very fast and accurate.*

Keywords: Fuzzy logic, New technologies; Automation; Human error.

1 Introduction

The computer will always proceed in according to circumstances stated in programming that determines, in the final instance, the decisions of cyber component incorporated into automated command of an aircraft, for example. The theory assumes that the two different capacities are complementary in processing information. The literature on the state of the art in aviation indicates that the expectation of the responsible for automating aircraft are translated in the combination of the capabilities of the human and the computer to integrate like "a perfect symbiosis to conduct aircraft". This article is a critical analysis of the learning processes and the implementation of the joint work of these two components, where supposedly the symbiosis in the distribution of tasks is proposed to extract the greatest mutual benefit. Specific prior knowledge of aircraft piloting are converted into machines procedures that operationalize the distributed actions supported by onboard computers. On this scenery, are included actions on critical moments in the aviation field in automated aircrafts. The perfect harmonic and unrealistic combination of human and computer in the control of aircraft is symbolized at figure 1.

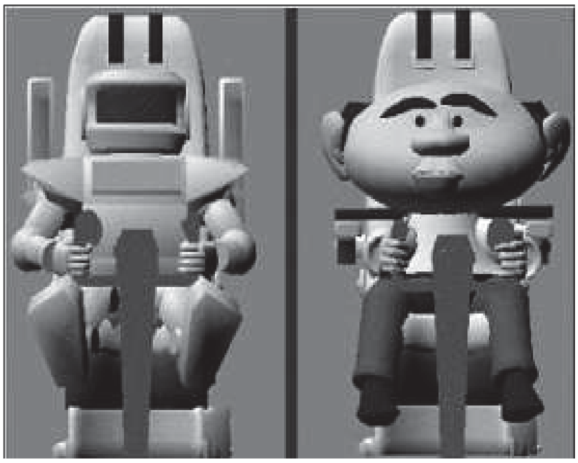


Fig. 1 The pilot and the computer piloting a highly automated aircraft (*Fly-by-Wire*)

The Human actions are anchored in subsumers where the actions are supported by computers. This "knowledgement" is converted into procedures commanded by programs and computers. A part of this program is used to assist pilots in controlling these aircraft and that also has its "subsumers", as Ausubel definition [1] , anchored in historical situations that reflect best piloting techniques. But also includes prior knowledge of risk situations and when they happen, are evaluated by computers, "defending" and blocking the aircraft unplanned procedures performed by pilots and that can lead to accidents. The logic normally used in the construction and programming these computers is the logic whose characteristic is boolean, binary, Cartesian, univalent, mutually exclusive and unirelacional. This provides accurattion and the final product is the review of the results presented by these machines, but the human being learns how to fly the aircraft and operates this knowledge in the same way that the computer looks into cartesian pattern where subsumers are transformed into action that is originate on the mental structures previously existing? The concept that establishes control of binomial Human-Computer at applied ergonomics on distributed system exhausts the possibilities of appropriate actions to prevent some rare existing accidents but it provides all critical situations becoming failsafe? The pilot's actions, while aided by computers, when make mistakes can be quickly corrected by computers and vice-versa? We know from recent history of accidents in aviation that the answer is NO. The problem always lies, for various reasons, including convenience and lack of a culprit in human error. We formulate a hypothesis across this question: The fuzzy logic applied in the "learning" of the on-board computers aircraft translated into your schedule might be a good solution to mitigate of the "mind" asynchronism between human and computers? If we consider the following three quotes, we can weave a doubtful scenario that allows failures behavior of harmony in the human work together and computers.

These statements below begin to establish similarities and differences of the human mind related to computers and reinforce the need for a possible change in the learning scenario of the two. A progress already achieved in the scientific community is the use of fuzzy logic.

The advantage of this tool, is adopt the use of the continuous intermediate values between 0 and 1 (or true and false). In a way, with fuzzy logic it is possible to devise a more flexible computing closer to reality of the human mind, and this is certainly a great advance. Still, one can not compute contradictions, although this logic allows transactions involving ambiguities, vaguidez, inaccuracies, noise and incomplete inputs. Fig. 2 metaphorizes the help of ergonomics distributed human-computer, as designed in beginning of the process of automation.

Fig. 2-The automatized help, implemented to prevent failure of human

Automated systems sometimes have left the pilots at risk for not knowing that the automatic attitudes controls are taking at any time as drawn in Figure 3.

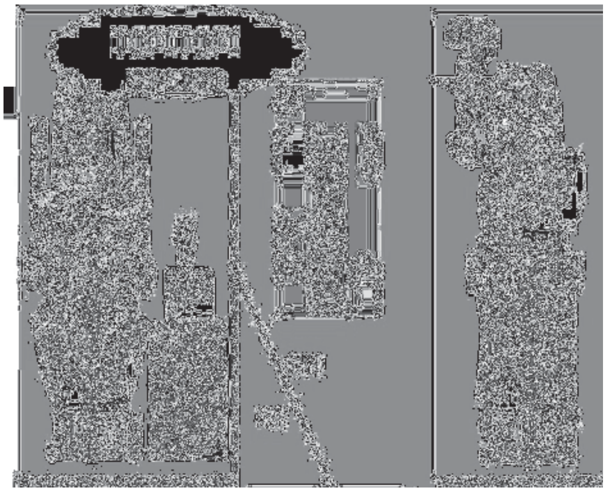


Fig.3- Often the pilot does not "know" what the computer is doing.

2 Fundamentals

The operation is built to be operated by humans and the computer in the control of automated aircraft but as acts each component? In automated aircraft, when a new component is installed, the corresponding computer to solidarity sensors and semantic establishes a new network of interrelationships between crew and aircraft environment. Command decisions are then redistributed between man and machine (computer / aircraft). Thus we have a symbiosis between different and complementary skills in information processing and command of the aircraft. But analyzing this operation working together and we will mention some serious errors, referencing accident cases where this type of task distribution failed. Initially we need to understand the operation of each "component": The human and the computer.

2.1 Classical logic, sequential processing and parallel and artificial intelligence

There are some theories about how the mind works but we can not prove that automation and electronic computing are similar from the standpoint of the nature of mental phenomena. Even the Cognitive Science, settled in bias of classical computing, does not establish the similarities necessary to consolidate this theory. The digital computer developed by Von Neumann was derived mainly from machine called Colossus, designed by Allan Turing during World War II. And this, in turn, is settled in the logic function. The mono-processor computing using procedural languages are based on classical logic and show the typical features of a logical / sequential system.

The technique called parallel processing on a single computer, which is still incipient and currently crawls in the area of business applications (if we do not consider the processing of applications deployed in distributed computer networks), we could have a slight correspondence with the human mind. The possible similarity with human beings was implemented by John Von Neumann in 1946-1952 on the machine called *IAS*. Neumann's project is invaluable in computational projects and was a major scientific breakthrough in the post-war. Promoted all this electronic scientific supported by electronic actuators represented today by micro-computer and personal micro-computers. But it is a sequential machine whose logic is binary, Cartesian, Boolean.

These computers are perfect and remarkable for accuracy but useless to establish absolute similarities with the workings of the mind before nebulous situations, inaccurate, unclear, not perfect, not visible or where it is required to process unplanned or insufficient information to establish decision-making in critical situations or that represent real and present danger. The big computer's ability to assist humans in almost all procedural areas of science obliterated his vision of the true difference of the human mind in the operation of knowledge structures in comparison with the formal, fast and precise applications that computers promote.

The man, impressed with the high ample opportunities for processing of computers is taken to perform a reverse engineering with the mind of the human, comparing it and establishing a parallel operation, with propriety and competence, as Pinker cites (2000) in their theories. Pinker does the reverse work of John Von Newman (when he developed the design of your computer by copying characteristics of the human being). Pinker presents the computational characteristics of the mind pointing, appropriately, for schedules and similar biological structures to electronic systems. But the branch of artificial intelligence is facing severe criticism and lines of work just because the mind is a hardened bunker with his secrets of its operation.

The various lines of interpretation and mind functioning of theories have been made since the last Decartes Pinker by Piaget, Vygotsky and Vernaut among many notable and obstinate scientists who have devoted themselves to the study of mind, intelligence, knowledge and training of the senses and language. What we will focus in this paper is mainly the implementation of this knowledge in the form of treasury stock and objective as a human and computer controlling complex machines in critical and unexpected situations.

2.2 As each component (human and computer) "learns" your own tasks.

Classical logic has a very close relationship with the natural language. However, some characteristics of natural language is not suitable for a formal procedure. For example, natural language is permeated with contradictions. Therefore Frege, founder of modern logic, sought the development of a more economical and accurate artificial language (unambiguous), A formal system must submit: (1) any set of symbols, or alphabet; (2) a set of terms "well-formed"; (3) a set of axioms; a finite set of rules.

Furthermore, the classical logic of computers principle works with two values: true and false. Thus, a predicate can be true or false, but never both true and false. In this classic logic there is concern that an expression is a really true or not for science or philosophy. Their procedures work independent of veracity. In other words, what is in the grip of this logic are formal procedures for from premises and achieve a result. The correspondence between this result and something external to the logic itself is not a matter that the LC is proposed. The AI is usually based on the LC to generate a model of the functioning of the mind, and in that sense the Allan Turing machine and Tommy Flowers (Colossus) is an abstract logical model of the mind. But you can ask to what extent this model is really appropriate. Or, what aspects of the human mind are evident through this model. The answer to these questions involve not only philosophical, but also computer and, if so, logic. The Colossus, precursor of computers, aimed to "learn" the war deciphers codes.

The difference between humans and machines is observed when each "learns" to solve problems The computer is an electronic machine that, to "learn", requires the installation of a program, which is a logical series of operations, today developed and installed by the Human. The man, in turn, requires that, theoretically, occur a series of connections and subsumers conjunction [2].

Learning object translates into information symbols and formatted signals are re-set in bits (binary digit- bit or binary digit is the smallest unit of information stored and / or handled on computers) of material (neurons), which is transmitted in the form of connection patterns and activities of the neurons [3]. In this way no information is lost when migrates to another physical station, such as oral information that is formatted in sound patterns transmitted through the air to the auditory system which switches back of material. At the end, processing returns to the brain through the neuronal activity.

The symbols formed by this same brain-mind are not just the result of an entry / internal representation, from the senses. Are symbols which may contain, in addition to the representational information causal properties, which means that both contain information and are part of a chain of physical events, or can generate information and / or actions.

Then, the information bits processed by the human brain can trigger other mind-bit symbol components to produce sense: validate or not information (true or false, which will form the individual set of beliefs); or can trigger bits connected with muscles, resulting in movement. Thus, mental computation is complex and enables the combination of processes involving, for example, a symbol processed under given set of rules, event triggers a mechanical (or electronic, as with an actual computer, or a programmable automaton to perform functions, or, as I thought Alan Turing in 1937, happen to a processor symbols able to read symbols and operate from a fixed set of rules). If an artificial system is based on the classical computer that operates from the classical logic, you can order such a system performing a given task. For example, determine the time needed to complete a flight in an aircraft, considering wind, aircraft speed and other conditions. Such a proposal involves subtleties which, perhaps, a machine can not compute. At first, the machine would calculate the distance traveled faster and more accurately than a human being to drive from Sao Paulo to Tokyo. On the other hand, a human could answer that it is not possible to drive from Sao Paulo to Tokyo. This example, although trivial, shows some interesting differences in the way of handling problems for humans and machines. Humans tend to consider the truth of the premises they work. Not that prevent the implementation of a purely formal, but whether or not to the truth of the premises can significantly change the relationship between individual and problem [4].

According to a perspective of Vygotsky, proven empirically by Luria, there are different steps for troubleshooting by humans [5]. First, there is the location of the problem to the historical and social world of the individual. The human being creates a conceptual framework (frame) that can treat the data in question, usually using the language as a control tool. It is a "question of how the non-monotonic reasoning, such as adding information affecting the state supporting the conclusions, is able to inhibit or discard options inferences" [6]. Secondly, the individual performs the actual computing activities. He makes the calculation in question through a formal procedure.

The importance of Vygotsky's description is to put the context (external) socio-historical perspective in the resolution of the problem, and this means that before any formal logic and computing the individual will probably check the relevance and veracity of the premises. In the case of processors supported by procedural systems, such as occurs in Artificial Intelligence, there is only the logical computation. An artificial system is not, in principle, able to establish socio-historical relations. That is, it is unable to locate the problem in an individual socio-historical context. Even so, a system could even conclude that it is not possible to travel by car from New York to Tokyo, but such a solution would result in a more complete computer programming, and not of verifying the relevance of assumptions.

It is possible to simulate an artificial system, and efficiently procedures performed by humans. But this does not seem sufficient to explain the workings of the mind, as the computer continues to work just syntactic operations without verifying the relevance of its premises and conclusions. This model is a good tool for better understanding of the nature of mental processes, an artifact that allows empirically test hypotheses and theories about the mind and reproduces certain parts of its operation, particularly its logical-formal reasoning, its operation deductive . But on the other hand, it is difficult to argue that this model is possessor of a mind just like the human mind due to the limits of classical logic and non-consideration of other possible types of reasoning to be realized.

Anyway, on the one hand the difference consists in the fact human verify the relevance of the assumptions that logic works, their relationship with respect to what is external to itself. On the other hand, the problem is internal to the logic limited to a formal logic which does not allow ambiguities or inconsistencies. In this sense, the use of a logic allowing greater proximity to the human mind and its natural language has great interest. In particular a logic that allows intermediate states between the true and the false, and even allowing the emergence of contradictions. There is a structured paradigm called on Computational Theory of Mind, assuming for the functioning of the human mind the nature of the computational process information in the form of symbols [7].

The computational process is associated with the human brain ability to mentally represent knowledge (visual representation, phonological, grammatical and an internal mental language of the human being) in complex layers and interrelated associative networks of meanings. This peculiar biological-human species informational system is also equipped with processing systems of rules that would be infinitely more flexible than those rules comprise any conventional computer program, which enable not only the categorization precise knowledge and / or probabilistic (fuzzy) logic abstract but also, for example, allowing the human being to recognize a face or even the sense of individuality.

This powerful genetically improved software also functions to keep the mental representation, preserving the relations of exact or probabilistic true that formed the alleged true relationship observed in reality, the first time that the brain operated on that symbol. "These events are a computer, because the mechanism was devised so that if the interpretation of symbols that trigger the machine is a true statement, the interpretation of symbols created by the machine will also be a true statement." [8].

Characteristics of Fuzzy Logic and Fuzzy:

- Bivalence is, Since Aristotle, a classical logic and is based on bivalence V, F - (Non-Contradiction:

- Multivalency was developed by Lukasiewicz to deal with the uncertainty principle in mechanics. Quantum V, F, IN - 1920-3 values / 1930 - n values

- • Fuzzy Logic and developed by Lotfi Zadeh [9] (Fuzzy Sets) where elements belong to a certain set with different degrees (degree of ASSOCIATION, relevance)[10]

- Aristotle (384-322 BC) was the first scholar to make a representation of the thought process, through the systematization of logical reasoning.

Commercial Applications:

• Control: Aircraft Control (Rockwell Corp.), Sendai Subway Operation (Hitachi), Automatic Transmission (Nissan, Subaru), Space Shuttle Docking (NASA)

• Optimization and Planning: Elevators (Hitachi, Fujitech, Mitsubishi), Stock Market Analysis (Yamaichi)

• Signal Analysis: TV Image Adjustment (Sony), Autofocus for Video Camera (Canon), Video Image Stabilizer (Panasonic)[11].

An application of fuzzy logic is the construction of fuzzy systems, which are input compounds expert systems and output figures, fuzzification method, fuzzy rules, fuzzy inference and defuzzification method [12]. These rules can be combined by logical connectives such as AND and OR. The inference engine defines the way of how the rules are combined, providing a basis for decision-making [13] Mendel 1995]. There are many inferential procedures on fuzzy logic, but the most used are Mamdani and Takagi-Sugeno-the Kangl.

Fuzzification is a domain mapping of real numbers - discrete, in general - to the fuzzy domain. This process assigns linguistic values defined by membership functions, the input variables [14].

3 Method

Defuzzification is a method used to decode the linguistic variables output inferred by fuzzy rules for real values - generally discreet. This method is the Central of the Maximum the Middle-of-Max and the Middle-Weighted [15]. Fuzzy logic allows to obtain greater generality, higher expressive power, ability to model complex problems, modeling expert knowledge systems, manipulate uncertainties and complexities reduce problems [16]. Because of its properties, fuzzy logic has been used in areas such as: Expert Systems, Computing with Words, Approximate Reasoning, Natural Language, Process Control, Robotics, Systems Modeling Partially Open, Pattern Recognition, Decision Making Process (Decision Making), among others.

Expert systems are knowledge-intensive computer programs obtained by the expertise of specialists in limited areas of knowledge. They can help in decision-making raising relevant questions and explaining the reasons for taking certain actions. Traditional programs require accuracy and precision (also called crisp): on or off, yes or no, right or wrong. But this does not happen in the real world. We know that 50 ° C is hot and -40^{o} C is cold; but 23^{O}C is hot, warm, mild or cold? The answer depends on factors such as wind, humidity, personal experience and the clothes that each is using. The basic Idea: Fuzzy sets are functions that map a scalar value in a number between 0 and 1, which indicates its pertinence to this set.

4 Conclusion

The results of this model shows that it is feasible to create tools based on fuzzy logic to aid decision making in Air Traffic Control, provided that these tools can be widely tested and validated. This model has a considerably high risk index (6.7%) for safety standards in Air Traffic Control. Thus, the group of experts approved with reservations using the tool, requiring a model of development for it to become fully operational. However, a point also being analyzed is the difficulty of human beings to use and rely on these types of tools for highly dynamic environments, complex and risky. The acquired perception was that this tool can not replace the human air traffic controller, due to various factors related to the experience of human beings and that are not currently sufficiently reflected in computational tools.

5 Applications

In this work were carried out only sixty analyzes the applicability of the model, quantity judged as insufficient to have an effective diagnosis of the model. This requirement is supported by Mcneill and Thro [16] that is indicating the need for greater dedication to the test phases, simulation and validation of fuzzy systems. The main limitation of this study is that only certain types of aircraft are included. For operation of other types of aircraft (such as Airbus 340, Boeing 747, Tucano, Navajo, Rio, Brasilia, Buffalo, etc.) a new model is needed, because the controlling decision-making are different for different aircraft (the performances are different). As a suggestion for further work, designs may be implemented to include a greater number of aircraft types as well as a simulator to test the model for testing the operational environment involving a great risk for air traffic control operations [17].

The use of type two, Fuzzy Logic is also a relevant proposal to be studied in the proposed scenario. This new concept allows you to handle greater depth the uncertainties of expert knowledge for modeling systems such as imprecise limits of fuzzy sets, uncertainty regarding the degree of membership, and uncertainty format or some parameters of the membership functions [18].

Another possibility is to build a model that uses more input variables and analyze the performance in the system results, and analysis of the computational cost for the proposed model. Some candidate's variables have been identified, such as moisture content of the track, adverse wind conditions, cloud height, other types of aircraft, among others, as these factors also affect the decision scenario.

Finally, another possibility would be to implement a mature and thoroughly tested model, since the model of this project can be considered somewhat incipient, because it does not address all possible variables used for decision making. For a more complete and autonomous model, it is probable that its implementation can be used as the input variables, identified herein, may be collected electronically using radar systems, sensors and existing information systems in air traffic operations.

6 References

[1] AUSUBEL, Novak e Hanesian - Educational Psychology: *A Cognítíve View,* Editora Rinehart and Winston, New York (1978, 2013 2.edit.).

[2] AUSUBEL, D. P. A aprendizagem significativa: a teoria de David Ausubel. São Paulo: Moraes, 1982. AUSUBEL, D.P., Novak, J.D. and Hanesian, H. (1978). *Educational psychology.* New York: Holt, Rinehart and Winston. Editora Interamericana, Rio de Janeiro, 1980. Em espanhol por Editorial Trillas, México, 1981. English by Werbel & Peck, NewYork, 1986.
Mendel, Jerry M. (1995) "Fuzzy Logic Systems for Engineerings: A Tutorial", IEEE. Professional.

[3] Pinker, Steven – Como a mente funciona. Editora Edgard Buker – São Paulo, 2010

[4] AMORIM, C.A, Márcio Pereira Lima- ARTIGO- *Proposta de um modelo fuzzy para apoio à tomada de decisão no Controle de Tráfego Aéreo do Aeroporto Internacional de Salvador- 1Departamento de Ciências Exatas e da Terra –* Colegiado de Análise de Sistemas – Universidade do Estado da Bahia (UNEB) (2009)

[5] FRAWLEY, W. *Vygotsky e a Ciência Cognitiva*. Porto Alegre: Artemed, 2000.

[6] GERSHENSON, C. *Lógica multidimensional:* un modelo de lógica paraconsistente. In: Memorias del XI Congreso Nacional ANIEI, p. 132- 141, 1998.

[7] WADLER, P. *Proofs are Programs: 19th Century Logic and 21st Century Computing*. 2000

[8] Márcio Pereira Lima1, Cláudio Alves Amorim **(** 2005)- Proposta de um modelo fuzzy para apoio à tomada de decisão no Controle de Tráfego Aéreo do Aeroporto Internacional de Salvador. Departamento de Ciências Exatas e da Terra – Colegiado de Análise de Sistemas – Universidade do Estado da Bahia (UNEB) (2009)

[9] ZADEH, L. A. *Toward a theory of fuzzy systems*. In: KALMAN, R. E. e DeCLARIS, N. (Eds). Aspects of network and system theory. New York: Holt, Rinehart and Winston, p. 469-490, 1971. ‹http://www.research.avayalabs.com/user/wadler/.html›. Acessed 03 december 2012.

[10] ZADEH, L. A. Fuzzy sets. *Information and Control,* v. 8, p. 338-53, 1965.

[11] OLIVEIRA, Luis F.- A Flexibilização da Lógica em Direção a uma Melhor Modelagem da Mente pela IA– Artigo- Programa de Pós-graduação em Filosofia, área de concentração em Ciência Cognitiva e Filosofia da Mente. Faculdade de Filosofia e Ciências UNESP (2008)

[12] Kasabov, Nikola K "Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering", 2. ed. Londres: Massachusetts Institute of Technology. . (1998)

[13] Mendel, Jerry M.) "Uncertain rule-based fuzzy logic systems: introduction and new directions". Londres: Prentice Hall P. T. R. (2001, 2011)

[14] Shaw, Ian S. e Simões, Marcelo Godoy "Controle e Modelagem Fuzzy", 1. ed. São Paulo: Edgard Blücher Ltda. (1999)

[15] The Mathworks Inc (1999) "Fuzzy Logic Toolbox for use in Matlab User's Guide", 2. versão. http://www.mathworks.com/fuzzy/fp351dup8.html, Janeiro, 2006.

[16] Klir, George J. e Yuan, Bo. (1995) "Fuzzy Sets and Fuzzy Logic: Theory and Applications". Londres: Prentice Hall P. T. R.
Mcneill, F. M. e Thro, Ellen. (2012) "Fuzzy Logic: a practical approach", Londres: AP

[17] E. Martins, “Study of the implications for health and work in the operationalization and the aeronaut embedded in modern aircraft in the man-machines interactive process complex,” “Estudo das implicações na saúde e na operacionalização e no trabalho do aeronauta embarcado em modernas aeronaves no processo interativo homem-máquinas complexas,” thesis, Centro de Pesquisas Aggeu Magalhães, Fundação Oswaldo Cruz, Pernambuco:Brasil, Aug. 2010, pp. 567-612.

[18] Karnik, N.N. e Mendel, J.M. “Introduction to type-2 fuzzy logic systems”. Fuzzy Systems Proceedings, IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference. Vol. 2, 4-9 May 1998 p.915 – 920. . (1998)

Identifying Opinion Mining Elements Based on Dependency Relations and Fuzzy Logic

Abdelaziz Bouras†, Haiqing Zhang*, Aicha Sekhari*, Yacine Ouzrout*
†ictQATAR Chair, Computer Science & Engineering Department, Qatar University,
PO Box. 2713, Doha, Qatar
*DISP laboratory, University Lumière Lyon 2, France, 160 Bd de l'Université 69676 Bron Cedex
abdelaziz.bouras@qu.edu.qa

Abstract: *Opinion mining mainly involves three elements: feature and feature-of relations, opinion expressions and the related opinion attributes (e.g. Polarity), and feature-opinion relations. Although many works are emerged to achieve the aim of gaining information, the previous researches typically handle each of the three elements in isolation that cannot give the sufficient information extraction results and hence increases the complexity and the running time of information extraction. In this paper, we propose an opinion mining extraction algorithm to jointly discover the main opinion mining elements. Specifically, the algorithm automatically builds kernels to combine closest words into new terms from word level to phrase level based on dependency relations, and we ensure the accuracy of opinion expressions and polarity based on fuzzy measurements, opinion degree intensifiers, and opinion patterns. The analyzed reviews show that the proposed algorithm can effectively identify the main elements simultaneously and outperform the baseline methods.*

Keywords: opinion mining; dependency relations; fuzzy sets and logic; opinion degree intensifiers; feature-by-feature analysis

1 Introduction

The widely used Web communication on mobile and web-based technologies has dramatically changed the way individuals and communities express their opinions. More and more reviews are posted online to describe customers' opinions on various types of products. These reviews are fundamental information to support both firms and customers to make good decisions. The features and attributes of a product extracted from online customer reviews can be used in recognizing the strengths and weaknesses of the heterogeneous products for firms. While customers do not always have the ability to wisely choose among a variety of products in the market, they commonly seek product information from online reviews before purchasing a new product.

Identifying the opinions in a large-scale document of customer reviews is an opinion mining issue, which is a sub-division of information extraction that is concerned with the features, with the opinion it expresses. Two fundamental problems of mining such information are opinion features extraction and opinion words locating.

Opinion features are characteristics of the products on which opinion has been described. Two issues are generated in product feature extraction. One is that synonyms are often occurring in extraction of features. The other one is some product features are combined by several nouns. Hence, feature-of relation is used to record the synonyms of features and rebuild the noun terms to more accurately represent product features.

Opinion expressions are the opinion words that the reviewers have adopted to describe their opinions on the related features. Opinion expressions are commonly composed by an opinion pattern involving adjectives, adverbs, and verbs instead of a single opinion word. Thus, opinion-of relation extraction is adopted to keep the opinion patterns. Opinion expressions also need to express the evaluation for correct targets. The feature-opinion relation extraction is necessary to be proposed to express the opinion expressions corresponding with the related opinion features.

This paper aims to solve the information extraction issues. And the remainder of this paper is organized as follows. Section 2 reviews the main related works. Section 3 represents the mechanism of assigning the polarity and intensity of opinion expressions. Section 4 introduces the opinion mining algorithm to jointly complete opinion mining tasks. Section 5 conducts the experiments in multiple aspects and analyzes the superiorities and the deficiencies of the proposed algorithm by comparing with the baseline works. Section 6 concludes the paper with future works.

2 Related work

In this paper, we focus on jointly detecting the three principle elements in the reviews: feature and feature-of relation, opinion and opinion pattern extraction, and feature-opinion extraction. In previous works, these elements have mostly been studied in isolation. Therefore, we treat these three elements as three separate tasks and study the related works.

The existing works on feature extraction can be divided into three groups: frequent term mining, supervised sequence labeling, and unsupervised and knowledge-learning based approach. The most representative work for "frequent term mining" approach is Hu and Liu (2004), which is restricted to detecting the features that are strongly associated with a single noun and considering only adjectives collocated with the near feature words as opinion expressions. Some additional works (Zhuang et al., 2006; Qiu et al., 2011) involve manually constructed rules and semantic analysis, but these still cannot fully reduce the disadvantages of this branch. The "supervised sequence labeling" (Jakob and Gurevych, 2010; Choi and Cardie, 2010) usually needs a large amount of training data that are mainly composed by hand-labeled training sentences. All of the methods mentioned above do not have the ability to group semantically related expression aspects together. The existing works belonging to "unsupervised and know-learning based approach (topic modeling) are based on two models: PLSA (Probabilistic Latent Semantic Analysis) (Hofmann, 1999) and LDA (Latent Dirichlet Allocation) (Blei et al., 2003). According to the work(Titov and McDonald, 2008), the existing models are not suitable to be used to detect features, because they can only work well for capturing global topics, but cannot intelligently understand human judgments.

The opinion expressions consist of a set of opinion words, which are used to present the polarities of sentiments and measure the strength of the expressed opinions. Previous research can be divided into two categories: CRF (Conditional Random Field)-based approaches and parsing-based approaches. Most of the CRF-based approaches mainly focus on one direction and single word expressions. However, all of the approaches belonging to this category are token-level and cannot efficiently extract phrase-level information. Although semi-CRFs (Okanohara et al., 2006) are proposed to allow sequence labeling in phrase-level, these methods are known to be difficult to implement (Yang and Cardie, 2012). Previous works(Kobayashi et al., 2007; Joshi and Penstein-Rosé, 2009) show that adopting syntactic parsing features to identify opinion expressions and the related attributes is more helpful than the CRF-based approach. Moreover, some combination approaches (Brody and Elhadad, 2010; Kobayashi et al., 2007) are proposed by considering the impacts between some internal elements.

In conclusion, all of the approaches have their own advantages and disadvantages. Although some models obviously outperform others in each element, to the best of our knowledge, there is no solution that is simultaneous proficient in all three elements in practice**.** In the opinion mining processes, the three elements usually lie in a labyrinth of relationships and one element will encounter another element in each sentence, which makes the opinion mining results not as straightforward to obtain. To be able to gain more benefits from actual practice for firms and customers, we aim to find a compromising solution that allows the three elements to be taken into account as an integrated unit instead of seeking the best approach for one element.

3 Fuzzy weights assigning for opinion expressions

Around 6800 positive and negative English opinion words were compiled by Hu and Liu (2004). We have extended these opinion words by adding some words that can express the degree of intensity in the customer's emotion. We have collected 62 adverbs that are called ***Opinion Degree Intensifiers,*** which can be used in both a positive and negative situation to express the opinion degree or to change the orientation of the opinion. Opinion Degree Intensifiers are grouped into two types: adverbs that only change the opinion degree; and adverbs that will change the orientation of the opinion. The opinion expressions have the characteristics of uncertainty as different customers will adopt different words to express the same opinion and the same word has different opinion intensity under different circumstances. Fuzzy logic is a sophisticated approach to tackle uncertain and inaccurate issues (Zhang et al., 2014). Therefore, five fuzzy degrees are defined for the first type of words based on the intensity of the adverbs. Three fuzzy degrees are given for the second type of words, because there are fewer words that have such function and the gaps among these words are narrow.

The 6800 opinion words are updated with assigned weights that lie in [-1, 1]. The sets of opinion words are categorized into five levels based on the orientation of the word. Some words are defined as the benchmark (core), which can be used as the standard when determining the other words' polarities.

To be able to know the fuzzy weights of every reviewer, two different cases are defined based on different combinations of opinion words in the proposed patterns.

Definition 3.1.1 (weights of case 1) The opinion is the combination of the opinion degree intensifiers and 6800 opinion words that include adjectives and verbs. The weights of the opinion in case 1 are defined in four types of situations based on the words' orientation, which is shown in the following equation:

$$weights\ of\begin{bmatrix}(\text{RB/RBR/RBS})\,combination\\ with\ (\text{JJ/RB/RBR/RBS})\end{bmatrix}=$$

$$\begin{cases}1.\ \text{degree}(\text{RB/ RBR/ RBS})\oplus\text{degree(JJ/ RB/ RBR/ RBS)},\\ if\ \text{degree}(\text{RB/ RBR/ RBS})>0\ \&\ \text{degree(JJ/RB/RBR/RBS)>0}\\ \quad \text{eg: very good, extremely high}\\ elseif\ \text{degree}(\text{RB/ RBR/ RBS})<0\ \&\ \text{degree(JJ/RB/RBR/RBS)>0}\\ \text{eg: not good, not high}\\ 2.-\text{degree}(\text{RB/ RBR/ RBS})\oplus\text{degree(JJ/ RB/ RBR/ RBS)},\\ if\ \text{degree}(\text{RB/ RBR/ RBS})>0\ \&\ \text{degree(JJ/RB/RBR/RBS)<0}\\ \quad \text{eg: very bad, extremely annoyed}\\ 3.\ \text{degree}(\text{RB/ RBR/ RBS})\otimes\text{degree(JJ/ RB/ RBR/ RBS))},\\ if\ \text{degree}(\text{RB/ RBR/ RBS})<0\ \&\ \text{degree(JJ/RB/RBR/RBS)<0}\\ \quad \text{eg : not bad, not annoyed}\end{cases}\qquad(1)$$

Definition 3.1.2 (weights of case 2) Some opinion words appear together with case 1. For instance, "not a very good camera", "extremely high quality", etc. The opinion phrases of such types are calculated by Eq.2 and Eq.3.

$$weights\ of\begin{bmatrix}(not\,/\,never\,/\,...)\begin{bmatrix}combination\ with\\ (\text{RB/RBR/RBS})\\ combination\ with\\ (\text{JJ/RB/RBR/RBS})\end{bmatrix}\end{bmatrix}$$

$$=\begin{cases}1.\ \text{degree}((\text{RB/RBR/RBS})\,combination\ with\\ (\text{JJ/RB/RBR/RBS}))\oplus(-(-\text{degree}(not\,/\,never\,/\,...))^2)\\ if\ \text{degree(JJ/RB/RBR/RBS)>0}\\ eg: not\ very\ good,\ \text{not extremely high}\\ 2.\ \text{degree}((\text{RB/RBR/RBS})\,combination\ with\\ (\text{JJ/RB/RBR/RBS}))\oplus((-\text{degree}(not\,/\,never\,/\,...))^2)\\ if\ \text{degree(JJ/RB/RBR/RBS)<0}\\ eg: not\ very\ bad,\ \text{not extremely annoyed}\end{cases}\qquad(2)$$

$$weights\ of\begin{bmatrix}(very\,/\,so\,/\,...)\begin{bmatrix}combination\ with\\ (\text{RB/RBR/RBS})\\ combination\ with\\ (\text{JJ/RB/RBR/RBS})\end{bmatrix}\end{bmatrix}$$

$$=\begin{cases}1.\ \text{degree}((\text{RB/RBR/RBS})\,combination\ with\\ (\text{JJ/RB/RBR/RBS}))\oplus((\text{degree}(very\,/\,so\,/\,...))^2)\\ if\ \text{degree(JJ/RB/RBR/RBS)>0}\\ eg: \text{very}\ very\ good,\ \text{so extremely high}\\ 2.\ \text{degree}((\text{RB/RBR/RBS})\,combination\ with\\ (\text{JJ/RB/RBR/RBS}))\oplus(-(\text{degree}(very\,/\,so\,/\,...))^2)\\ if\ \text{degree(JJ/RB/RBR/RBS)<0}\\ eg: \text{very}\ very\ bad,\ \text{so extremely annoyed}\end{cases}\qquad(3)$$

Definition 3.1.3 (Weight for a review). The weight of a review is calculated based on fuzzy operation. The appearance frequency of the opinion features in the review and the related fuzzy weights of opinion words are two important elements that can determine the weight of a review.

$$\text{RW}=\frac{\sum_{i=1}^{n}\text{fuzzy scale}(opinion\ words)\otimes f(\text{ Related features})_i}{\sum_{i=1}^{n}f(\text{Related features})_i}\qquad(4)$$

$(where,\ n\ is\ the\ total\ number\ of\ features\ in\ a\ review)$

The weights of the extracted opinion expressions are defined in case 1 and case 2, and the weight for a review is defined in definition 3.1.3. Fuzzy logic is used in the calculation process to make sure the obtained weights are accurate. In order to deeply answer the necessary information of an opinion, the opinion words and the features should be accurately extracted. In the next section, the algorithms of opinion words and feature extraction will be given and the dependency structure will be employed to express the relations between opinion expressions and features.

4 Jointly execute opinion mining extraction tasks

4.1 Extraction rules defined based on dependency relations

The extraction is mainly between features and opinion words. For convenience, some symbols are defined easy reusability. The relations: between opinions and features are defined as FO↔Rel, between opinion words themselves are OO↔Rel, and between features are FF↔Rel. Four basic extraction tasks are defined to separate information extraction: (1). Extracting products' features by using opinion words (FO↔Rel); (2). Retrieving opinions by using the obtained features (OF↔Rel); (3). Extracting features by using the extracted features (FF-Rel); (4). Retrieving opinions based on the known opinion words (OO-Rel). Four categories of running rules are clarified and depicted in Table 1.

In Table 1, o (or f) represents for the obtained opinion expressions (or features). O (or F) is the set of known opinions (or features) either given or obtained. POS (O/F) means the POS information that contains the linguistic category of words, such as *noun* and *verb*.{NN, NNS, JJ, RB,VB} are POS tags. O-Dep, that represents the opinion word O, depends on the second word based on O-dep relation. F-dep means the feature word F depends on the second word through F-dep relation. MR={nsubj, mod, prep, obj, conj,dep}, 'mod' contains {amod, advmod}, 'obj' contains {pobj, dobj}. Finally, rules ($R1_i$ –$R4_i$) are formalized and employed to extract features (f) or opinion words (O).

Table 1. Rules for features and opinion expressions extraction

Rule	Input	Representation Formula	Output	Example
$R1_1$	O	$O \xrightarrow{\text{Depend (O-Dep)}} F$; where, $O \in \{O\}$, O-Dep $\in \{MR\}$, $POS(F) \in \{NN,\ NNS\}$	f=F; FO↔Rel	Canon PowerShot SX510 takes **good** *photos*. (good→amod→photos)
$R1_2$	O	$O \xrightarrow{O-Dep} H \xleftarrow{F-Dep} F$ s.t. $O \in \{O\}$, O / F-Dep $\in \{MR\}$ $POS(F) \in \{NN,\ NNS\}$	f=F FO↔Rel	The Canon PowerShot SX510 *HS* is a very **good** *value* thanks to a new sensor. (good→amod→value←nsubj←HS)
$R1_3$	O	$O \xrightarrow{O-Dep} H \xrightarrow{F-Dep} F$ s.t. $O \in \{O\}$, O / F-Dep $\in \{MR\}$, $POS(F) \in \{NN, NNS\}$	f=F FO↔Rel	It works **great** for a kindle camera. (great ←prep←for←pobj←camera)
$R2_1$	F	$O \xrightarrow{O-Dep} F$; s.t. $F \in \{F\}$, $POS(O) \in \{JJ,\ RB, VB\}$	o=O OF↔Rel	Same as $R1_1$, *photos* as the known word and *good* as the extracted word.
$R2_2$	F	$O \xrightarrow{O-Dep} H \xleftarrow{F-Dep} F$ s.t. $f \in \{F\}$, O / F-Dep $\in \{MR\}$ $POS(O) \in \{JJ,\ RB, VB\}$	o=O OF↔Rel	Same as $R1_2$, *HS* as the known word and *good* as the extracted word, also extract the middle word *value*
$R2_3$	F	$O \xrightarrow{O-Dep} H \xrightarrow{F-Dep} F$ s.t. $f \in \{F\}$, O / F-Dep $\in \{MR\}$ $POS(F) \in \{JJ,\ RB, VB\}$	o=O OF↔Rel	Same as $R1_3$, *camera* as the known word and *great* as the extracted word. (camera→pobj→for→prep→great)
$R3_1$	F	$F_{i(j)} \xrightarrow{F_{i(j)}\text{-Dep}} F_{j(i)}$ s.t. $F_{j(i)} \in \{F\}, F_{i(j)}$-Dep $\in \{conj\}$ $POS(F_{i(j)}) \in \{NN, NNS\}$	f=F FF↔Rel	It takes breathtaking **photos** and great **videos** too. (photos→conj→videos)
$R3_2$	F	$F_{i(j)} \xrightarrow{F_{i(j)}\text{-Dep}} F_{j(i)}$ s.t. $F_{j(i)} \in \{F\}, F_{i(j)}$-Dep $\in \{NN\}$ $POS(F_{i(j)}) \in \{NN, NNS\}$	f=F FF↔Rel	The image **quality** is great. quality←nn←image
$R3_3$	F	$F_i \xrightarrow{F_i-Dep} H \xleftarrow{F_j-Dep} F_j$ s.t. $F_i \in \{F\}, F_i / F_j - Dep \in \{MR\}$ $POS(F_j) \in \{NN, NNS\}$	f=F FF↔Rel	**SX500** has a smaller camera and a good sized zoom. (SX500→nsubj→has←dobj←camera←conj←zoom)
$R4_1$	O	$O_{i(j)} \xrightarrow{O_{i(j)}-Dep} O_{j(i)}$, s.t. $O_{j(i)} \in \{O\}$, $O_{i(j)} - Dep \in \{advmod,\ conj\}$, $POS(O_{i(j)}) \in \{RB\}$	o=O OO↔Rel	Canon PowerShot *SX510* takes significantly **better** indoor *photos*. (better←advmod←significantly) This camera is **light** and easy to hold. (light←conj←easy)
$R4_2$	O	$O_i \xrightarrow{O_i-Dep} H \xleftarrow{O_j-Dep} O_j$, s.t. $O_i \in \{O\}$, $O_i - Dep == O_j - Dep$ $POS(O_{i(j)}) \in \{JJ\}$	o=O OO↔Rel	If anybody wants a new light, smart, easy use camera, I highly recommend Canon PowerShot. (new→amod→camera←amod←light; new→amod→camera←amod←smart;…)

4.2 Opinion mining extraction algorithm

Table 2 shows the detailed opinion mining extraction algorithm. The initial values of the proposed algorithm are shown as: opinions dictionary O, the opinion degree intensifiers OD, and the review data RD. This algorithm adopts a single review from customers as the basic analysis unit. For each review, anytime the customer mentions a feature name, such as camera, those words are considered unique and should be excluded from the analysis. In other words, if the review talks about the “camera’s zoom” feature, and afterwards the same word “zoom” appears again in the same review; the word "zoom" will be excluded from being analyzed further. This assumption determines the stop point of the proposed algorithm. If no new feature words are found in the review, then the algorithm will stop its analysis for the current review and begin to analyze the next review.

Table 2 Algorithm 1: opinion mining extraction algorithm

Algorithm Opinion_Mining_Extraction()

Input: Opinion word dictionary O, Opinion Degree Intensifiers OD, Review Data:RD
Output: The set of features F, the set of expanded opinion words EO, the opinion polarity (or orientation) for a product: OW
BEGIN
Expanded opinion words: EO= $\varnothing$; F= $\varnothing$; ODI= $\varnothing$
For each dependency parsed review RD_k
  for each word tagged JJ,RB, and VB in RD_k
    Traversing the RD_k, and extracting the opinion words (OP_i) if they are appearing in O; i++;
    Extracting new opinion words $\{OP_j\}$ in RD_k by using the Rules $R4_1$-$R4_2$ based on extracted opinion words $\{OP_i\}$; j++;
    Inputting the obtained OP_i and OP_j into EO, and then $EO=\{OP_{[1,\ldots,i]}, OP_{[1,\ldots,j]}\}$(for short $EO=\{OP_{1-i}, OP_{1-j}\}$);
    Traversing the RD_k, and extracting the degree intensifier words (DW_d) if they are appearing in OD;
    Inputting the obtained DW_d into ODI, and then $ODI=\{DW_{1-d}\}$; d++;
  End for
  Extracting features $\{F_{fi}\}$ in RD_k by using the Rules $R1_1$-$R1_3$ based on opinion words $EO=\{OP_{1-i}, OP_{1-j}\}$; fi++;
  if (Extracted new features not in F)
    Extracting new features $\{F_{fj}\}$ using Rules $R3_1$-$R3_3$ based on the new extracted features $\{F_{fi}\}$; fj++;
    Extracting and updating new opinion words $\{OP_{1-p}\}$ using Rules $R2_1$-$R2_3$ based on extracted features $F=\{F_{fi}, F_{fj}\}$;
    Extracting new features $\{F_{fp}\}$ in RD_k by using the Rules $R1_1$-$R1_3$ based on new opinion words $EO=\{OP_{1-p}\}$; fp++;
  End if
  Setting $F=\{F_{fi}, F_{fj}, F_{fp}\}$; $EO=\{OP_{1-i}, OP_{1-j}, OP_{1-p}\}$;
  KernelFeature_OpinionSets=Build_kernel(F, EO, RD_k);
Recording appearing frequency af of EO based on related F;
  if the opinion words EO have the corresponding degree intensifier ODI
    Building triple {ODI, EO, F}
  Else if
    Building triple {null, EO, F}
  End if
Unique and update {ODI,EO,F};
Calculating the opinion polarity{OW} based on **Definition 3.1.1- 3.1.3, Triple {ODI, EO, F}, and af;**
End for
END

5 Performance comparison between baseline approaches and proposed method

The similar products S110, SX510 HS, and SX280 HS have 232, 381, and 517 reviewers respectively. The total number of sentences for each dataset is marked in Table 3. For each sentence of each review, it has five rows that include the sentence itself, POS, dependency relations, detailed dependency relations, and the sequence markers. The sequence markers are F, O, D, and N in the data sets. F denotes the features, O denotes the opinion words, DO denotes the opinion degree intensifier words, and N denotes none of them. We generate the experiments results in sentiment classification, feature and opinion extraction to make deeper analysis of algorithms performance. The classification results demonstrate that the proposed method is more effective than the other algorithms. The reason for this is that we clearly defined each opinion words’ fuzzy scale, considering some adverbs and verbs as the opinion words, and finding the modifier that could give the additional intensity information of an opinion word.

In order to test the information extraction performance, we compare the proposed method with Qiu et al. (hereinafter called Qiu2011), and **c**onditional **r**andom **fi**elds (CRF) (Jakob and Gurevych, 2010, hereinafter called Jakob2010).Qiu2011 adopted dependency parser to identify syntactic relations between opinion words and features and proposed a double propagation algorithm to do information extraction. Qiu2011 claimed that the proposed propagation algorithm outperforms CRF significantly (Lafferty et al., 2001), Popescu (Popescu and Etzioni, 2007), and Kanayama (Kanayama and Nasukawa, 2006). Jakob2010 argue that the advanced CRF-based algorithm clearly outperforms the baseline algorithms on all datasets, which improves the performance based on F-score in four single domains. Hence, we employ Qiu2011 and Jakob2010 approach as the baseline.

Table 3 gives the comparison results of different approaches. The precision of feature extraction of our method is 6.43% higher on average than Jakob2011 and Qiu2011 respectively, which means that our method can extract more effective instances among feature elements. The recall of feature extraction is also significantly improved, which is up 29.46% and 14.86% on average by comparing with Jakob2011 and Qiu2011 respectively. We observe our method outperforms better than the other methods for opinion extraction in terms of precision and recall. Meanwhile, the gain in F-score is between 0.6713 in S110 (feature extraction) and 0.8211 in SX510 HS (opinion extraction), and the achieved F-score is higher than the other methods in all datasets. The reason is that we match the reviewed data with an intensive opinion words dictionary and consider the dependency relations up to the phrase level by building kernels between closely related words of each sentence. Although the proposed method clearly outperforms the other baseline approaches, the same generation trend also exists in the individual results: Opinion extraction yields better results than feature extraction. It is because the feature words are more complex and changeable. The opinion words and the obtained feature words are used as guide words to iteratively find new features words, whereas the reviewers may adopt synonyms or analogies to describe the same feature. In general, the comprehensive analysis shows that our method is more effective and more suitable to be used in real-life cases.

Table 3 Precision, Recall, and F-score of our method, Qiu2011, and Jakob2011

Selected Product ID (High-End, Advanced Digital Canon Camera)	Directions	Methods	P	R	F
Canon PowerShot S110 No. of Reviews: 232 Sentences: 2054	Feature extraction	Our method	**0.6575**	**0.6857**	**0.6713**
		Qiu2011	0.6139	0.6043	0.6091
		Jakob2010	0.5714	0.4400	0.4972
	Opinion extraction	Our method	0.7625	**0.8222**	**0.7912**
		Qiu2011	**0.7778**	0.7125	0.7437
		Jakob2010	0.6625	0.7143	0.6874
Canon PowerShot SX510 HS No. of Reviews: 381 Sentences: 2456	Feature extraction	Our method	**0.8046**	**0.6575**	**0.7237**
		Qiu2011	0.7241	0.4118	0.5250
		Jakob2010	0.5172	0.2941	0.3750
	Opinion extraction	Our method	**0.7812**	**0.8654**	**0.8211**
		(Qiu et al., 2011)	0.7677	0.5135	0.6154
		CRF	0.6970	0.4662	0.5587
Canon PowerShot SX280 HS No. of Reviews: 517 Sentences: 4992	Feature extraction	Our method	**0.6892**	**0.7183**	**0.7034**
		Qiu2011	0.6204	0.5986	0.6093
		Jakob2010	0.4599	0.4437	0.4516
	Opinion extraction	Our method	**0.7958**	**0.7434**	**0.7687**
		Qiu2011	0.6069	0.5789	0.5926
		Jakob2010	0.4437	0.4145	0.4286

6 Conclusions and Future Work

In this paper, we proposed an opinion mining extraction algorithm that can jointly identify features, opinion expressions, and feature-opinion by using fuzzy logic to determine opinion boundaries and adopting syntactic parsing to learn and infer propagation rules between opinions and features. Our algorithm allows opinion extraction to be executed at the phrase level and can automatically detect the features that contain more than one word by building kernels through closest words. This work presents opinion intensifier sets that can aid to extract opinion degree words. In addition, we also have discovered more dependency relations between features and opinions than the previous works. Experimental evaluations show that our algorithm outperforms the baseline approaches on different extraction tasks. Recognition of important features based on the proposed algorithm will be further studied. Meanwhile, identification of proper features to improve for both product orientation and consumption quantities will be analyzed deeper in the future work as well.

7 References

[1] Bethard, S., Yu, H., Thornton, A., Hatzivassiloglou, V., Jurafsky, D., 2006. Extracting opinion propositions and opinion holders using syntactic and lexical cues, in: Computing Attitude and Affect in Text: Theory and Applications. Springer, pp. 125–141.

[2] Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022.

[3] Brody, S., Elhadad, N., 2010. An unsupervised aspect-sentiment model for online reviews, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 804–812.
[4] Choi, Y., Cardie, C., 2010. Hierarchical sequential learning for extracting opinions and their attributes, in: Proceedings of the ACL 2010 Conference Short Papers. Association for Computational Linguistics, pp.269–274.
[5] Hofmann, T., 1999. Probabilistic latent semantic indexing, in: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp.50–57.
[6] Hu, Mingqing and Bing Liu. 2004. Mining and summarizing customer reviews. In Proceedings of IGKDD'04, pages168–177.
[7] Jakob, N., Gurevych, I., 2010. Extracting Opinion Targets in a Single- and Cross-domain Setting with Conditional Random Fields, in: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP '10. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 1035–1045.
[8] Jin, W., Ho, H.H., 2009. A novel lexicalized HMM-based learning framework for web opinion mining, in: Proceedings of the 26th Annual International Conference on Machine Learning. Citeseer, pp. 465–472.
[9] Joshi, M., Penstein-Rosé, C., 2009. Generalizing dependency features for opinion mining, in: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. Association for Computational Linguistics, pp. 313–316.
[10] Kanayama, H., Nasukawa, T., 2006. Fully automatic lexicon expansion for domain-oriented sentiment analysis, in: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 355–363.
[11] Kobayashi, N., Inui, K., Matsumoto, Y., 2007. Extracting Aspect-Evaluation and Aspect-Of Relations in Opinion Mining., in: EMNLP-CoNLL.pp.1065–1074.
[12] Lafferty, J., McCallum, A., Pereira, F.C., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.
[13] Okanohara, D., Miyao, Y., Tsuruoka, Y., Tsujii, J., 2006. Improving the scalability of semi-markov conditional random fields for named entity recognition, in: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 465–472.
[14] Popescu, A.-M., Etzioni, O., 2007.Extracting product features and opinions from reviews, in: Natural Language Processing and Text Mining.Springer,pp.9–28.
[15] Qiu, G., Liu, B., Bu, J., Chen, C., 2011. Opinion word expansion and target extraction through double propagation. Comput. Linguist. 37, 9–27.
[16] Titov, I., McDonald, R.T., 2008. A Joint Model of Text and Aspect Ratings for Sentiment Summarization., in: ACL. Citeseer, pp. 308–316.
[17] Wu, Y., Zhang, Q., Huang, X., Wu, L., 2009. Phrase dependency parsing for opinion mining, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3. Association for Computational Linguistics, pp. 1533–1541.
[18] Yang, B., Cardie, C., 2012.Extracting opinion expressions with semi-markov conditional random fields, in: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics,pp.1335–1345.
[19] Zhang, H., Sekhari, A., Ouzrout, Y., Bouras, A., 2014. Optimal Inconsistency Repairing of Pairwise Comparison Matrices Using Integrated Linear Programming and Eigenvector Methods. Math. Probl. Eng. 2014, e989726. doi:10.1155/2014/989726
[20] Zhuang, L., Jing, F., Zhu, X.-Y., 2006. Movie review mining and summarization, in: Proceedings of the 15th ACM International Conference on Information and Knowledge Management. ACM, pp. 43–50.

Fuzzy Evaluation of Job Satisfaction of Hotel Employees

Rahib H.Abiyev[1], Günay Sadikoğlu[2], Esmira Abiyeva[3]
[1]Applied Artificial Intelligence Research Centre, [2]School of Tourism and Hotel Management, [3]Department of Economics, Near East University, North Cyprus, Mersin-10, Turkey

Abstract - *The evaluation of job satisfaction is based on criterions defined by the experts. Fuzzy sets are used to represent the linguistic values of the variables of these criterions. In the paper, the integration of fuzzy sets and conjoint analysis is applied for the evaluation of job satisfaction of hotel employees. Fuzzy set theory provides an excellent framework for describing imprecise meaning of preferences of the linguistic terms. The statistical data describing satisfaction levels of hotel employees are collected. Using these data and conjoint analysis the similarity measure describing the closest of opinions of the employees and experts' are determined. These similarity measures are used for the evaluation of job satisfaction of hotel employees.*

Keywords: Job satisfaction, fuzzy conjoint model, similarity degree, membership degree.

1 Introduction

Job satisfaction is a collection of positive or negative feeling of workers that they have about their job. Job satisfaction is one of basic factor that plays impotent role in the normal functioning and increasing the efficiency of business companies. For this reason, the measuring of job satisfaction becomes important for the business organizations. The measurement of job satisfaction is based on the evaluation of some criterions, by respondents (or employees). Each criterion is characterized by a set of variables which are basically depends on preferences and satisfactions (or agreements) of respondents and represented by symbolic values. Respondents using these variables express their positive or negative degree of feelings based on their work meanwhile employed in a business organization [1,2].

There are a number of methods for measuring job satisfaction. These are Likert scale (named after Rensis Likert), Yes/No questions, True/False questions, point systems, checklist, forced choice answers [2]. A Likert-type scale is the most common method which is used to measure the preferences and satisfactions (or agreements) on certain subjective questions. Nowadays Minnesota satisfaction questionnaire and Job description index is commonly used for measurement of job satisfaction.

Recently a number of research works have been done for measuring job satisfaction. Some of these studies use percentage and mean values for the analysis. [3] uses Likert scale to study and evaluate job satisfaction of library staff at the University of North California. [4] uses Likert scale for evaluation of job satisfaction in South African University. [5] uses five and seven points Likert scale to evaluate job satisfaction by analysing the measures of statistical mean, standard deviation and correlation. [6] gives analyse of job satisfaction of college teachers. The use of artificial intelligence techniques in assessment and evaluation of performances of teachers has been considered in [7,8].

The analysis of job satisfaction of hotel employees is important in service business. Hotel enterprise is as an important part of hospitality. The level of job satisfaction of hotel employees has influence to their behaviour, and then to the customer's satisfaction directly. The satisfaction of employees increases their retention, productivity, a higher level of service quality [9]. There are few studies on job satisfaction of the hotel employees. [10,11,12] examines the relationship between demographic characteristics of hotel employees and job satisfaction. These researches are basically based on statistical analysis.

The analysis of previous research works shows that the traditional approach basically uses percentage, statistical mean measures for data analysis. In general, evolution of job satisfaction is based on subjective evaluations and is very dependent on human interpretations. The measurement of preferences on certain subjective questions is based on ordered linguistic terms. Fuzzy sets can be efficiently used to describe imprecise meaning of preferences and their subjective nature [9]. Recently a set of research works have been done for evaluation of job satisfaction using fuzzy set theory. [10] used fuzzy set theory for evaluation of job satisfaction of academic staff. Fuzzy sets theory is also used for evaluation of students' perceptions on computer algebra [11], teaching quality [12], teachers' beliefs [13], credit card services [14]. In these researches, different approaches are used for solution of job satisfaction problem.

The obtaining of reliable, consistent result is very important for job satisfaction. Analysis of previous research demonstrates that there is no common approach for the analysis of job satisfaction. The job satisfactions of hotel employees are basically based on analysis of statistical data. In this paper in order to obtain reliable, consistent result, fuzzy job satisfaction of hotel employees is considered. Fuzzy conjoint analysis (CA) is applied for evaluation job satisfaction of the hotel employees of North Cyprus. The comparison of different techniques has been considered on given example.

2 Fuzzy Evaluation of Job Satisfaction

Fuzzy sets that were introduced by L.Zadeh (1965) is intended to deal with imprecision or vague information. Fuzzy set theory has ability to describe sets of concepts in human language which the one is impossible using traditional set theory. In the paper, we apply a fuzzy conjoint analysis for job satisfaction of hotel employees. The job satisfaction evaluation is based the evaluations of the variables of the criterions defined by experts. The measurements of preferences on variables of certain criterions have been used by many researchers. The answers to these variables are basically linguistic terms. These linguistic terms are represented by fuzzy sets using a preference level.

Conjoint analysis was developed to study preferred levels of individuals and relative importance of the multiple attributes of market goods [15,16]. Individuals can evaluate the multi- attributes using responses that are approximately interval in a measurement level. Conjoint analysis is concentrated on calculations of weights demonstrating the strength of stated choice. Respondents express their preferences by providing the importance of the attribute. The preferences of respondents are basically vague, uncertain and they have a subjective nature based on feeling individuals. Fuzzy set theory provides an excellent framework for describing preferences, their subjective nature. Fuzzy conjoint analysis proposed by Turksen and Willson (1994) is used for the analysis of consumer preferences in marketing [16]. Fuzzy sets are used to represent the values of the attributes evaluated by respondents. The membership degree of element y_j for the linguistic label representing item A is defined as

$$\mu_R(y_j, A) = \sum_{i=1}^{n} \left[\frac{w_i}{\sum w_i} \right] \cdot \mu_{F_i}(x_j, A) \tag{1}$$

where w_i is a score of linguistic value given by i-th respondent, $w_i \big/ \sum w_i$ is the weight that represent level of satisfaction, $\mu_{F_i}(x_j, A)$ is the membership degree for respondent j for item A according linguistic label x_j=1,2,…,n, n is a number of linguistic term, A is an item/a question.

The membership degree represents the fuzzy set of response of the respondent. This fuzzy set is compared with fuzzy set defined by expert. The comparison is done using fuzzy similarity measure which is based on Euclidian distance of two fuzzy sets. The similarity measure is calculated as follows.

$$Sim(R_i(y_j, A), F(x_j, l)) = \frac{1}{\left[1 + \sqrt{\sum_{j=1}^{n} (\mu_{R_i}(y_j, A) - \mu_F(x_j, l))^2} \right]} \;;i=1,..,M,\ j=1,...,N \tag{2}$$

where $R_i(y_j, A)$ is the fuzzy sets calculated using the responses of respondents, $F(x_j, l)$ is the standard fuzzy sets defined for linguistic label *l.* M is the number of attributes, N is the number of linguistic terms. The similarity is computed for product m for each of the n possible linguistic terms. The similarity score ranges from 0 to 1. Here N is the number of members in the linguistic variable vector. The formula demonstrates that how experts' opinions correspond to the employees' opinions.

3 Experimental Results

Job satisfaction problem is solved using statistics collected from employees of the hotels. The statistical data for job satisfaction is obtained from the questionnaires. The Minnesota satisfaction questionnaire is applied for measuring of job satisfaction. The questionnaire used in the paper is given in Table 1. The questionnaires are distributed among 248 employees of different hotels of North Cyprus. In this study, the employees of six 5-stars hotels have taken part. The employees are questioned about all possible 20 states. From these employees 166 (66.9%) male and 82 (33.1%) female are used in experiments. It is observed that most of the employees are in the age category of 26 – 35 years, (45,2%). 18 – 25 years of the age category consist of 28,6%. This is followed by the age category 36- 45 years (20.6%), where the age category 46 -55 years was 4,8%. The least amount of employees falls in the categories 56-60 years and older (0.4%). The result reflects that most of the employees fall into the age category of 26- 35 years old. Result of the study indicates that 83.9% of the employees have less than 5 years of experience, where 16.1 % of the employees have more than 5 years of experience.

The variables used for evaluation of job satisfaction are: Activity, Independence, Variety, Social Status, Supervision/ human relations, Supervision/ technical, Moral values, Security, Social Service, Authority, Ability, Policies and practices, Compensation, Advancement, Responsibility, Creativity, Work Conditions, Co-workers, Recognition, Achievement. Using these attributes the questionnaire is prepared.

In order to ascertain the job satisfaction of employees, we requested to specify their level of satisfaction or dissatisfaction using Likert's 5 point Scale. Each of the questions was evaluated by linguistic terms. In the paper, five linguistic terms with five satisfaction level were created. The scale was represented as follows: Very Satisfied (VS), Satisfied (S), Neutral (N), Dissatisfied (DS) and Very

Dissatisfied (VDS). Fuzzy set theory is applied to represent these linguistic variables. The triangular membership functions are used to represent the fuzzy sets. The membership functions of the fuzzy terms are represented by following formula.

Table1. The Minnesota Satisfaction Questionnaire

N^0	Questions	Facets
Item1	Being able to keep busy all the time	Activity
Item2	The chance to work alone on the job	Independence
Item 3	The chance to do different things from time to ti	Variety
Item 4	The chance to be somebody in the community	Social Status
Item 5	The way my boss handles his/her workers	Supervision /human relations
Item 6	The competence of my supervisor in making decisions	Supervision/ technical
Item 7	Being able to do things that don't go against my conscience	Moral values
Item 8	The way my job provides for steady employment	Security
Item 9	The chance to do things for other people	Social Service
Item 10	The chance to tell people what to do	Authority
Item 11	The chance to do something that makes use of my abilities	Ability
Item 12	The way company policies are put into practice	Policies and practices
Item 13	My pay and the amount of work I do	Compensation
Item 14	The chances for advancement on this job	Advancement
Item 15	The freedom to use my own judgment	Responsibilit y
Item 16	The chance to try my own methods of doing the job	Creativity
Item 17	The working conditions	Work Conditions
Item 18	The way my coworkers get along with each other	Co- workers
Item 19	The praise I get for doing a good job	Recognition
Item 20	The feeling of accomplishment I get from the job	Achievement

$$\mu_A(x)=\begin{cases}\dfrac{x-a_1}{a_2-a_1}r & \text{if } a_1\le x\le a_2\\ \dfrac{a_3-x}{a_3-a_2}r & \text{if } a_2\le x\le a_3\\ 0 & \text{otherwise}\end{cases} \tag{3}$$

where a_2 is centre, a_1 is left and a_3 is right sides of triangle.

The level of satisfaction of the employees for each question in terms of 5 point Likert's scales is given in Table 2. The numbers of the employees and their percentages are given in the table. The input data given in table were provided to the system input.

Table 2. Job Satisfaction Level Analysis

	VDS		DS		N		S		VS	
Ques.	Fr.	%	Fr.	%	Fr.	%	Fr.	%	Fr.	%
Item 1	31	13	31	13	44	17	106	42	36	15
Item 2	28	11	35	14	47	19	100	41	38	15
Item 3	28	11	32	13	39	16	108	43	41	17
Item 4	22	9	27	11	29	12	93	37	77	31
Item 5	21	8	19	8	33	13	106	43	69	28
Item 6	10	4	35	14	37	15	94	38	72	29
Item 7	22	9	25	10	45	18	77	31	79	32
Item 8	16	6	15	6	40	16	116	47	61	25
Item 9	15	6	22	9	42	17	103	41	66	27
Item 10	23	9	24	10	37	15	103	41	61	25
Item 11	21	8	23	9	38	15	100	41	66	27
Item 12	33	13	35	14	68	27	75	31	37	15
Item 13	43	17	42	17	61	25	71	28	31	13
Item 14	34	14	30	12	45	18	97	39	42	17
Item 15	29	12	32	13	41	17	108	43	38	15
Item 16	22	9	21	8	35	14	115	47	55	22
Item 17	20	8	27	11	46	19	109	43	46	19
Item 18	22	9	27	11	39	16	98	39	62	25
Item 19	33	13	23	9	41	17	81	33	70	28
Item 20	23	9	20	8	28	11	91	37	86	35

The basic steps of the job satisfaction system are given in Figure 1. In a first step, the attributes, preference levels, and all states are determined. Using attributes and preference levels the questionnaire for job satisfaction is crated. As mentioned above 20 attributes are used for the job satisfaction questionnaire. Preference levels are determined using linguistic terms 'very satisfied', 'satisfied', 'neutral', 'dissatisfied', 'very dissatisfied'. According to the number of linguistic terms, the universe of fuzzy set is defined as (k= 1, 2, 3, 4, 5).

The five membership functions are defined using following expressions.

very satisfied' $F_1=\left\{1/1,\ 0.6/2,\ 0.2/3,\ 0/4,\ 0/5\right\}$,

'satisfied' $F_2=\left\{0.6/1,\ 1/2,\ 0.6/3,\ 0.2/4,\ 0/5\right\}$, 'neutral' $F_3 = \left\{ {0.2}/{1}, {0.6}/{2}, {1}/{3}, {0.6}/{4}, {0.2}/{5} \right\}$,

'dissatisfied' $F_4 = \left\{ {0}/{1}, {0.2}/{2}, {0.6}/{3}, {1}/{4}, {0.6}/{5} \right\}$,

'very dissatisfied' $F_5 = \left\{ {0}/{1}, {0}/{2}, {0.2}/{3}, {0.6}/{4}, {1}/{5} \right\}$

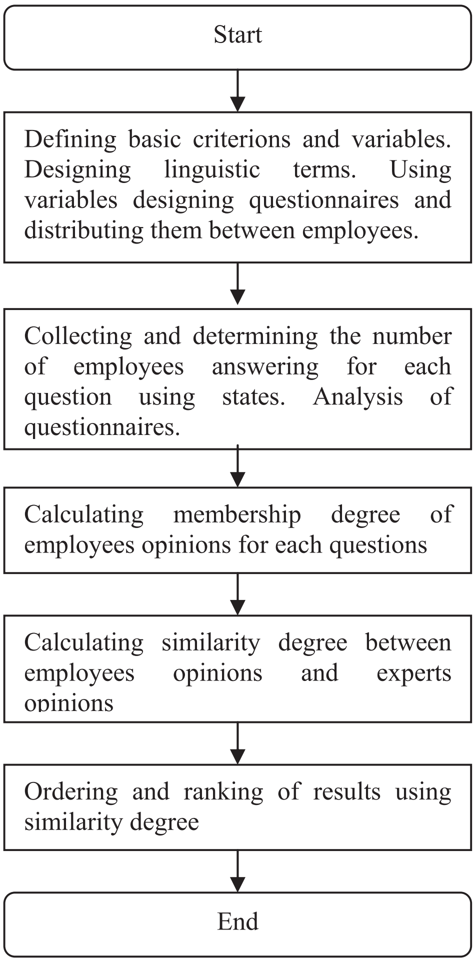


Figure 1. Basic steps of fuzzy job satisfaction

In the second step using attributes and preference levels the questionnaire for job satisfaction problem is constructed and distributed among hotel employees. Hotel employees' opinions for each attribute in the questionnaire are collected. 248 questionnaires are collected from the employees of different hotels of North Cyprus. The collected questionnaires are analyzed using employees' opinions regarding to a selected linguistic variable. Table 2 demonstrates the results of the analysis. For example, in the table for the first item, 31 employees had chosen very dissatisfied (13% of employees), 31 (13%) dissatisfied, 44(14%) neutral, 106 (42%) satisfied, 36 (15%) very satisfied. Analysis has been performed for each attribute of questionnaires. After analysis of questionnaire results, in third step, the satisfaction degree levels for each attribute is calculated using fuzzy CA model of Turksen and Willson's formula (1). The operations have bee performed by computing weight and correspondingly the membership degree (R) for respondent j for the item *A* according linguistic label. Table 3 depicts the values of membership functions. After calculating membership degrees in the fourth step the similarity degree values between employees' opinions and experts' opinions are calculated. This operation has been done by computing similarity degree between fuzzy sets *R* and *F* using formula (2). In last fifth step, the maximum amount of similarity degrees for each state is determined. Similarity degree demonstrates the maximum closeness of experts and customers' opinions to each other. Table 4 demonstrates the similarity degrees between fuzzy sets *F* and *R*.

Table 3. The Values of membership degree of fuzzy sets R.

	F_1	F_2	F_3	F_4	F_5
R1	0.1250	0.1250	0.1774	0.4274	0.1452
R2	0.1129	0.1411	0.1895	0.4032	0.1532
R3	0.1129	0.1290	0.1573	0.4355	0.1653
R4	0.0887	0.1089	0.1169	0.3750	0.3105
R5	0.0847	0.0766	0.1331	0.4274	0.2782
R6	0.0403	0.1411	0.1492	0.3790	0.2903
R7	0.0887	0.1008	0.1815	0.3105	0.3185
R8	0.0645	0.0605	0.1613	0.4677	0.2460
R9	0.0605	0.0887	0.1694	0.4153	0.2661
R10	0.0927	0.0968	0.1492	0.4153	0.2460
R11	0.0847	0.0927	0.1532	0.4032	0.2661
R12	0.1331	0.1411	0.2742	0.3024	0.1492
R13	0.1734	0.1694	0.2460	0.2863	0.1250
R14	0.1371	0.1210	0.1815	0.3911	0.1694
R15	0.1169	0.1290	0.1653	0.4355	0.1532
R16	0.0887	0.0847	0.1411	0.4637	0.2218
R17	0.0806	0.1089	0.1855	0.4395	0.1855
R18	0.0887	0.1089	0.1573	0.3952	0.2500
R19	0.1331	0.0927	0.1653	0.3266	0.2823
R20	0.0927	0.0806	0.1129	0.3669	0.3468

After calculating similarity degrees, the maximum similarity degrees for each attribute among all states are selected. Ranks are designated for the selected maximum values. Ranking is based maximum similarity degree among all states and adapted to the procedure of (Baheri et al.2011). The results of the selection of the maximum similarity degrees and results of ranking are shown in Table 5.

As can be seen in Table 5, the results are obtained with 5% neutral, 35% satisfied, and 60% very satisfied. The 20-th state- achievement with 9% very dissatisfied, 6% dissatisfied, 11% neutral, 37% satisfied, and 35% very satisfied has a best rank. The states 4 (with 8.8% very dissatisfied, 11% dissatisfied, 11.7% neutral, 37.5% satisfied and 31.5% very satisfied), 5 (with 8.5% very dissatisfied, 7.7% dissatisfied, 13.3% neutral, 42.7% satisfied and 27.8% very satisfied) and 7 (with 8.9% very dissatisfied, 10% dissatisfied, 18.1% neutral, 31% satisfied and 32% very satisfied) are ranked as the second. The worst states are 13 (with 17.3% very dissatisfied, 17% dissatisfied, 24.5% neutral, 28.6% satisfied and 12.5% very

Table 4. The Values of similarity degree between fuzzy sets F and R

	F_1	F_2	F_3	F_4	F_5
R1	0.4777	0.4729	0.5076	0.5385	0.5291
R2	0.4789	0.4764	0.5110	0.5375	0.5297
R3	0.4744	0.4697	0.5041	0.5407	0.5351
R4	0.4658	0.4566	0.4885	0.5403	0.5736
R5	0.4602	0.4524	0.4910	0.5496	0.5702
R6	0.4615	0.4607	0.4979	0.5463	0.5684
R7	0.4694	0.4617	0.4964	0.5363	0.5705
R8	0.4539	0.4500	0.4960	0.5585	0.5645
R9	0.4591	0.4559	0.4988	0.5527	0.5666
R10	0.4661	0.4595	0.4969	0.5463	0.5588
R11	0.4643	0.4580	0.4962	0.5470	0.5645
R12	0.4902	0.4887	0.5228	0.5252	0.5195
R13	0.5040	0.4975	0.5190	0.5144	0.5098
R14	0.4819	0.4744	0.5067	0.5348	0.5330
R15	0.4756	0.4712	0.5058	0.5400	0.5317
R16	0.4613	0.4555	0.4957	0.5513	0.5550
R17	0.4659	0.4647	0.5071	0.5494	0.5431
R18	0.4680	0.4624	0.4990	0.5444	0.5583
R19	0.4776	0.4648	0.4951	0.5322	0.5600
R20	0.4621	0.4505	0.4832	0.5401	0.5849

Table 5. Maximum similarity degree and ranking

R	Maximum similarity degree for each state	Ranking	Linguistic values
R1	0.5385	6	satisfied
R2	0.5375	6	satisfied
R3	0.5407	5	satisfied
R4	0.5736	2	very satisfied
R5	0.5702	2	very satisfied
R6	0.5684	3	very satisfied
R7	0.5705	2	very satisfied
R8	0.5645	3	very satisfied
R9	0.5666	3	very satisfied
R10	0.5588	4	very satisfied
R11	0.5645	3	very satisfied
R12	0.5252	7	satisfied
R13	0.5190	8	neutral
R14	0.5348	6	satisfied
R15	0.5400	5	satisfied
R16	0.5550	4	very satisfied
R17	0.5494	5	satisfied
R18	0.5583	4	very satisfied
R19	0.5600	3	very satisfied
R20	0.5849	1	very satisfied

satisfied), 12 (with 13.3% very dissatisfied, 14% dissatisfied, 27.5% neutral, 30.2% satisfied and 15% very satisfied), 14 (with 13.7% very dissatisfied, 12.1% dissatisfied, 28.1% neutral, 39% satisfied and 17% very satisfied), 2 (with 11.3% very dissatisfied, 14.1% dissatisfied, 19% neutral, 40.3% satisfied and 15.3% very satisfied) and 1 (with 12.5% very dissatisfied, 12.5% dissatisfied, 17.7% neutral, 42.7.2% satisfied and 14.5% very satisfied). Since input parameters of the model are fuzzy variables, the output of proposed model (rankings) would be fuzzy.

Comparison of the results of conjoint model developed by Turksen-Wilson method is compared with results obtained with other techniques. As an example we use Wang's (1997) [17] and Biswas's (1995) [18] approaches. Comparison of the methods is based on Spearman's rank correlation test. The results of comparison have been provided. From the comparison, it was obtained that the ranking of the satisfaction of hotel employees using Turksen-Willson's method are reliable and results can be used by the managers.

4 Conclusion

The fuzzy logic and conjoint analysis is used for evaluation of the job satisfaction of hotel employees. Respondents and experts have stated their opinions about each attribute (state), and these opinions are used for measuring of job satisfaction. Using preference levels the similarity measure between employees' opinions and experts' opinions are determined. These similarity measures are used for evaluation of job satisfaction. The experimental results have been obtained using the opinions of employees of different hotels of North Cyprus. The Turksen-Willson's method is applied to job satisfaction of the hotel employees. It has been found that 20-th achievement attribute with 9% very dissatisfied, 6% dissatisfied, 11% neutral, 37% satisfied, and 35% very satisfied has the best rank. The states 4, 5 and 7 are ranked as the second. Analysis demonstrates that Turksen and Willson's method are reliable and results obtained from conjoint analysis can be used as an alternative method for analyzing job satisfaction.

5 References

[1] Spector, P.E. Job satisfaction: Application, assessment, causes and consequences, Thousand Oaks, CA, Sage Publications, Inc., 1997.

[2] Lacy F. J. and Sheehan B.A., Job Satisfaction Among Academic Staff: An International Perspective, Higher Education, Vol. 34, 305-322, 1997.

[3] Mapesela M. and Hay D. H., The Effect of Change and Transformation on Academic Staff and Job Satisfaction: A case of a South African University, Higher Education, Vol. 52, 711-747, 2006.

[4] Murray R. A., Job satisfaction of Professional and Paraprofessional Library Staff at the University of North Carolina at Chapel Hill, University of North Carolina. 1999.

[5] Rajareegam A. and Christie Doss I., Analyzing for the Future Expectations on Job Satisfaction of Teachers in Engineering Colleges in Puducherry. IOSR Journal of Research & Method in Education (IOSR-JRME), Vol. 4(1), Ver. III, 46-51, 2014.

[6] Azadeh A., Rouzbahman M. and Saberi M., Utilization of Artificial Intelligence Approach for Assessment of Job Satisfaction , International Journal of Intelligent Information Technology Application, Vol. 2(6), 250-255, 2009.

[7] Khan A.R., Amin H.U. and Rehman Z.U., Application of Expert System with Fuzzy Logic in Teachers Performance Evaluation, International Journal of Advanced Computer Science and Applications, Vol. 2(2), 51-57, 2011.

[8] Zadeh L. A., Fuzzy Sets," Information and Control, Vol. 8, 338-353, 1965.

[9] Arasli H. and Ersoy E., Employability Skills for Hotel Sector: A Comparative Study of Management Expectations and Student Perceptions, Advances in Business-Related Scientific Research Conference 2010 (ABSRC 2010), Olbia, Sardinia, Italy, September 8-10, 2010.

[10] Liu Z. and Yang J., A Study on Job Satisfaction of Hotel Employees. IEEE International Conference on Information Management, Innovation Management and Industrial Engineering, 204-209, 2009.

[11] Chien M. Y., Tourism involvement, work engagement and job satisfaction among frontline hotel employees. Annals of Tourism Research, Vol. 42, 214–239, July, 2013.

[12] Lam T., Zhang H., and Baum T., An investigation of employees' job satisfaction: the case of hotels in Hong Kong. Tourism Management, Vol. 22, Issue 2, 157–165, April 2001.

[13] Rasmani K. A. and Shahari N.A., Job Satisfaction Evaluation using Fuzzy Approach. IEEE Third International Conference on Natural Computation (ICNC 2007). Vol.4, 544-548, 2007.

[14] Lazim M.A., Abu-Osman M. T. and Abdullah W.S.W., Fuzzy Set Conjoint Model in Describing Students' Perceptions on Computer Algebra System Learning Environment. IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March, 2011.

[15] Othman M, Ku-Mahamud K.R., Abu-Bakar A., Fuzzy evaluation method using fuzzy rule approach in multiriteria analysis. Yugoslav Journal of Operations Research. Vol. 18(1), 95-107, 2008.

[16] Lazim M.A., Abu Osman M.T., Measuring teachers' beliefs about Mathematics: A fuzzy set approach. World Academy of Science, Engineering and Technology. International Journal of Social, Management, Economics and Business Engineering Vol.3(9), 350-354, 2009.

[17] Baheri E., Dalvand M.R., Ansarinejad A., Miri-Nargesi S. and Hatami-Shirkouhi L., A fuzzy conjoint analysis approach for evaluating credit card services: A case study of Iranian bank. African Journal of Business Management Vol.5 (7), 2753-2765, 2011.

[18] Louviere J.J., Analyzing decision making: Metric conjoint analysis. Newbury Park, CA:Sage Publications, Inc., 1988.

[19] Turksen I.B. and Wilson I.A., A Fuzzy Preference Model for Consumer Choice, Fuzzy Sets and Systems, Vol. 68, 253-266, 1994.

[20] Wang W.J., New similarity measures on fuzzy sets and on elements. Fuzzy Sets Systems, Vol.85(3): 305–309, 1997.

[21] Biswas R., An application of fuzzy sets in students' evaluation". Fuzzy Sets and Systems, Vol. 74(2), 187–194, 1995.

SESSION

KNOWLEDGE DISCOVERY AND MACHINE LEARNING

Chair(s)

Dr. Raymond A. Liuzzi
Dr. Roger Dziegiel
Dr. Todd Waskiewicz
Dr. Peter M. LaMonica

Cognitive RF Systems and EM Fratricide – Part III

Gerard T. Capraro
Capraro Technologies, Inc., 401 Herkimer Road, Utica, NY 13502 USA

Abstract

The United States Department of Defense and researchers throughout the world have been addressing the overcrowding of the radio frequency (RF) spectrum. When the frequency spectrum is measured over time, technologists have shown that the spectrum is underutilized. This has led to numerous studies concerning cognitive radios, networks, and radar systems to intelligently choose frequencies, waveform parameters, antenna beam patterns, etc. to operate with conventional receivers without causing electromagnetic (EM) fratricide. In many of these studies there is an inherent assumption that the cognitive system knows when and where the fratricide occurs. In two previous papers [11, 17] we presented two approaches for determining if a cognitive solution is causing EMI in nearby receivers. In this paper a solution requiring a new cognitive user to perform an analysis and negotiate a solution before transmitting is presented, thereby establishing that the EMI is due to fratricide and not some other phenomena such as spoofing. Five major issues are addressed related to this problem area and four are discussed in detail.

1.0 Introduction

The radio frequency (RF) spectrum is crowded and more space is needed for wireless internet access, cell phone communications, and for military and civilian usage. The US Congress passed a bill to open up more spectra [1] to auction off RF frequencies belonging to the television broadcast industries. The cell phone industry is deploying microcell, picocell and femto cell technologies. However, these initiatives alone will not solve spectrum crowding. Through Department of Defense (DoD) research, we now have two distinct users defined as the primary user (PU) (i.e. those who own the license for the frequency range) and the cognitive user (CU) (i.e. those users trying to share the spectra either by using broadband signals or sampling the spectra in time and transmitting when the PU is not transmitting). Most significant projects in this area include the DARPA XG program and the Wireless Network after Next (WNaN) program. In addition to these efforts, there has been a move to apply Cognitive Radio (CR) technologies to the radar domain (Cognitive Radar efforts) and radio networks. Some of these systems sample the spectrum and transmit if no one else is transmitting at any given frequency. This approach can cause electromagnetic interference (EMI) in nearby receivers. People have recognized this problem and have addressed it in different ways [2 - 6]. Many of their solutions inherently assume they know information about the victim receiver, but do not address how this information is obtained. [11] Addressed this issue. We presented two approaches for determining if a cognitive solution is causing EMI in nearby military receivers. Since the publication of [11] DARPA released a request for proposal (RFP) [16] asking for approaches to solving the EMI problem through techniques that would cooperatively share the spectrum between military radar systems and military and civilian communication systems, e.g. mobile cell communications. An approach of how radar and communication systems can cooperatively share the spectrum and reduce EMI was presented in [17]. Since this publication DARPA has expanded upon [16] and issued a new RFP [18] where they are looking for solutions that will enable the coexistence of communications systems with existing or near-by future radar systems.

Section 2 briefly reviews cognitive radio and radar technologies. Section 3 defines the problem we are addressing. Section 4 provides a description of EM considerations that must be addressed. Section 5 provides a possible solution. Section 6 presents a summary and conclusions.

2.0 Some Cognitive Efforts

The XG (neXt Generation Communications) program is developing an architecture that will open up the spectrum for more use by first sensing and then using unused portions of the spectrum [7]. This program has developed its own XG policy language (XGPL) which uses OWL as its standard representation and will be implemented within the Policy Reasoner.

The Wireless after Next (WNaN) being performed by Raytheon BBN Technologies and funded by DARPA [8] is developing a scalable, adaptive, ad hoc network capability that will provide reliable communications to the military. The basic ingredients of their design are composed of a Dynamic Spectrum Address capability based upon the XG program. It also has 4 multiple transceivers and a disruptive tolerant networking (DTN) capability. The four transceivers provide fault tolerance and allow the system to pick the best channel for communications. The DTN capability allows the nodes to store packets temporarily during link outages. The WNaN also has content based access that allows users to query the network to find information and allow the system to store critical data at locations to minimize time and bandwidth.

Another effort related to communications, and having similar goals to the XG program, is the Cognitive Radio [9]. Its objectives are to efficiently utilize the RF spectrum and to provide reliable communications at all times. A general overview and projections of the Cognitive Radio in our society can be found in [10].

Interest in cognitive radar is growing. [12] Describes a cognitive radar that is primarily concerned with the tracking stages of a radar. A cognitive radar architecture is presented in the first textbook written on this subject [13]. The commonality of their radar designs is the feedback loop between the transmitter and receiver, use of outside sources of information, and the implementation of a learning process.

3.0 Problem Definition

In all of the above programs and many others, the CU chooses which frequency to transmit using frequency policy rules based upon location and whether someone is currently transmitting within the range of interest. There are at least five issues with this approach. The first is related to the sensing of the environment. What happens if a nearby receiver is not transmitting but is waiting to receive a signal at a specific frequency, for example a bistatic radar receiver or an electronic warfare receiver? They don't transmit, they just receive. The second issue relates to the following scenario. Let us assume that one decides to transmit broadband signals below the sensitivity levels of any nearby receivers. As the number of similar CU increases, the signals within a nearby PU receiver's passband may exceed the noise floor and interfere with the performance of the receiver [14]. The third issue occurs when a CU decides to transmit at a particular frequency because there are no signals present. The chosen frequency is based upon a linear relationship between the frequency chosen and the sensed environment. The decision policy does not take into account the effect of the nonlinearities between the chosen frequency and other nearby frequencies which can mix nonlinearly and cause receiver intermodulation or mix within the receiver's frontend and cause spurious responses. Nor does the transmitter consider its non-linear emissions and the tuned frequencies of nearby receivers. Most EMI situations are nonlinear. The fourth issue is how the receiver identifies that its performance is being degraded by EMI due to fratricide and not by other means such as jamming or spoofing. The fifth issue is to cooperatively solve the EMI fratricide issue and increase the performance of the victim receiver.

4.0 EM Considerations

To solve the EM fratricide issues discussed above some people are looking to change the beam pattern of the transmitter so that the power coupled to a victim receiver is reduced, some wish to change the transmitted signal's polarization and/or power output, and of course, there is the attenuation gained by employing orthogonal waveforms. All these solutions and others help reduce the amount of degradation caused to a friendly receiver. However, these techniques inherently are assuming that one knows that the receiver is being degraded. How would a cognitive radar, radio or a WNaN know about the receiver?

Over the past several years, DARPA and other DOD agencies have initiated numerous research projects in dynamic spectrum sharing among various military and commercial systems. Some of these approaches were discussed above. However, these spectrum sharing approaches, which relied upon spectrum sensing, did not consider the presence of recently non-radiating transceivers nearby. They do not implement cooperative techniques amongst those sharing the spectrum, nor have they taken into consideration the benefits possible through anticipating the near-term actions of other users. The DARPA RFPs [16, 18] address these issues, and requests solutions having radar and communications devices share information on EMI and how to best remedy the situation in near real time.

References [16, 18] address three real world scenarios, the interaction of radar with 1.) small cell broadband (SCB), 2) 802.11 Wi-Fi hot spots for the commercial world and, 3.) Mobile ad hoc networks (MANET) systems for the military world. Operations in an electronic countermeasures environment must also be considered. Figure 1 presents a scenario with radar having the higher priority (primary) and communications as secondary.

To delve deeper into a solution let us first address the five issues mentioned earlier. Our solution is based on distributing similar cognitive processing on all transceivers. See Figure 2. All transceivers must log where they are to a distributed database maintained by each transceiver. This would be similar to what was developed within the XG program [7] where a system policy reasoner and a system strategy reasoner contains information about the spectrum policy rules dependent upon where it is deployed and a set of ontologies so it can reason, using OWL, on how to share the spectrum. This capability is shown within the strategy creator in Figure 2. We would have to add to its knowledge base the position of all friendly equipment in the area along with all its linear and non-linear characteristics such that the reasoners can include their locations and characteristics before assigning frequencies and modulations for transmissions. This is necessary so that if a CU wishes to change its frequency it is aware of what receivers (both PU and CU) are operating nearby even though they are not transmitting and will use this information to determine if it will cause EMI to those receivers, i.e. solving the first issue.

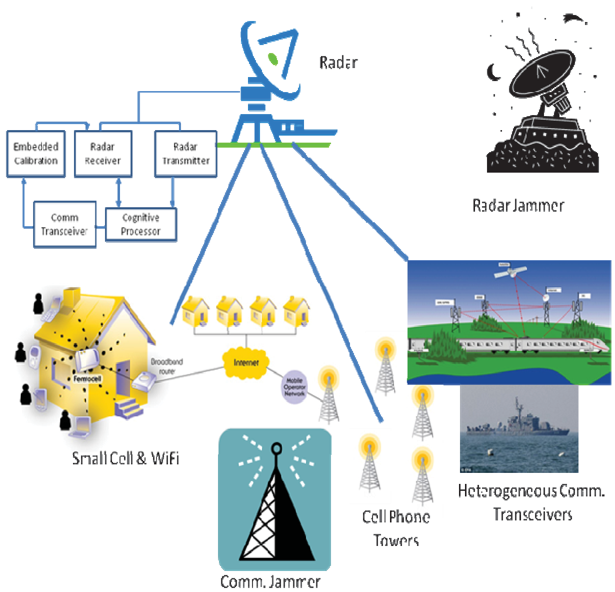


Figure 1 Cooperative Spectrum Sharing

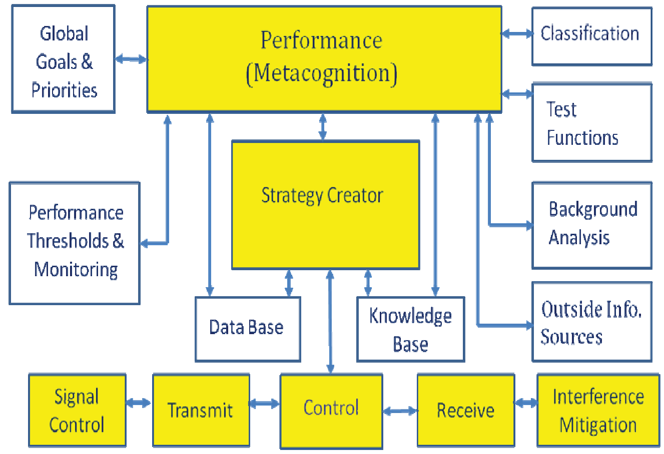


Figure 2 Cognitive Node Software Architecture

The second and third issues are related to the first. That is we must know where all transceivers are located and all their operational characteristics so that it can compute whether or not by transmitting a broad band signal if it will increase the noise floor in the receiver to a level where it will degrade the receiver's performance. This must be done whether it is a radar, comm. MANET, or device trying to communicate with a WiFi node or SCB. This can be accomplished by summing all the power of the transmitters impinging on those nearby receivers operating in the same frequency band. If the adding the CUs transmitted frequency will raise the noise floor to a detrimental level then the transmitter must find another solution. The third issue must choose its transmitted frequency such that it is not being used by a nearby receiver but also it must compute that the frequency chosen will not cause a spurious response in a nearby receiver or cause an intermodulation product that will fall into the receiver's passband. (See [19] or the latest version of Mil-Std-461.) Consider the following equation.

$$F = |mF1 - nF2|$$

where m and n are positive integers,
F1 = frequency of transmitter 1,
F2 = frequency of transmitter 2.

If transmitter 1 is transmitting F1 and transmitter 2 is a CU transmitter and decides to transmit frequency F2 then the resultant frequency F may be equal to a receiver that is operating nearby. For example let F1 = 350MHz, F2 = 375MHz, and let m = 2 and n =1 then the resultant frequency F = 325 MHz. If there is a nearby receiver operating at 325 MHz then the choice of F2 = 375MHz has the potential of causing EMI in the receiver if the power levels, distances and characteristics are such that the desired signal level is masked by the interference, e.g.

$$S/(I + N) < \text{Threshold},$$

where S is the desired signal strength at the receiver, I is the interference level at the receiver (e.g. at F due to third order intermodulation), the threshold is the ratio that must be exceeded for the receiver to have good performance, and N is the ambient noise of the environment.

The choice of F2 must be analyzed with all transmitted signals in the area (including spurious transmitter emissions) and compared to all other receivers in the area for third and higher odd order intermodulations (where the order is equal to the sum of m and n).

The equations for spurious response are similar to the above accept it relates the frequency chosen (F2) and the frequency of the local oscillator (F1) of the potential victim receiver and the receiver's intermediate frequency (F), in a heterodyne receiver. The objective of this discussion is to emphasize that in order to assign frequencies strictly based on linear frequency relationships will not suffice in eliminating EMI in nearby receivers. There are other nonlinear phenomena that can cause EMI such as receiver intermodulation, cross modulation, structural non-linearities, non-linear transmitter emissions, saturation, and desensitization.

Given that we employ all of the solutions discussed above through the architecture in Figure 2 there still is a chance of EMI. This leads to the fourth issue of how does a receiver know that it is suffering from EMI. It knows it because the receiver's performance is degraded. However, there are four ways a receiver's performance can be degraded. First a signal is sent to a receiver but it is never received. This can be due to EMI for example if the interference is so high that the S/(I + N) is below the threshold of the receiver or the desired signal level is attenuated and its level is below the receiver's threshold. In addition, a receiver may not receive the desired signal because a "man-in-the-middle" attack [20] has captured the signals from a commercial WiFi transceiver and the receiver is not aware of this occurring. Second the signal is received but it is corrupted in some manner e.g. packet errors are high and retransmission is lost and the signals cannot be reconstituted at the receiver. Or a radar detects the target but do to interference the estimate in range and/or Doppler is not correct. However, this same receiver may have a loss in performance but it may be due to jamming or spoofing by the enemy. Third a receiver may not receive a signal in a MANET system because there are misbehaving nodes in the network that do not pass on message packets or through self jamming [21 – 24]. Fourth, the desired transmitter is degraded, requires repair, or is destroyed. Each type of receiver must include a cognitive ability to address these issues within the strategy creator of Figure 2 such that it can discern weather it is suffering from EMI and whether it is caused by fratricide or some other phenomena.

As an example the radar may require a cognitive embedded calibration capability using metacognition such as the insertion of known signals (non-interfering false targets) in the receiver front-end. In doing so, the radar will correlate how well it detects these false targets against other signals that it receives from nearby emitters or real targets. This will help the radar in determining if it is suffering EMI. For a WiFi system one may have each node and access point send a heart beat signal to each other to determine that they are still in communication. For a MANET each node may send a test packet to the furthest of nodes to determine connectivity. These are some of the approaches that may be employed using metacognition as shown in Figure 2 to determine that the receivers are not suffering from fratricide or EMI from other source.

5.0 A Proposed Solution

This section will lay out the beginning of a proposed solution where each node will have a software architecture similar to that shown in Figure 2. Each node should have mission goals and be able to set priorities. We have to establish performance metrics to know how well each node is functioning. We will use databases and knowledge bases to assess and maintain a node's status, and ontologies to describe the rules of spectrum operation in different theaters. Classification is necessary for radar in defining clutter statistics and in communication nodes to help define multipath and propagation loss. To assess how well a node is performing, we implement a series of tests. Background analysis is a required capability for computing propagation losses, multipath, clutter statistics, etc. where they will use, for example, Land Use Land Cover (LULC) and Digital Elevation Map (DEM) databases. Auxiliary information is required to obtain information such as jammer locations.

To illustrate our approach see Figure 3. Each node will contain all the details of each PU and CU node that may enter into the geographical area of operation. It will contain their characteristics including non-linear emissions, bandwidth, antenna characteristics, output power, frequency bands of operation, modulations, sensitivity levels, etc. It will also contain mathematical models for computing propagation loss over various terrain, antenna gain patterns, receiver susceptibility models, non-linear models for intermodulation and spurious responses, etc.

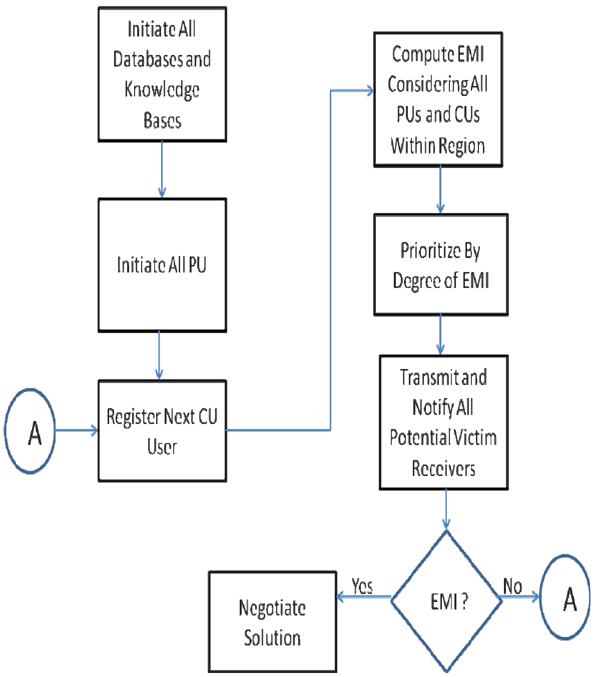


Figure 3 A Proposed Solution

The next step is to initiate all Pus, which should not incur any EMI if the frequency assignment was done properly. As one or more CU nodes enter the geographical area they will register by sending encrypted messages to all the nodes in the area to help in assessing its ability to operate. The new CU will decide what frequency, power level, beam pattern, modulation, etc. it wishes to use and then begin computing whether it will be interfered with or cause any EMI to any PUs and CUs in the area. The first process will be a frequency culling computation where it will search all receivers and transmitters to determine if the frequency of choice and its bandwidth will interfere with any nearby receivers or the nearby transmitters will interfere with the new CU. It will then compute any non-linear interactions and emitter spurious emissions against all of the receivers to determine if these phenomena will cause EMI and whether it will suffer EMI. Once completed they will be prioritized as to their possibility of causing EMI where linear interactions are more likely to cause EMI than non-linear interactions and the lower the non-linear order the higher the likely hood of EMI. Once prioritized, then the power level of each potential EMI will be computed at each receiver and the possibility of EMI, by comparing the S/(I + N) at the receiver to its threshold. These computations will take into consideration, as a minimum, power output, propagation loss, antenna gains, terrain data, noise level, receiver sensitivity, modulations, and multipath considerations. Once computed each interaction should be prioritized and computations terminated when the possibility of EMI is very low. If none of the current nodes will interfere with the CU then we proceed. If EMI will occur then the CU should note what existing emitters and non-linear phenomena is causing the EMI and proceed.

The next step is to notify all of the receivers that the, the new CU, will be transmitting a test sequence and that they request notification if any of the receivers suffers any performance degradation. If no receivers report any EMI then the CU will continue operation to identify if it suffers any degradation from the existing emitters. If there are EMI situations then the existing users and the new CU will communicate and find a solution to the EMI occurrence. This may entail placing a null in the antenna pattern, changing modulation schemes, changing polarization, time sequencing transmissions, etc.

The process of adding one new CU to the area of operation will help determine what and if an EMI is occurring due to fratricide as opposed to jamming, spoofing, or other phenomena as discussed above. In addition, the amount of computational overhead primarily burdens the new CU. This process should proceed until all CUs are added to the operational area. All communications between nodes should be encrypted and tested by periodically cycling through the CU emissions for establishing if EMI due to fratricide is occurring. This especially should be done if there is dynamic movement of the nodes or changes in their operation.

5.0 Summary/Conclusions

We have briefly reviewed cognitive radio and radar systems and the current efforts by DARPA addressing communications and radar systems which will cooperatively share the spectrum. We have defined the problem and iterated the main EM considerations that must be addressed. As part of a solution we are investigating that each node have a cognitive software architecture as shown in Figure 2. This capability will be embedded in military/commercial radar and radios e.g. MANET, cell towers, WiFi routers, and SCB routers. The major contribution of our proposed solution is to add one CU at a time so that the system can determine whether any EMI exists and that it is due to fratricide and not some other phenomena. There is a lot of work that needs to be done in developing these cognitive nodes and how they will cooperatively work together to solve EMI in near real time, learn from their discoveries and pass this knowledge onto other nodes so they can learn from each other. Further efforts will be related to cooperatively solving EMI fratricide occurrences.

Acknowledgements

The author would like to thank Mr. Edward Jones and Mr. Richard Rabe for their help in the creation of this paper.

References

[1]http://blog.broadcastengineering.com/blog-opinions/2012/02/17/congress-paves-way-for-broadcast-spectrum-auctions/

[2] P. Popovski, Y. Hiroyuki, K. Nishimori, R. Di Taranto, and R. Prasad, "Opportunistic Interference Cancellation in Cognitive Radio Systems", 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007.

[3] O. Ozdemir, E. Masazade, C. Mohan, P. Varshney, I. Kasperovich, R. Loe, A. Drozd, and S. Reichhart, "Spectrum Shaping Challenges in Dynamic Spectrum Access Networks with Transmission Hyperspace", Waveform Diversity and Design Conference, January 2012

[4] Y. Fei and Z. Wu, "An Interference Cancellation Scheme for Cognitive Radio Network", 6th International Conference on Wireless Communications Networking and Mobile Computing, Sept. 2010

[5] V. Chakravarthy, Z. Wu, A. Shaw, M. Temple, R. Kannan, and F. Garber, "A general Overlay/Underlay analytic Expression Representing Cognitive Radio Waveform", Waveform Diversity and Design Conference, June 2007

[6] E. Beadle, A. Micheals and J. Schroeder, "New Alternatives for Interference Tolerant Waveforms Hosted on a Software Programmable Multi-Mission Platform", Waveform Diversity and Design Conference, January 2012

[7] D. Elenius, G. Denker, andD. Wilkins, "XG policy Architecture", ICS-16763-TR-07-108, SRI Project No. 16763, Contract No. FA8750-05-C-0230, April 2007

[8] J. Redi and R. Ramanathan, "The DARPA WNaN Network Architecture", The 2011 Military Communications Conference, November 2011

[9] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications", IEEE Journal on Selected Areas in Communications, vol. 23, no. 2, pp. 201 – 220, February 2005

[10] Ashley, S. "Cognitive Radio", Scientific American, http://www.sciam.com/article.cfm?chanID=sa006&colID=1&articleID=000C7B72-2374-13F6-A37483414B7F0000, February 20, 2006.

[11] G. Capraro and I. Bradaric, "Cognitive RF Systems and EM Fratricide" Proceedings of the 2012 International Conference on Artificial Intelligence, Volume I, pp 110-114, July 16-19, 2012

[12] S. Haykin, "Cognitive Radar", IEEE Signal Processing Magazine, pp. 30 – 40, January 2006

[13] J. Guerci, "Cognitive Radar", Artech House, 2010

[14] I. Bradaric, G. Capraro, and D. Weiner, "Ultra Wide Band (UWB) Iterference – Assessment And Mitigation Studies", AFRL-SN-RS-TR-2006-42, February 2006, http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA446049

[15] H. Keyton, "Networks: Adapting To Uncertainty", DARPA 50Years of Bridging the Gap, http://www.darpa.mil/WorkArea/DownloadAsset.aspx?id=2570

[16]DARPA-BAA-13-24 https://www.fbo.gov/index?s=opportunity&mode=form&id=8e85f738e53747b502b4b9c3732c2e54&tab=core&_cview=1

[17] G.Capraro, "Cognitive RF Systems and EM Fratricide – Part II" Proceedings of the 2013 International Conference on Artificial Intelligence, Volume I, pp 172-176, July 22-25 2013

[18]DARPA-BAA-15-24 https://www.fbo.gov/index?s=opportunity&mode=form&id=c51cd2e9f90715c6f09f4f0a551fe64f&tab=core&_cview=1

[19] Capraro, G. T. and Perini, J. "A Graphical Approach to the Intermodulation and Spurious Response Problems," IEEE Transactions on Electromagnetic Compatibility, February 1974.

[20] http://www.veracode.com/security/man-middle-attack

[21] Suseendran, G. and Chandrasekaran, E., "Interference Reduction Technique in Mobile Adhoc Networks using Mathematical Prdiction Filters", International Journal of Computer Applications (0975 – 8887), Vol 60, No. 6, December 2012

[22] Marti S., Giuli, T. J., Lai, K. and Baker, M., "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks", Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, pp 255 – 265, August 2000.

[23] Li, W, Joshi, A., and Finin, T., "CAST: Context-Aware Security and Trust Framework for Mobile Ad-hoc Networks Using Policies", Journal of Distributed and Parallel Databases, 31(2), pp 353-376, June 2013.

[24] Peethambaran, P. and Jayasudha, J. S., "Survey of Manet Misbehaviour Detection Approaches", International Journal of Network Security & Its Applications (IJNSA), Vol 6., No.3, May 2014.

Predicting Locations of Interest with Biased Lévy Flights

A. Wheeler
3 Sigma Research, Inc., Indialantic, FL, USA

Abstract - *We apply the metaphor of crime hotspot analysis to combat mission locations/areas of interest by investigating the utility to the combat mission planner of a novel algorithm using biased Lévy flights and cellular automata for identifying hotspots in space-time. These hotspots represent locations of interest for mission planners to direct their efforts. We apply our research to the insurgency in Afghanistan and report on our initial results.*

Keywords: cellular automata, Lévy flights, locations of interest

1 Introduction

The U.S. military continues to improve the battlespace awareness of their mission planners through enhanced identification of locations/areas of interest over large areas. This capability will come through batch/stream processing of diverse spatially referenced data sources (environmental, social, cultural), automated feature extraction, data normalization, space-time pattern recognition, and visualization.

This research addresses this goal through a biologically inspired cellular automaton model of human search for attractive locations based on biased Lévy flights. We use output from this model to generate heat maps of locations of interest from the blue or red mission planner perspective. These heat maps identify locations with a high confidence of enemy activity.

The inspiration for our approach comes from work done by scientists at UCLA in crime mapping [1][2]. They model the space-time evolution of crime hotspots using cellular automata models. Their initial work modeled the criminal behavior using a biased random walk. Subsequent research modified criminal behavior to allow criminals to move significant distances before narrowing their search for a target using Lévy flights. We also take inspiration from the application of crime analysis to insurgency and counterinsurgency [3].

We believe that this approach generalizes well to situations where people must select a location over some great distance from a number of alternatives with attractiveness that can change over time. A cellular automaton model with Lévy flights seems especially appropriate for the needs of U.S. military mission planners because it can model the temporal evolution of locations/areas of interest over large gridded areas from the perspective of either the blue or red perspective. Furthermore, linking technologies for domestic law enforcement crime mapping with military mission planning creates opportunities for cross-fertilization and commercialization. When applied to the right data, these tools and GIS offer a number of space-time analysis functions that could benefit combat mission planners.

Geographic information systems (GIS) can easily convert spatially referenced data to raster (grid) format, as well as georectify and reproject raster data products to a common grid size and geographic coordinate system. The raster data format supports the representation of environmental data (e.g. elevation, air quality, land use), demographic data (e.g. population, religion, language, ethnicity), places (e.g. hospitals, schools, air fields), and space-time events (e.g. IEDs, protests). Vector data, collected either from points (e.g. space-time events, point measurements), lines (e.g. roads, rivers), or areas (e.g. census data) can also convert easily a raster format. Thus, a cellular automaton model of human search over large areas very easily lends itself to fusing data from diverse spatially referenced sources.

Our research makes a number of significant benefits and contributions. First, it addresses the need of the U.S. military to identify temporally evolving locations/areas of interest over large areas to enhance battlespace awareness to improve mission planning and execution. Second, it generalizes the idea of crime mapping to military applications involving location/area of interest identification. Finally, our approach has a theoretical foundation in a biologically inspired model of spatial search for attractive locations and contributes to the scientific literature regarding the applicability of this model to biological behavior patterns.

2 Background

We apply the metaphor of crime hotspot analysis to combat mission locations/areas of interest by investigating the utility to the combat mission planner of a novel algorithm for identifying hotspots in space-time. Others have used crime analysis in the context of insurgency and counterinsurgency [3].

The original algorithm that forms the basis of our research describes a two-dimensional cellular automaton of criminals moving through space to commit crimes. The model incorporates the attraction of the location to the criminal, prior crimes at the location, and nearby criminal activity. The model assumes criminal movement resembles Lévy flights. Lévy flights describe movement that involves occasional long distance travel along with local random travel. Lévy flights appear to describe both human and animal movement [2]. This pattern of movement could also describe combat situations where blue or red units travel a great distance then search for a local target. Similarly, red units may travel a long way to a safe haven where they feel free to move around safely. The research presented here represents our efforts to test this hypothesis.

We make the following assumptions for identifying locations of interest for adversary activity.

1. Define locations of interest as locations with high probability of adversary activity.

2. Adversaries looking for targets exhibit a type of foraging behavior.

3. Look for opportunities locally, but will travel large distances.

4. Existing activity attracts more participants, analogous to the broken windows effect for crime.

2.1 Lévy Flights and Biased Lévy Flights

Lévy flights appear as short distance movements with occasionally long distance movements. The movements represent Brownian motion with an inverse power law distribution. The probability density function of the flight distance l_j of move j has the form $P(l_j) = l^{-\mu}$ and $1 < \mu \leq 3$. Lévy flights enable optimal search for predators in environments with sparse clusters of prey beyond the range of their senses [4]. This of course generalizes to other search environment with sparse clusters of unknown goal states.

Scientists believe at least 14 species of open-ocean predatory fish exhibit foraging behavior described by Lévy flights [4]. Researches have also observed honey bees searching for their hive using a pattern of movement described by Lévy flights. The honey bees move quickly in one direction then search for landmarks indicating the direction of their hive [5].

A biased Lévy flight adjusts the direction of Brownian motion toward more attractive locations. This adaptation of the original Lévy flight model essentially extends the sensory awareness of the searcher beyond immediate neighboring sites. Scientists have also proposed Lévy flights as a way to describe criminals searching for an opportunity to commit a crime [2].

2.2 Cellular Automaton Model

A Lévy flight represents a random walk with a step size having a power law distribution. A biased Lévy flight adjusts the direction of motion toward more attractive location [2]. We now present the equations that describe the formation of hotspots [1][2].

We begin by defining the attraction $A_k(t)$ of location k as

$$A_k(t) = A_0 + B_k(t) \tag{1}$$

Self-excitation term $B_k(t)$ represents the attraction location k has given previous success at the location.

$$B_k(t+\delta t) = \left[(1-\eta) B_k(t) + \frac{\eta}{z} \sum_{s'} \right. \tag{2}$$

where the parameter δt indicates the time interval, $\hat{\eta}$ the strength of the self-excitation, ω the decay rate of the dynamic attraction, θ a proportionality constant, z the number of neighbor locations, and s a neighbor of location k. The variable $N_k(t)$ indicates the mean number of agents at location k.

We incorporate Lévy flights into the model through the relative weight W of an agent moving from i to k. The parameter μ represents the power law exponent for the Lévy flight and l the grid spacing.

$$W_{i \to k} = \frac{A_k}{l^{\mu} |i-k|^{\mu}} \tag{3}$$

We use W to compute the transition probability q of an agent moving from i to k by

$$q_{i \to k} = \frac{W_{i \to k}}{\sum_{j \neq i} W_{i \to j}} \tag{4}$$

The average number of agents at site k $N_k(t)$, with new agents recruited with rate Γ becomes

$$N_k(t+\delta t) = \sum_{i \neq k} N_i(t) \big(1 - A_i(t \tag{5}$$

Finally, we assume the probability $p_k(t)$ of an event at site k occurs according to the standard Poisson process

$$p_k(t) = 1 - e^{-A_k(t)\delta t} \tag{6}$$

3 Data and Methods

This section describes our data and relevant details of our implementation of our cellular automaton model of biased Lévy flight for battlespace awareness. We use GRASS GIS [6] to create and manage the raster maps and R [7] to implement the cellular automaton model, run the simulations, and perform the analysis. We use Quantum GIS [8] to create the maps.

3.1 Data

Our first data set consists of total coalition casualties for the time period 2001 to 2011 [9]. We use this data to measure the performance of our model. Figure 1 shows vector data of total casualties of coalition forces in Afghanistan by province from 2001-2011.

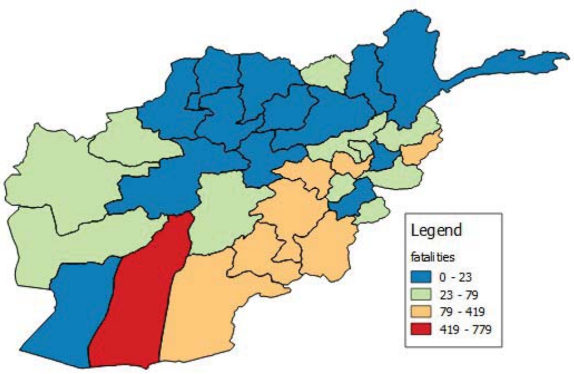


Figure 1. Total coalition casualties, 2001-2011 [9].

Equation 1 of our model specifies an initial attraction A_0 for each cell in the cellular automaton. Large population centers would seem to have more high value targets for insurgents and therefore locations near these centers would have more attraction. Figure 2 below shows the largest cities in Afghanistan.

Figure 2. Largest cities in Afghanistan.

The largest cities represent parts of the region most influenced by the outside world and represent centers of government control, even if it does not always extend far from the city. Thus, smaller settlements away from the largest cities may serve as attractive destinations for attackers because they can serve as local bases of operation. Figure 3 shows vector point data of settlements in Afghanistan.

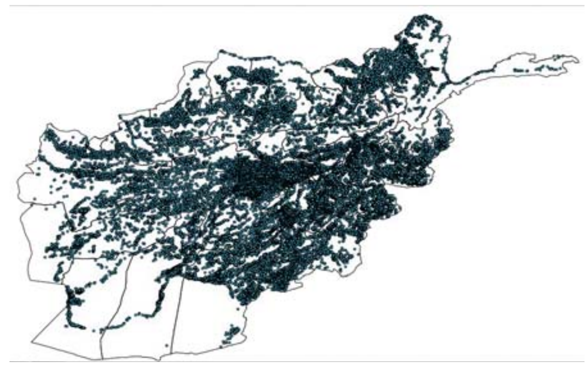


Figure 3. Settlements in Afghanistan [10].

Activities of the Taliban insurgency in Afghanistan before 2011 appear to have focus in part on areas suitable for poppy cultivation. The Taliban used heroin sales to finance their other activities. Land suitable for cultivation of poppies has a strategic importance to the insurgency so it makes sense for them to prefer targets in these areas because they serve a dual purpose of attacking US allied forces and gaining land for funding more attacks. Insurgence would target areas where they could grow poppies and not necessarily where farmers currently grow poppies. We account for this by using soil data and identifying areas having sandy loam soil, which best suits poppy cultivation (Figure 4) [11].

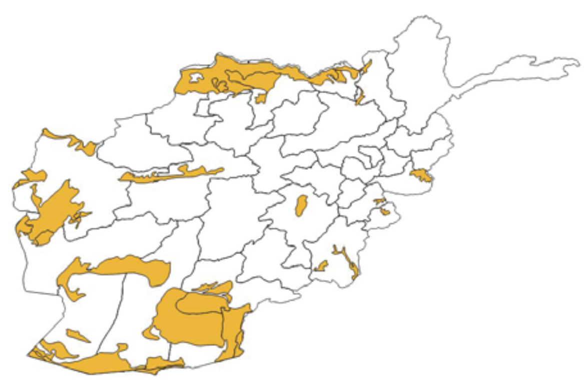


Figure 4. Soils suitable for poppy cultivation [12].

3.2 Methods

The cellular automaton evolves according the following algorithm:

1. Use GIS map algebra to create a raster grid of initial attraction $A_k(0)$ in each cell k from select raster data layers.
2. Initialize self-excitation $B_k(0)$ with estimates from past history. Assume zero.
3. Initialize average number of agents $N_k(0)$ by estimates from past history. See below.
4. Evolve the system (1)-(5) for some time period or to a steady state.

Lastly, we create a heat map of the probability values $p_k(t)$ to show locations/areas of interest.

We create a rasterized density map of population (Figure 5) from the largest cities (Figure 2).

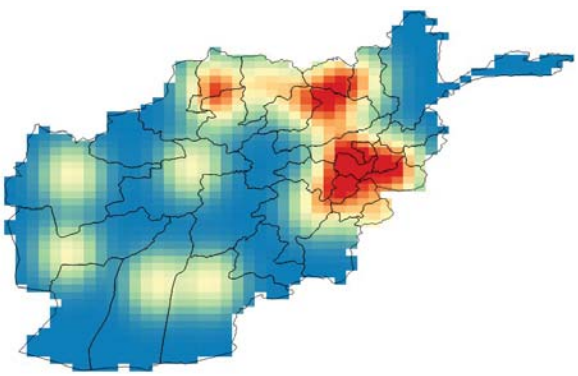


Figure 5. Population density from the largest cities in Afghanistan.

Similarly, we create a rasterized density map of settlements (Figure 6) from the distribution of settlements (Figure 3).

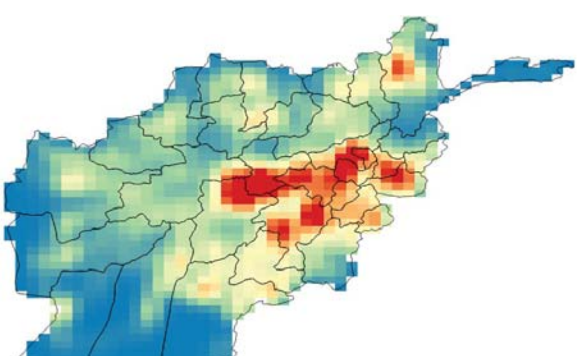


Figure 6. Settlement density.

In addition to the raster data in Figures 5-6, we also use the soil data (Figure 4) to estimate the initial attractiveness A_0 . We will compute the attractiveness using map algebra in GRASS GIS to create single values for each raster cell from the three attraction sources.

Equations 2 and 5 require an estimate for the initial number of insurgents at each location. Here we make the assumption that insurgents recruited from less educated, less informed, and conservative populations found in settlements farther from large, urban centers. Furthermore, coalition forces had a more difficult time securing these areas because of large area and small, dispersed settlements [13]. Again, we apply map algebra in GRASS GIS to the population density and settlement density maps to estimate the initial density of insurgents.

4 Analysis

We hypothesize that biased Lévy flights represent a realistic model of anti-collation activity in regions like Afghanistan. Such regions exhibit sparse clusters of meaningful targets. Adversaries must travel long distance to these clusters, then search in a smaller local area for a viable target. A successful attack by one adversary will likely attract others to lend support.

Do biased Lévy flights offer a realistic predictive model for hotspots of insurgent or criminal activity? Since this model describes optimal motion for individual searchers, it seems we can quickly answer in the negative. Biased Lévy flights guide individuals to attractive locations. Knowing the initial attractiveness of each location means we already know the final distribution of activity.

However, this assumption fails to consider the initial distribution of searchers relative to these attractive locations and the distance decay governing how far they will travel to make an attack. Many might feel reluctant to travel too far from home. Nearer to home means familiarity, support, and refuge. Thus, attractive sites need not experience the largest incidences. As attacks increase in these nearer sites, the broken window effect may result in these areas becoming more attractive than they would otherwise. The initially most attractive locations for attacks may not experience the greater number naively expected because the attackers found opportunities nearer to themselves and their success drew others to them.

We perform three types of simulations and compare their performances in predicting total coalition casualties. The first simulation uses the attraction data computed from population density, settlement density, and soil type. This allows us to test whether we need a model of search at all to predict location of interest over large areas. In the second simulation, we test whether Brownian motion, involving only local search, can explain the observed distribution of coalition casualties. We test the performance of biased Lévy flights in our third simulation.

We will analyze the results of these three simulations in R to determine there ability to accurately predict total coalition casualties. Models from the three simulations do not need to accurately predict the actual casualty counts, but only the relative values of the provinces.

5 Conclusions

We will continue our investigation into the suitability of a cellular automata model of biased Lévy flights by performing simulations under various scenarios and analyzing the results compared to historical data of insurgency activity in Afghanistan. We also intend to acquire more accurate and varied data with higher spatial resolution, then calibrating our cellular automaton model for all years of data to measure predictiveness on real-world data.

We anticipate a renewed interest in algorithms like the one presented here for identifying locations of interest for adversaries search large areas for locations to attack. We expect IS forces in Syria and Iraq will eventually fragment, leaving the hardcore members to conduct more guerrilla-style attacks throughout the region. Modeling their movement as biased Lévy flights seems like a promising way to anticipate where they will strike next. The insight provided by our cellular automaton gives mission planners another piece of valuable evidence to use it deciding where to position allied assets and conduct operations.

6 References

[1] M.B. Short, M.R. D'Orsogna, V.B. Pasour, G.E. Tita, P.J. Brantingham, A.L. Bertozzi, and L.B. Chayes (2008). A Statistical Model of Criminal Behavior. Mathematical Models and Methods in Applied Sciences, 18:1249-1267. http://www.math.ucla.edu/~bertozzi/papers/M3AS-final.pdf

[2] S. Chaturapruek, J. Breslau, D. Yazdi, T. Kolokonikov, and S.G. McCalla (2013). Crime Modeling with Lévy Flights. SIAM J. Appl. Math, 73(4):1703-1720. http://www.mathstat.dal.ca/~tkolokol/papers/levycrime.pdf

[3] Mark V. Lonsdale (2008). Criminal Activity in an Insurgent Environment and Counterinsurgency. http://www.drmcc.org/IMG/pdf/COIN_MCC_Lonsdale_18_AUG_2008.pdf

[4] Nicolas E. Humphries, Nuno Queiroz, Jennifer R. M. Dyer, Nicolas G. Pade,Michael K. Musyl, Kurt M. Schaefer, Daniel W. Fuller, Juerg M. Brunnschweiler, Thomas K. Doyle, Jonathan D. R. Houghton, Graeme C. Hays, Catherine S. Jones, Leslie R. Noble, Victoria J. Wearmouth, Emily J. Southall, and David W. Sims (2010). Environmental context explains Lévy and Brownian movement patterns of marine predators. Nature, 465(7301):1066-1069. http://www.nature.com/nature/journal/v465/n7301/pdf/nature09116.pdf

[5] Andrew M. Reynolds, Alan D. Smith, Randolf Menzel, Uwe Greggers, Donald R. Reynolds, and Joseph R. Riley (2007). Displaced Honey Bees Perform Optimal Scale-Free Search Flights. Ecology, 88(8):1955-1961. ftp://mail.honeybee.neurobiologie.fu-berlin.de/paper/Reynolds_etal_2007_Ecology.pdf

[6] GRASS GIS. OSGeo Project. http://grass.osgeo.org/

[7] The R Project for Statistical Computing. http://www.r-project.org/

[8] QGIS: A Free and Open Source Geographic Information System. http://qgis.com/

[9] Operation Iraqi Freedom and Operation Enduring Freedom Casualties. http://icasualties.org/

[10] Free GIS Shapefiles, Software, Resources and Geography Maps. http://www.mapcruzin.com/free-afghanistan-arcgis-maps-shapefiles.htm

[11] Jim Hogshire (2004). Opium - Poppy Cultivation, Morphine and Heroin Manufacture. https://www.erowid.org/archive/rhodium/chemistry/opium.html

[12] (2005) Soil Regions Map of Afghanistan. Natural Resources Conservation Service Soils. United States Department of Agriculture. http://www.nrcs.usda.gov/wps/portal/nrcs/detail/soils/use/?cid=nrcs142p2_054000

[13] Scott McGirr and Lucas Keenan (2011). Environmental Influence on Insurgent Activity in Afghanistan. MSS Proceedings, NSSDF. http://nsarchive.gwu.edu/NSAEBB/NSAEBB370/docs/Document%205.pdf

Potentially Explaining Literature Based Discoveries in Non-Medical Domains

M. Heidi McClure
Intelligent Software Solutions
Colorado Springs, CO 80919, USA
heidi.mcclure@issinc.com

Abstract

Literature base discovery has been used for many years as a way to identify possible discoveries of related concepts where the concepts have never been mentioned previously in the same document. Linking concepts and the nature of the relationship between the hidden discovery and the linking concepts may be gleaned from the documents using existing relationship extraction techniques. Little has been done to automatically explain the relationship between the hidden knowledge - the literature based discovery candidate pair. This paper presents our work in automatically identifying the nature of the relationship of the hidden discovery using supervised machine learning techniques. It starts with the research that has been completed on the medical domain and proposes work that may be done in the intelligence domain.

Keywords: literature based discovery, semmeddb, relationship extraction, classification

1 Introduction

Literature based discovery (LBD) is the discovery of hidden knowledge in large sets of documents where the discovery is never explicitly mentioned in any single document. The discoveries are often described as being between two concepts, A and C. Sometimes a term, B, is found in the documents containing A and C and this B term is called a linking term. There are statistical approaches to LBD where there may not be a linking B term in the corpus - instead, A and C are discovered by semantic relatedness of the documents using, for example, latent semantic analysis (LSA) techniques (section 2.2). Once candidate discoveries are found, experiments may be performed to prove or disprove the hypotheses. These experiments may be costly and very time consuming. To assist in deciding what discoveries are worth validating, having a system that could automatically suggest candidate relationships would be of great assistance to researchers.

We present in this paper our novel approach that automatically explains the relationships between concepts that have been identified as being related using LBD techniques. Existing approaches in LBD only assert that A and C are connected - not how they are connected despite this likely being of interest to those using LBD systems. Our approach is currently being evaluated against the medical domain, but the same approach may be used in other domain where LBD presents related A and C concepts that then need to be explained. In this paper, we propose application to the domain that includes people, places, organizations and events - we will call these named entities in this paper.

The rest of this paper provides background in LBD and topics related to this work (section 2), discussion of our approach (section 3), discussion of how we evaluate our findings (section 4), and conclusions and suggestions of future work (section 5).

2 Background

This section will provide background on what LBD is, what latent semantic analysis is, an overview relationship extraction, and a summary of evaluation techniques currently used with LBD systems.

2.1 Literature Based Discovery

As noted in the introduction, LBD attempts to discover hidden or latent knowledge in texts where the discovery is a relationship between some A and C concept where the discovery is never mentioned in any single document. In Ganiz et al. (2005) and Kostoff et al. (2007) they conclude that an LBD system must follow the following four basic constraints:

1. Arguments, concepts or terms are identified in published literature where they are often pulled from scientific articles.

2. Connections are made between pairs of arguments, for example, when discussed in LBD these would be a pair (A, B) or pair (B, C)

3. Comparisons are made among connected pairs to find those sharing an argument, B, but where the A and C are not mentioned together.

4. If no current literature mentions the A and C pairs that share some set of common B terms, this is a candidate discovery of new knowledge - an LBD A-C pair.

Original work in LBD by Swanson (1986) consisted of manually reviewing medical journals for concepts like Raynaud's Disease and for other concepts associated with Raynaud's like blood aggregation. The other concepts are the B linking terms. The corpus is then reviewed to find other documents that mention the B linking terms but not Raynaud's. Swanson found two very real discoveries using LBD - Raynaud's Disease and Fish Oil noted in Swanson (1986) and Migraines and Magnesium identified in Swanson (1988). An LBD approach is open if only starting A terms are known and closed if starting A and ending C terms are both known.

Systems that perform LBD assert connections between A, B and C terms but they do not try to explain the relationships. They are identifying relatedness by co-occurrence of terms. The work described in this paper tries to automate identification of how A and C are related.

2.2 Latent Semantic Analysis

Latent Semantic Analysis (LSA) is the process of using statistical approaches to create indexes which will allow the retrieval of similar concepts (Deerwester et al., 1990). LSA does not require, necessarily, a vocabulary, but, instead, finds similar documents based on latent semantic indexing (LSI) or enhancements to LSI like Random Indexing (Cohen et al., 2010). LSA assumes that if terms or concepts are found in similar sets of text (not always the same text, but similar) then these terms or concepts may be related concepts, may represent similar concepts, or may be the same concept. LSA uses word counts in documents to build term by document matrices. LSA and Latent Semantic Indexing (LSI) in particular may be used as a way to discover LBD pairs. LSI is described in Gordon and Dumais (1998). Widdows and Cohen (2010; Widdows and Ferraro (2008) have developed a software package called Semantic Vectors (SV) for performing LSI and it uses the performance enhancements of RI and RRI.

2.3 Relationship Extraction

The relationships being analyzed in our work focus on how A and B, B and C and, ultimately, A and C relate to each other. In Li et al. (2008), they group relationship extraction approaches into three types - co-occurrence analysis, rule-based approaches and statistical learning. Co-occurrence and rule-based approaches are an extension of named entity recognition approaches that identifying concepts in text. Statistical learning is where relationship extraction may be treated as a classification problem. Relationships may also be identified using supervised and semi-supervised techniques according to Jurafsky and Martin (2009).

Another type of relationship extraction is from the Open Information Extraction research (Open IE) (Etzioni et al., 2011). This form of relationship extraction is a semi-supervised approach - models are trained on known relationship models and then system tries to find hidden relationships. Culotta et al. (2006) present a different approach to relationship extraction that uses natural language processing (NLP). They build linear conditional random field (CRF) models, and they show how their approach can look at familial relationships and make conclusions. Their example focuses on finding in texts that George H. W. Bush has a son, George W. Bush; that George H. W. has a sibling, Nancy Ellis Bush; and that she has a son, John Prescott Ellis. From this extracted information and after applying the CRF models, they infer the relationship cousin between George W. and John.

2.4 Evaluation Techniques

Most current LBD systems are evaluated in one of four general ways (Yetisgen-Yildiz and Pratt, 2009; Yetisgen-Yildiz and Pratt, 2006; Bruza and Weeber, 2008; Ganiz et al., 2005). These authors focused on the medical domain but their evaluation techniques may be generalized for any domain. Techniques are:

1. Using experts: Experts in the corpus' domain validate or discount the LBD discovery - this is based on their current knowledge.

2. Publishing results: Put discoveries in front of experts and they validate or discount the discoveries using further study or experimentation.

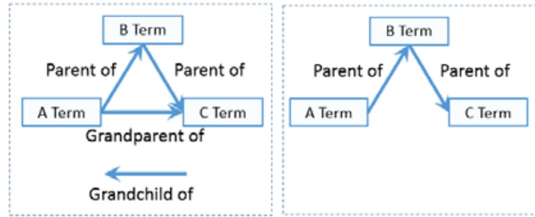
3. Replicating previously discoveries: For example, in the medical domain, considering Swanson's discoveries as gold standards.

4. Time slicing corpus: Perform LBD on a corpus from an older time range and then validate against newer documents.

In (Yetisgen-Yildiz and Pratt, 2009), the authors describe the time slice method. They propose that if LBD is run against data from a specific date and earlier, discoveries may be found which were never mentioned in the same set of data. Then, to validate if the discovery is a real one, examine data after the specific date and see if the discovery is mentioned in more recent documents. If the discovery is not mentioned prior to the specific date but is mentioned in more recent documents, then LBD successfully discovered something not previously known. The work described in this paper used the time slice method for evaluating results in the medical domain.

3 Approach

This section will describe the approach used by our system to identify LBD candidates and then to automatically explain the relationships between the A, B and C concepts.



(a) (b)

Figure 1: (a) shows fully qualified triangle with two directional A-C relationships; (b) shows a partial triangle where the A-C side is the LBD unknown relationship

Our approach uses classification to explain the relationship between A and C concepts. As a reminder, A and C are LBD candidate pairs. B linking terms are identified where A and B are mentioned in one set of documents and B and C are mentioned in another and these sets are disjoint. Partial LBD triangles are identified where the nature of the A-B and B-C relationships are known but the nature of the LBD A-C side of the triangle is not yet known. Fully qualified A-B-C triangles exist outside of LBD where nature of the relationship is known between all three sides: A-B, B-C and A-C. See figure 1.

Having a set of fully qualified triangles provides data that is used to train classifiers. Choices are made on what features to include to provide the best classification results. For example, in the medical domain, the concepts, the semantic types of the concepts, the relationships between concepts and the direction of the relationship may be the classification features chosen. Testing is performed using partially qualified triangles - in LBD, the relationship between A-C side of the triangle along with the direction are the unknown features that we want the classifier to identify for us.

The remainder of this section will discuss the data set used in our tests to date along with an interesting data set that includes various named entities. Then we will discuss the pre-processing of the data that is necessary and the pieces of the system that explain and evaluate the LBD relationships.

3.1 Data Sets

Citations and abstracts of medical journals are available from the US National Library of Medicine (NLM) National Institutes of Health and their system called PubMed. In particular, the MEDLINE corpus provides access to over 24 million abstracts as of 2012 (MEDLINE, 2012). Biomedical text mining or BioNLP has been a large place of research and often includes studies involving the MEDLINE corpus.

Using the MEDLINE corpus, Thomas Rindflesch and Marcelo Fiszman have developed a system called SemRep that extracts relationship information for concepts identified in the Unified Medical Language System (UMLS) (Rindflesch and Fiszman, 2003). They generalize the relationships into a small set of types of relationships. For example, TREATS and CAUSES are the general relationships but may be derived from more specific and technical phrases. A database called the Semantic MEDLINE database or SemMedDB (Kilicoglu et al., 2012) has been developed based on information found in SemRep.

When studying the MEDLINE corpus, another program provided by NLM called MetaMap may be used to identify synonyms in the data (MEDLINE, 2012; Preiss, 2014). This is important so that concepts presented with various synonyms are treated as the same concept. An example of this is the various phrases used to represent "Raynaud's Phenomenon" which include "Raynaud's Disease", Raynaud's Syndrome, Raynaud's and Raynaud's Phenomenon. MetaMap assigns a single concept unique identifier (CUI) for all of these representations of Raynaud's Disease".

For future expansion of our system into named entities, we would need a corpus of articles like that which may be obtained from the open source center [1]. At this time, the authors have a set of older articles dating from March 2012 thru February, 2015 that contains over 7700 articles. These articles focus on Mexico and narcotics. After extracting entities, LBD techniques applied on this domain could produce interesting hidden knowledge that could be validated against today's known relationships in this domain.

3.2 Data Preparation

Abstracts from MEDLINE corpus published in year 2002 with publication dates between 1980-1984 were used for the LBD candidate and B linking term discovery part of this work. As noted above in section 3.3, chemicals and main headings lists provided the concepts to study. In order to avoid problems with synonyms, MEDLINE's MetaMap program was used to pre-process all documents. MetaMap was used to identify synonyms of concepts being studied (see section 3.1). Once concepts or their synonyms were identified in the documents, they were replaced with the CUI so that there would be no question as to whether or not the concept is in the document, or not. Since the LBD is being done using semantic vectors which studies co-occurrence and does not look at the semantic meaning of the sentences being studied, replacing phrases like "Raynaud's phenomenon" or "Raynaud's disease" with the CUI, "C0034734", would not affect the task of identifying LBD candidates.

3.3 Details of Approach

In the this subsection, we describe in detail how we use classification to explain LBD relationships in the medical domain. Figure 2 provides an overview of our approach. We first limit our study to a set of diseases of interest (Parkinson's Disease or PD, Raynaud's Disease or RD, Cataracts and Multiple Sclerosis or MS). We used 9317 chemicals and 8103 main headings that appeared in more than 100 documents in the MEDLINE corpus - that is, they were occurred frequently enough that statistical approaches of LSA would present meaningful results. Then, in order to have an open LBD system but one that is smaller than considering all concepts found in MEDLINE, we randomly choose a subset of 500 concepts to use along with the studied set of 4 diseases. Once we have the set of concepts to study, these steps are performed:

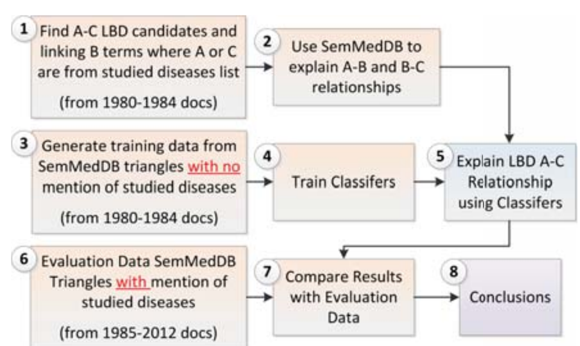


Figure 2: Summary of Approach

[1]http://www.opensource.gov

1. We perform LBD using the Semantic Vectors java package which performs an LSA-based analysis on the documents, and we study only documents from 1980-1984. From the LSA analysis, we separate related concepts pairs into those never mentioned in same documents (LBD candidates) and pairs where the concept pairs are mentioned together (Linking B terms).

2. Of the LBD candidate pairs, we use SemMedDB to explain any A-B and B-C relationship discovered.

3. We also use SemMedDB to identify complete triangles that do not mention the diseases being studied. This is the training data for classifiers. For training data, we only study documents from 1980-1984.

4. We train classifiers using fully qualified triangles found in previous step - no mention of studied diseases and only documents from 1980-1984.

5. We apply classifiers to our candidate LBD pairs and linking B terms to suggest explanations of A-C LBD pairs.

6. From SemMedDB, we generate evaluation data by finding sets of fully qualified triangles that do mention the diseases being studied. This data is from documents that are from 1985-2012 (newer).

7. We compare the explained LBD discoveries with our evaluation data to see if there are any matches.

8. In the last step, we study the explained LBD relationships comparing them with evaluation data to determine the viability of our approach and to draw conclusions.

3.4 Limited Open Discovery

The LBD in this work uses open discovery approach - the focus is on the four diseases noted above in section 3.3 (RD, PD, MS and Cataracts). However, our open approach is a limited approach in that the candidate B and C terms are from a subset of all possible MEDLINE concepts - these terms were randomly selected from a set of terms that contained 900 or more connections in SemMedDB. This trimming was performed to ensure that there would be enough documents mentioning the terms and the random selection for using only 500 of these terms further reduced the number of terms being analyzed.

3.5 Relationship Classifiers

As noted earlier, classifiers are trained using fully qualified triangles found in SemMedDB that link a set of A, B and C concepts with a relationship. To reduce the number of terms used and to try to keep the discoveries on track with identifying new treatments or causes of diseases, only some of the semantic types were allowed in the discovery of triangles. This is related to the need to filter what is studied as described in (Preiss, 2014). A total of 3174 fully qualified training triangles were identified in the SemMedDB and were used as training data.

Once training and test data is available, Weka is used as a platform on which to perform machine learning in the form of classification (Hall et al., 2009). An example of features used by the classifiers is the concepts Raynaud's disease and blood viscosity, the respective semantic types of disease or syndrome and of physiological function, the relationship of causes and the direction of the relationship which is B to A.

The training data is used to train classifiers - currently using J48, Random Forest and Simple Logistic classifiers. The cross validation where some of the the training data is used to train the classifier and the rest of the training data is used to validate the results of the classifier model generated by the first set of data. After the classifier is trained, it is applied to the test data to try to identify the relationship of the A to C side of the triangle (the LBD pair). The novel piece of this work is to use fully qualified triangles to train the classifiers and then use the classifiers to suggest explanations of the LBD candidate pair.

4 Evaluation and Results

There are two parts to evaluation of our results - one is to actually study the current literatures to see if the discovery has been confirmed in the newer documents (time slicing) and the other is to examine the suggested probabilities provided by the Weka clas-

LBD candidate discoveries with linking B terms							J48		Simple Logistic		Random Forest	
A	A-B	A-B dir	B	B-C	B-C dir	C	A-C	Probability		Probability		Probability
PD	TREATS	<-	sinemet	INTERACTS_WITH	->	madopar	ca_TREATS	100.0%	ca_COEXISTS_WITH	52.7%	ca_TREATS	40.1%
											ca_CAUSES	32.2%
MS	COEXISTS_WITH	<-	uveitis	COEXISTS_WITH	->	cataract	ca_COEXISTS_WITH	43.7%	ca_COEXISTS_WITH	49.0%	ca_COEXISTS_WITH	35.2%
MS	COEXISTS_WITH	<-	uveitis	COEXISTS_WITH	<-	cataract	ca_COEXISTS_WITH	43.7%	ca_COEXISTS_WITH	50.4%	ca_COEXISTS_WITH	32.5%
MS	COEXISTS_WITH	<-	rubella	COEXISTS_WITH	->	cataract	ca_COEXISTS_WITH	43.7%	ca_COEXISTS_WITH	49.0%	ca_COEXISTS_WITH	35.2%
salsolinol	CAUSES	->	parkinsonism	COEXISTS_WITH	<-	PD	ca_INTERACTS_WITH	34.8%	ca_INTERACTS_WITH	35.6%	ca_COEXISTS_WITH	30.2%
madopar	INTERACTS_WITH	<-	sinemet	TREATS	->	PD	ca_INTERACTS_WITH	33.3%	ca_INTERACTS_WITH	49.7%	ca_COEXISTS_WITH	21.5%
cataract	COEXISTS_WITH	<-	uveitis	COEXISTS_WITH	->	MS	ca_COEXISTS_WITH	43.7%	ca_COEXISTS_WITH	49.0%	ca_COEXISTS_WITH	35.2%
cataract	COEXISTS_WITH	->	uveitis	COEXISTS_WITH	->	MS	ca_COEXISTS_WITH	44.6%	ca_COEXISTS_WITH	43.8%	ca_COEXISTS_WITH	38.3%
cataract	COEXISTS_WITH	<-	rubella	COEXISTS_WITH	->	MS	ca_COEXISTS_WITH	43.7%	ca_COEXISTS_WITH	49.0%	ca_COEXISTS_WITH	35.2%
PD	COEXISTS_WITH	->	parkinsonism	CAUSES	<-	salsolinol	ca_CAUSES	100.0%	ca_COEXISTS_WITH	59.3%	ca_INTERACTS_WITH	21.7%

Figure 3: Results

sifier runs. This is the related to the cross-validation used to determine how well a classifier may work.

At this time, one of the discoveries identified in the 1980-1984 corpus has been documented in more current literatures - Parkinson's disease is treated by Madopar. When carefully studying the 1980-1984 corpus, we found that PD and Madopar were mentioned in the same documents - however, the corpus we used did not contain these documents.

Some of the results observed with Weka are summarized in figure 3. The first 5 rows of results were originally discovered partial LBD triangles. The second set of 5 are the reverse permutation where A-B-C is flipped to show C-B-A concepts with directions of relationships also swapped. The first row in each group is the PD to Madopar LBD pair along with simemet as a linking B term. The arrows indicate for example an A to B (−>) or a B to A (<−) relationship. The first row says that sinemet treats PD and that sinemet interacts with Madopar. The results from the classifier for the first row is that, from J48 classifier, Madopar may treat PD (the ca_ prefix indicates the direction of the relationship and means it's a C to A relationship). The confidence probabilities have a wide range and the best that may be concluded is that Madopar may treat PD with higher probability than coexists, causes and interacts with.

As has been noted in others' works, LBD techniques may generate much noise and may also only generate a very few number of worthy discoveries ((Kostoff et al., 2007; Preiss, 2014)). Our results are similar. However, the Parkinson's to Madopar discovery and subsequent automatic relationship explanation by our system is encouraging especially since there are papers that confirm that PD is treated with Madopar.

During our experimentation, we learned that preparing data for classifiers affected how the classifiers performed. The best approach so far is to include the direction of the relationship along with the semantic type of the relationship and to use both permutations of the data - A-B-C gets flopped to show C-B-A triangles (lower portion of the results table).

So far, not many successful relationships have been explained. We would like to perform more studies using more than just 500 randomly selected candidate B and C terms and possibly use less connected concepts instead of those with a greater than 900 connections. We would also like to apply this classification approach to the named entity like people, places, organizations and events and learn if we can explain LBD candidate discoveries in this domain in the same fashion that we explain LBD pairs in the medical domain.

5 Conclusion

We have presented a novel approach that uses classification to explaining LBD relationships. We performed LBD using an LSA approach using the Semantic Vectors package. We trained our classifiers on known fully qualified triangles found in SemMedDB. We also used SemMedDB to explain the relationships between the A and C concepts and the B linking terms. We have proposed additional research in another domain outside of the medical domain - that involving people, organizations, events and places.

References

Peter Bruza and Marc Weeber. 2008. *Literature-based Discovery*. Springer Publishing Company, Incorporated, 1 edition.

Trevor Cohen, Roger Schvaneveldt, and Dominic Widdows. 2010. Reflective random indexing and indirect inference: A scalable method for discovery of implicit connections. *Journal of Biomedical Informatics*, 43(2):240 – 256.

Aron Culotta, Andrew McCallum, and Jonathan Betz. 2006. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 296–303, Stroudsburg, PA, USA. Association for Computational Linguistics.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. 2011. Open information extraction: The second generation. In Toby Walsh, editor, *IJCAI*, pages 3–10. IJCAI/AAAI.

Murat C. Ganiz, William M. Pottenger, and Christopher D. Janneck. 2005. Recent advances in literature based discovery. Technical Report LU-CSE-05-027, Lehigh University.

Michael D. Gordon and Susan Dumais. 1998. Using latent semantic indexing for literature based discovery. *J. Am. Soc. Inf. Sci.*, 49:674–685, June.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Daniel Jurafsky and James H. Martin. 2009. *Speech and Language Processing, 2/E*. Prentice Hall.

Halil Kilicoglu, Dongwook Shin, Marcelo Fiszman, Graciela Rosemblat, and Thomas C. Rindflesch. 2012. Semmeddb: a pubmed-scale repository of biomedical semantic predications. *Bioinformatics*, 28(23):3158–3160.

Ronald N Kostoff, Joel A Block, Jeffrey L Solka, Michael B Briggs, Robert L Rushenberg, Jesse A Stump, Dustin Johnson, Terence J Lyons, and Jeffrey R Wyatt. 2007. Literature-related discovery (lrd). Technical Report ADA473438, OFFICE OF NAVAL RESEARCH, November.

Jiexun Li, Zhu Zhang, Xin Li, and Hsinchun Chen. 2008. Kernel-based learning for biomedical relation extraction. *Journal of the American Society for Information Science and Technology*, 59(5):756–769.

MEDLINE. 2012. Pubmed data files. U.S. National Library of Medicine. http://www.ncbi.nlm.nih.gov/pubmed/ or http://www.PubMed.gov/.

Judita Preiss. 2014. Seeking informativeness in literature based discovery. *ACL 2014, Workshop on Biomedical Natural Language Processing*, page 112.

Thomas C. Rindflesch and Marcelo Fiszman. 2003. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. *Journal of Biomedical Informatics*, 36:462–477.

Don R. Swanson. 1986. Fish oil, raynaud's syndrome, and undiscovered public knowledge. *Perspect Biol Med*, 30(1):7–18.

Don R Swanson. 1988. Migraine and magnesium: eleven neglected connections. *Perspectives in biology and medicine*, 31(4):526–557.

D. Widdows and T. Cohen. 2010. The semantic vectors package: New algorithms and public tools for distributional semantics. In *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*, pages 9 –15, sept.

Dominic Widdows and Kathleen Ferraro. 2008. Semantic vectors: A scalable open source package and online technology management application. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odjik, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

M. Yetisgen-Yildiz and W. Pratt. 2006. Using statistical and knowledge based approaches for literature-based discovery. *Journal of Biomedical Informatics*, 39(6):600–611.

Meliha Yetisgen-Yildiz and Wanda Pratt. 2009. A new evaluation methodology for literature-based discovery systems. *Journal of Biomedical Informatics*, 42(4):633 – 643.

Analyzing Foreign-Language Social-Media Reaction to Televised Speeches: Lessons Learned

Brian Amanatullah
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Steven Minton
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Matthew Michelson
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Greg Barish
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Kane See
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

***Abstract*—This paper updates our approach to analyzing social media response to speech events, such as the President's State of the Union address, presenting some new lessons learned in deploying this system. We previously described how we determine which specific lines in the speech resonate with different cohorts of the audience on social media (and how that alignment takes place, etc). In this paper, we describe an update to our approach that allows us to incorporate foreign languages into this analysis. That is, given a speech in English, such as the State of the Union, we describe how we can incorporate Tweets in foreign languages and still perform the clustering. From our development and deployment, we learned two important lessons to share: first, how to bootstrap the data collecting when supporting foreign language analysis, and second, how to order the clustering and translation operations to support a wide variety of languages with minimal cost (in terms of man hours).**

I. Introduction

In our previous work [1] we discussed our approach to clustering social media responses to live speech events, such as the US President's State of the Union address. We designed the approach to allow analysts, social scientists, and policy researchers to measure public reaction to various talking points (and visuals) in a speech. One can measure which points generate the most reaction, including those that may be surprising to the speaker and his/her staff. Further, marketers, journalists, political junkies and the general public can also use the system (and its related, public facing website) to better understand the speech's effect on different groups of people, their opinions, etc.

At its core, our approach is a clustering engine. It takes an input a set of social media posts (Tweets in this case), and a speech, and it creates clusters (e.g., groupings of Tweets that share some common topic in the language). It then classifies each cluster into one of three groups. In turn, our classifications are defined along two axes, which we call "referent" and "temporal." If a cluster refers specifically to a line (or lines) in the speech, we call this cluster "referent." This is in contrast to clusters whose topic is not directly related to the speech itself, such as reaction to what the viewers are seeing on the screen at that moment (e.g., the President's tie). We call these clusters "non-referent." By "temporal," we mean clusters whose Tweets occur in bursts, i.e., there is high volume of Tweets in a short interval. For instance, if many of the Tweets in a cluster happen to fall within a short time-window of one another, we call them "temporal." This usually happens when there is reaction to something specific in the speech's time-line, such a line of the speech that resonates with a large group or when a specific action happens on the screen, such as the camera panning to an audience member frowning. We contrast this behavior with "non-temporal" clusters, which are clusters that refer to the speech generally. For instance, a large number of Tweets commenting on the President's attire or appearance may be grouped together, but occur throughout the speech, since they are not tied to a particular point in time. By definition, we note that non-temporal clusters are also non-referent.

This leaves us with three classifications for a cluster: Temporal/Referent, Temporal/Non-Referent and Non-Temporal (since Non-Temporal cannot be Referent). This classification is important in understanding the reaction to the speech. For instance, if the cluster is Temporal/Referent, then we know it refers to that part of the speech at that time, and therefore that part sparked social reaction. If a cluster is Temporal/Non-Referent, then something in the broadcast outside of the speech itself, such as what is on-screen at that time, prompted reaction. Finally, we can exclude Non-Temporal clusters from the timeline analysis, since they would provide broad color (possibly), but not provide much deeper temporal analysis. We note that in previous work we demonstrated how we perform this classification [1]. Examples of each type of cluster are given in Table I.

Table I makes the clustering clear with a few examples taken from the 2014 State of the Union speech. The first cluster in the table is non-temporal. It reflects a number of Tweets from users watching the State of the Union speech, and Tweeting that they are doing so. The Tweets occur at various times throughout the speech, and do not, as a whole group, refer

TABLE I
DIFFERENT CLUSTER CLASSIFICATIONS

Non-temporal cluster
RT @tjholmes: Unless ur watching CSPAN, u might not know President of the United States is delivering State of the Union address n 50 mi...
What did you think of President Obama's State of the Union address?
#NowWatching President @BarackObama's "State Of The Union Address"...& you should be too!
Referent & Temporal cluster: reflects specific part(s) of the speech
Reacting to the line: "So lets get immigration reform done this year."
The tepid response to @BarackObama mentioning immigration reform tells me it probably won't happen this year.
Reps. Gutierrez (D-IL) & Diaz Balart (R-FL), partners in immigration reform, first to jump to feet after POTUS calls passing this year
RT @JuveMeza: Let's make this a year of action. Congress pass immigration reform or Obama should bypass you. #DACA4ALL #SOTU
Immigration reform has been tried since early in Obama's first term. With a partisan Congress, it's unlikely to happen this year, either.
Non-Referent & Temporal: Does not reflect specific part(s) of the speech
Theodore Roosevelt's 1941 #SOTU address? The Repubs are gonna hammer @WhiteHouse for that during mid terms.
Whoever runs the enhanced live stream sucks. They misspelled televised and showed a picture from "Theodor" Roosevelt's 1941 speech. #sotu
livestream needs a new fact checker. Pretty sure Theodore Roosevelt didn't deliver the 1941 state of the union address.
@WBLittlejohn Their slideshow is a riot. Lots of typos. Showed picture of SOTU from "Theodore Roosevelt" in 1941.

to a specific time period in the speech. The second cluster reflects Tweets about a specific topic that occurs at a specific time period in the speech. That is, the cluster is both temporal and referent (it refers to a line about immigration reform). The final example cluster in the table is temporal but non-referent. Instead, the time period reflected by the cluster is a point, just before the speech started, when the on-screen scroll showed an apparently incorrect historical fact.

While the main contribution of our approach is this ability to cluster the reactions, our previous work was limited because we could only analyze the reaction for the English speaking population. However, social and political scientists might also want to understand the reaction for non-English speaking audiences. Translation is a challenging capability to integrate for a number of reasons. For instance, where to fit the translation into the pipeline can have different implications (e.g., is it done early or late in the processing?). Another implication involves the initial data gathering. For instance, how do we bootstrap the collection process taking into account that our goal is to collect social media in foreign languages? Therefore, the crux of this paper is how we tie automated translation into our approach, focusing on our design choices and their implications (and how we addressed these questions).

Our overall approach for analyzing the social media response to a speech is given in Figure 1, which has been updated to emphasize the new translation capability. Briefly, the full system works as follows. During a speech, the system sources, collects and then cleans a set of Tweets. Next, the focus of this paper, we then translate the Tweets. We will discuss this architectural choice and its implications later. The Tweets are then clustered by topic, and also broken down by cohort, where each sub-cohort represents a group of users responding (for instance each cluster is further sub-divided into cluster members provided by men and those provided by women). Finally, the Tweet clusters are aligned temporally. This temporal alignment is the temporal classification we will focus on in this paper. Once the data is processed, we display the results on a webpage[1] where users can explore and analyze the results.

[1]http://www.socialreactiongroup.com

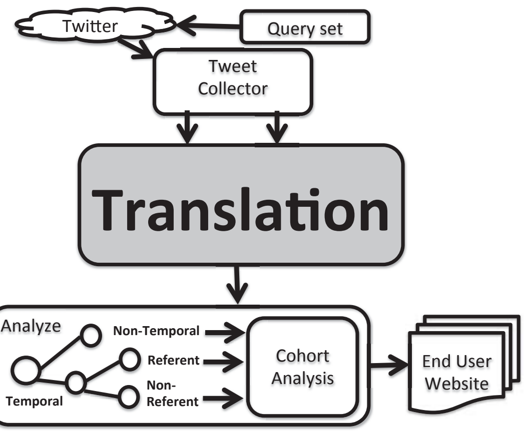


Fig. 1. Our architecture

As mentioned above, we deployed our approach in a public-facing website, where users can select various speeches and analyze the output themselves. A screen shot of this website is given in Figure 2. There are a number of design choices chosen used to make understanding the data more intuitive. The first involves the time aspect of the speech. At the top of the figure, there is a timeline with vertical bars. Each vertical bar represents a cluster linked to some line of the speech. The size the bar reflects the relative volume of Tweets on that line. So, taller bars mean more Tweets on that line (e.g., in the cluster). The bars are also split into colors, showing the proportion that one cohort represents of that cluster, versus the other. Therefore, at a glance, users can hone in on parts of the speech that have high volume, and determine those that looked skewed to a demographic. Then users can click on the bar to get details about that line of the speech and its associated cluster.

In the figure, we have clicked on one of the taller bars in the timeline (it is highlighted in grey) and the site has automatically scrolled to highlight that line of the speech (also highlighted in grey) and updated on the box on the bottom right with information about the cluster. The first item to note is text of the speech transcript on the bottom left. There are two visual aspects to the transcript. First, the size of the font reflects the amount of reaction to that line (e.g., the number of Tweets which is the size of the cluster). Therefore, lines with large text correspond to large clusters (much reaction) and those with small font have smaller clusters (less reaction). We picked a tall bar in the timeline, and as expected the associated line in the speed has huge font (it is one of the biggest clusters). Second, the text is color-coded if it appears that the cluster is skewed to one demographic. In this case, because this cluster skews significantly toward the female cohort, the cluster is colored pink (we color blue for the male cohort).

The second important aspect is the box on the bottom right, titled "Reaction Stats." At the top of this box are two infographics. The speedometer on the right reflects the relative number of Tweets on the speech line. "Going fast" (speedometer to the right) means many Tweets on the topic, and "going slow" (speedometer on the left) reflects few Tweets on the line. This speedometer updates in real-time as one scrolls throughout the speech, giving another view on volume. The pie-chart on the left reflects proportion of each cohort in the cluster for this line of the speech. In the example, we see that most of the cluster was identified as belonging to the female cohort.[2] The pie-chart not only shows the proportion, but also shows the z-score of the associated majority cohort (e.g., female in this case) which is the input to determine whether the cohort is significant or not. Finally, below the pie-chart, scrolling up and refreshing every few seconds, are actual example of random Tweets selected from the cluster.

One of the most interesting and relevant aspects to note for this paper is that the Tweets scrolling on the bottom right are in Spanish, even though the speech itself is in English. That is, we have aligned Spanish language social media reaction to the English speech. In fact, this visualization focuses on Spanish language reaction to the 2014 State of the Union address. As mentioned, Tweets will scroll in the bottom right, but those Tweets will be in Spanish (with their translation included). Therefore, we have allowed social and political scientists to perform deep analysis on language specific reaction to a public speech.

The rest of this paper is organized as follows. Section II describes the lessons that we learned incorporating translation into our analysis pipeline, and Section III contains our conclusions and future directions for this research.

II. Translation and Social Media Reaction

This paper focuses on the lessons we learned integrating translation into our social media analysis pipeline.[3] There were two important lessons, (i) when in the process to perform the translation, and (ii) improved mechanisms for gathering foreign language reaction. Here we discuss those lessons in more detail.

A. *Lesson I: When to translate*

As shown in Figure 1, our process kicks off with set of terms, called the "Query Set" (shown on the top right of the figure above). Ideally, we would collect every feasible Tweet during the speech and then filter out those that are irrelevant. However, this is impractical (both from a data and API service stand point). Therefore, instead we collect queries bootstrapped from terms in the query set. For instance, for the Statue of the Union, the query set might include terms such as "SOTU" (an acronym for State of the Union) or Obama. We are then given large samples of the Tweets that contain these terms. We continue this collection process for the duration of the speech we are analyzing (that is, we being collecting slightly before the speech begins, during its duration, and for a short time period afterward). In this way we ensure that our analysis captures the concurrent aspect of the reaction (though we note that we can align the Tweets to the speech using both topical and temporal analysis, as described above). Once Tweets are collected, they are then clustered and aligned to the speech for analysis.

Therefore, our design choice hinges upon where to perform the translation. We can collect the Tweets, translate them, and then perform our clustering. Or, we can collect the Tweets, cluster them, and then translate them to both align the cluster to the speech and to understand the driver of that cluster (e.g., the anchoring terms in the clustering).

One of the first important lessons that we learned was that it was preferable to translate the Tweets prior to clustering them, rather than vice-versa. This is largely an artifact of *data preparation and understanding.* Specifically, there are three important aspects to the language that dramatically improve clustering. First, it's often beneficial to perform stemming on terms before clustering (we use the Porter stemmer [2]). Stemming is an operation that turns words into their base form. For instance, a plural word might be stemmed into it's singular form, adjectives and adverbs to their base ("terrible" and "terribly" become "terribl"), etc. This allows clustering algorithms to consider variants of the same word to be the same when clustering, so that a high content word, in multiple forms, will yield the same basis for clustering. As an example, both "illegal immigrant" and "illegal immigration" stem to "illeg immigr."

A second important aspect to preparing the data for clustering involves stop words. For instance, clustering in general looks for words in common across text items (Tweets in our case), yet it's not meaningful if two Tweets share the words "and," "the," or "a." These common words that can be ignored are known as "stop words" and they reflect commonly occurring words that are generally not distinctive enough to be useful for tasks such as clustering. Our approach, like most, relies on a common stop word list, and knows to ignore

[2] Again, we note we always treat one pair of mutually exclusive cohorts at a time, e.g., men/women, red-state/blue-state, and users select between them on another page

[3] We thank and acknowledge SDL Language Weaver for helping us leverage their machine translation API in this effort.

Fig. 2. The SocialReactionGroup website

any words on that list. In general, stop words are built by examining word frequencies across a very large language corpus.

Finally, the third aspect is to understand synonymy since this might too form a basis for clustering. For instance, a Tweet that references "clapping" might also be clustered with one about an "ovation" (for instance, when talking about a Temporal but Non-Referent cluster, such as people describing audience reaction). Our synonym knowledge base included both WordNet [3] and some simple synonyms (and word substitutions) we derive from the corpus (such as POTUS is President).

The most important challenge is that while all of these aspects greatly improve clustering, they all depend heavily on the language. This is what drove our choice to translate the Tweets before clustering. Although translation might introduce noise (via imperfect translations), it allows us to leverage our knowledge of synonyms, stop words and stemming, using one common model, regardless of the input language. Otherwise, if we were to cluster prior to translation, while the input language might be cleaner, since it's not translated, it would required us to develop synonyms and a stemming algorithm, for each possible input language, which is challenging (we presume that developing stop word lists is easier to do). This is a costly proposition, requiring both development and testing of the models for each new language.

As clear examples of why this is problematic, consider the following cases from the Spanish language reaction to the State of the Union address. First, Table II shows some Tweets in a cluster centered around the common tokens "una," "por" and "que." We note that these Tweets have little common with one another, but because we did not have a collection of Spanish stop words for clustering, we ended up with this cluster.

Our next example, in Table III demonstrates two Tweets that should have been clustered, but were not. They both focus on investments in the United States versus China. If the system knew that EEUU was a synonym for Estados Unidos ("United States" in Spanish), then the clustering would have been clear. But, because we lack this domain knowledge in Spanish, they are separated.

Both of these serve to illustrate that the lack of language specific stop words, stemming algorithms and synonyms can lead to problematic clustering. Yet, if we translate before clustering, we can continuously re-use our knowledge about English to perform accurate clustering.

TABLE II
A CLUSTER AROUND SPANISH STOP WORDS: UNA, QUE, POR

RT @suvi_94: **Una** pregunta a los **que** votaran x el fmln. De aqu a 20aos,votaran **por** el viejo lin para presidente?Si no lo haran,porq van
RT @UniPolitica: Presidente Obama seala **que** las mujeres merecen igual paga **por** igual trabajo Que esto ocurra en 2014 es **una** vergenza #E
Espero **que** sea **por** **una** apuesta su estado por que soy capas de ir a buscarlo y pegarle.
@CFKArgentina sra presidente mientras ud esta de gira el pais es **una** cochinada aumento todo un veinte **por** ciento **que** dicen **por** el dolar
#EEUU Recin recib **una** nota de un amigo; dice **que** lleva +d 8 horas en trfico, en el mismo lugar, **por** la nieve. #Surprise!

TABLE III
TWO UNCLUSTERED TWEETS

Estados Unidos es el pais mas atractivo para invertir, por encima de China: Barack Obama
"@luisjorojas: Obama" China ya no es el mejor pais para invertir, **EEUU** Lo es #SOTU

B. Lesson II: Data Gathering

The second lesson involves collecting the data. As we discussed, we define a query set which acts as a filter, pulling Tweets out of the "Twittersphere" that we believe might be relevant. We emphasize that we are not making an assumption that any returned Tweet will be relevant. Rather, these are potential candidate Tweets that will be deemed relevant if they cluster appropriately.

TABLE IV
QUERY SETS IN ENGLISH AND SPANISH

English Query Set	Spanish Query Set
barack	democrata
barackobama	eeuu
barak	elpresidente
democrat	estado
obama	estado de la union
obama2014	estadodelaunion
potus	estados unidos
president	presidente
republican	republicano
sotu	
sotu2014	
stateoftheunion	
stateofunion	

When trying to perform a foreign language analysis, we again have two options. We can use the general query set, as described above, detect the language of the returned Tweets,[4] and then analyze buckets of Tweets in the target foreign language. While doable, this can be problematic because we are at the mercy of the sampling procedure of the given access methods provided by the data provider (such as the Twitter API). For example, using the query set shown in Table IV as "English query set," we captured 235,105 Tweets, but only 2.3% (5,523) were in our target language of Spanish.

An alternative is to modify the query set to include terms more aligned with the target language. For instance, for the State of the Union, rather than President, we may include the term "Presidente" in the query set (Spanish for President). This helps ensure a majority of the Tweets are in a specific target language. We contrast this approach by using the query set shown as "Spanish query set" in Table IV, which resulted in 25,136 Tweets, more than 99% of which were in our target language of Spanish.[5] We note that these query terms yield interesting results, which can be viewed on the website for this project.

While this is a small change overall, it is an important lesson to impart because it has a strong impact on the end results. If this step is not taken, then there may not be enough Tweets during data collection to provide a suitable analysis.

III. CONCLUSION

This paper describes our approach to incorporating translation into our social media analysis pipeline. In turn, this allows us to examine the social media reaction to speeches, even when the reaction is in a different language than the original speech. As before, we built a user facing website that allows policy analysts and social scientists to view the results and draw conclusions about the reaction.

Much of this paper focused on two important lessons we learned (and therefore shared) from this process. First, we argue that performing translation before clustering is beneficial, because it allows us to leverage our knowledge of language specific improvements to clustering (such as stop word lists, synonyms and language-specific stemming), even at the cost of potential translation errors. Second, we describe a small modification to our gathering of data that allows us to get more reaction in a target language, so we can ensure that we have a larger sample upon which to run our analysis.

REFERENCES

[1] B. Amanatullah, G. Barish, M. Michelson, and S. Minton, "Temporally aligning clusters of social media reaction to speech events," in *Proc. of the International Conference on Artificial Intelligence*, 2013.

[2] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[3] G. A. Miller *et al.*, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[4] We used SDL's language detection capability.

[5] We note that we did not include some obvious terms from the English query set, such as "SOTU" and "SOTU2014" in the Spanish set since they would have introduced English language Tweets. This helps explain some of the lower volume, since our query set could have surely been improved.

Distributed Dynamic Graph Analytic Framework: Scalable Layered Multi-Modal Network Analysis

M. Margitus[1] **and W. Tagliaferri, Jr.**[2]
[1]Information Fusion Group, CUBRC Inc., Buffalo, NY, USA
[2]Information Fusion Group, CUBRC Inc., Rome, NY, USA

Abstract – *Dynamic Graph Analytic Framework (DYGRAF) is a domain agnostic framework from which data alignment, data association, and layered multi-modal network analysis can be performed. Past installments of DYGRAF have been able to provide analytic insight for small and medium-sized data sets, however scalability becomes problematic as the amount of collected data increases. In this work, we discuss extensions to DYGRAF that allow analysis to be performed over data of considerable size. By augmenting its current modules with distributed computing components, DYGRAF's scalability can be extended to accommodate the effective analysis of the large amount of data being collected, as well as the massive layered multi-modal network derived.*

Keywords: multi-modal network analysis; distributed computing; information fusion; social network analysis; graph

1 Introduction

The introduction and incorporation of state-of-the-art data collection capabilities and platforms has afforded decision makers and analysts access to unprecedented levels of data. The sheer volume and disparity in content and representation of data within and across sources has created a challenge out of the tasks of analysis and identifying salient and/or actionable intelligence in a timely manner. To address these challenges, the analyst must be able to fully understand the environment of interest, requiring the ability to investigate interconnected relationships of many diverse data sources simultaneously, as they evolve both spatially and temporally. An effective way to meet this requirement is through the use of layered multi-modal network analysis.

Layered multi-modal network analysis (LMMNA) [1] is a technique in which multiple types of objects and relationships from a set of disparate layers of source data are represented, and analyzed as a graph, i.e. a finite nonempty set of vertices, together with a set of (un)ordered pairs of vertices called edges. The resultant network is valuable, providing a single graph which encapsulates the structural and semantic relationships of multiple data layers, and assembles previously disconnected information into a common picture.

Dynamic Graph Analytic Framework (DYGRAF) [2] is a framework through which LMMNA can be adeptly performed on small and medium-size data sets. The extensions presented in this work serve to extend the scalability of DYGRAF's analytic capabilities, to address the phenomenon of information overload, and provide timely situation awareness for analysts, improving their ability to better understand and anticipate activities within their area of responsibility.

The contribution of this work is a system that can be applied in order to achieve effective LMMNA at scale, through modifications and novel interactions between existing technologies. The remainder of this paper is structured as follows. Section 2 describes the current state, capabilities, and limitations of DYGRAF. Section 3 discusses the extensions implemented to achieve the scalability goals of DYGRAF. Section 4 discusses initial graph analytic results achieved through the distributed extensions, and Section 5 wraps up the discussion of DYGRAF and the extensions presented.

2 Dynamic Graph Analytic Framework

DYGRAF is a domain agnostic framework that provides the infrastructure to perform LMMNA via a modular design. At a high level, DYGRAF facilitates the alignment, association, and analysis of disparate layers of data to yield a cohesive and comprehensive picture of the evolving cross-layer situation.

The framework incorporates technologies that amalgamate, via standardization and normalization techniques, the disparate network layers; apply an entity resolution architecture to associate like vertices within and across layers; implement graph clustering and nesting techniques to (1) form a partition of associated information of interest from each layer and (2) introduce batch and real-time graph analytic algorithms and heuristics to derive and maintain graph measures, graph matching, and graph querying, to identify semantic and structural elements and discover new patterns of interest.

Four modules are responsible for DYGRAF's end-to-end functionality, illustrated in Figure 1: data alignment, data

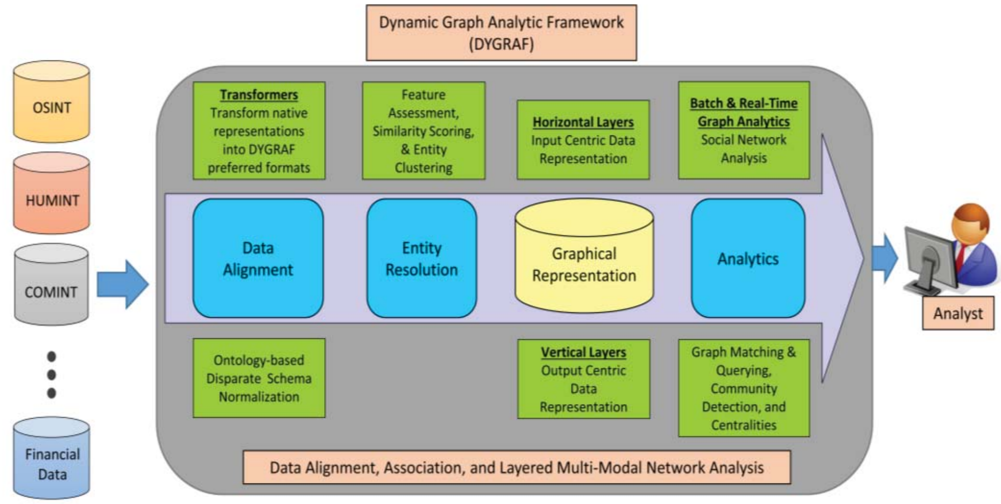


Figure 1. DYGRAF processing pipeline, including the data alignment, data association (entity resolution), Evidentiary Graph builder (graphical representation) and graph analytics modules.

association, Evidentiary Graph building, and graph analytics. Each module plays an important role in the preparation or analysis of the layered multi-modal network representation of the original disparate data sources.

The data alignment module is responsible for the standardization and normalization of heterogeneous data layers (e.g. communications, financial) through an ontological alignment process. To accurately compare disparate representations of entity attributes across data sources, a mapping is created to assign data instances to classes in ontologies. An ontology is a specification of a representational vocabulary for a shared domain, defining classes, relations, functions, and other objects [3]. DYGRAF utilizes ontologies comprised of classes that represent features and characteristics of people, locations, events, and other objects found within the data. Data alignment aids in determining the similarity of two entities during the entity resolution process by standardizing the representation of each entity's characteristics. This process eliminates the need to record and utilize each distinct representation of the same attribute (e.g. Last Name, Last, Surname); a set of descriptors that grows with the set of data sources.

The data association module utilizes homogenized data produced from the data alignment module to perform entity resolution. Within this process, entities are evaluated for similarity via attribute, rule-based, and semantic comparisons. Pairs of entities that meet or exceed a similarity threshold, subject to additional constraints, are merged. The process is performed hierarchically: within source, within layer, and across layers. Entity resolution is initially a batch process, then is incrementally performed as data updates, or additions, become available.

The results produced from the data association module are used in the construction of the Evidentiary Graph, DYGRAF's layered multi-modal network. The Evidentiary Graph is a single graph that encapsulates the structural and semantic relationships of the multiple data layers, as well as the unique entities and objects obtained through the entity resolution process. It provides a composite structured view from which emerging activities can be identified, and from which viable actions can be formulated. Within DYGRAF, layered multi-modal networks are represented by an attributed graph model (sometimes referred to as a property graph model), where objects and relationships are represented by vertices and edges, respectively, and there exists a variable set of attributes on each vertex and edge, describing the features of the object or relationship being modeled.

The graph analytics module utilizes the Evidentiary Graph as a starting point for LMMNA. The module consists of a set of algorithms and heuristics that execute a priori, or in real-time, to provide topological, semantic, and discovery analysis of the layered multi-modal network. Incorporated into this module are classical social network analysis centrality measurements; community detection algorithms; graph matching heuristics used to find user defined patterns within the Evidentiary Graph; graph querying capabilities; subgraph extraction tools; and path calculation algorithms. The plug-in design of the module permits extensibility and customization for an analyst's objectives, allowing algorithms to be added or exchanged, if needed.

3 Distributed extensions

DYGRAF has the capacity to achieve data alignment, entity resolution, and LMMNA on small and medium-sized data sets. When the Evidentiary Graph exceeds these

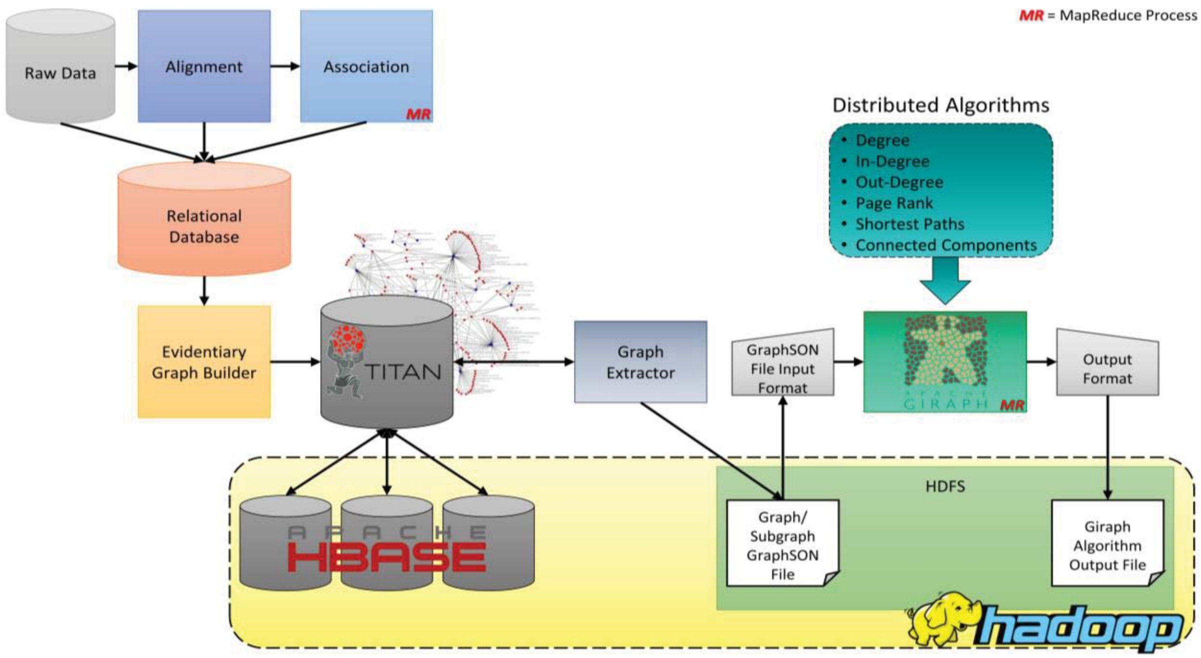


Figure 2. Extensions of DYGRAF, incorporating distributed technologies to improve analytic scalability.

thresholds, the computation time and ability of traditional algorithms included within DYGRAF begin to degrade, preventing the analyst from effectively meeting his or her objectives. Investigating massive data sources, which are quickly becoming typical, require the modules of DYGRAF to be augmented with components suited for distributed computing.

The development of distributed computing frameworks, for example, Apache Hadoop [4], and graph processing systems, have produced an improved toolset for large scale analysis than what has been previously available. Leveraging these advancements, extensions to DYGRAF have been developed and implemented to improve the scalability aspects of the data association module and the graph analytics module. Existing distributed technologies have been incorporated into these modules, and have been leveraged in order to perform large-scale entity resolution, distributed storage, and iterative distributed graph processing within a Hadoop environment, Figure 2.

3.1 MapReduce data association

To efficiently analyze and glean the best intelligence from a layered multi-modal network, the information described within the network must be robust and, at the same time, have a terse representation. In disparate layers of data, multiple mentions of a single entity often appear across a set of data sources. Generating a layered multi-modal network, and including these entities without a resolution, association, or deduplication process would result in multiple occurrences of the same entity within the graph, increasing the size of the network and segmenting information.

DYGRAF utilizes an entity resolution process to prevent duplication and segmentation of entities, their attributes, and their relationships. Entity resolution is the problem of identifying when entity mentions from within and across multiple data sources refer to the same entity. The process can be decomposed into two steps: scoring and clustering. The scoring step measures the similarity between entities for each pair represented in the data. The clustering step attempts to use the scores generated from the scoring step to group entities by their true unique identities.

DYGRAF implements entity resolution within the data association module using a non-parallelized graph association approach [5]. As the number of entities under investigation increases, a limit is reached where the amount of pairwise similarity comparisons and assignments exceeds the computation capabilities of a single computer. Transitioning from a linear data association module, where scoring and entity resolution are performed sequentially across all data sources, to a MapReduce-based association module, shown in Figure 3, eliminates the scalability limitations for the entity resolution process.

The MapReduce data association extension utilizes normalized data from the data alignment module, recording unique entity data to the Hadoop Distributed File System (HDFS). A MapReduce job is executed to read this data and compute similarity scores between every pair of entities. This

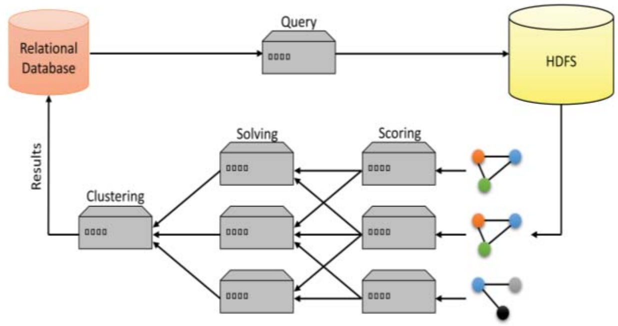


Figure 3. The MapReduce data association module. Data are retrieved from the relational database and transferred to HDFS where MapReduce jobs are executed to calculate similarity scores and determine which entities shall be clustered into a single unique entity.

sequence satisfies the scoring step. For the clustering step, a MapReduce job is executed to read the similarity scores and, based on the constraints of the model, decide when to assign similar mentions to a cluster. Each cluster represents what the algorithm has determined to be a unique entity—information used during the construction of the Evidentiary Graph.

The data association extension holds an advantage over DYGRAF's ordinary entity resolution process, providing parallelization during the scoring and assignment (solving) steps. The extension decreases the execution time needed for this task and, in situations which require the analysis of extremely large data sets are being analyzed, enables the module to complete its task when it otherwise could not.

3.2 Graph storage and data access

The large-scale topology of the Evidentiary Graph requires a storage solution that permits efficient data querying and analysis capabilities on a graph structure. Relational databases are proficient at persisting and querying regular structures, however, their schemas and table definitions are more rigid, and less suitable for a graph representation, especially when representing dynamic graphs, where new data fields may be introduced at each iteration of data ingest. Standard graph-oriented flat files, such as GraphML, also do not provide an acceptable solution, as these types of files aren't easily modifiable, and are more difficult to interact with as the file size increases.

To avoid the shortcomings of the aforementioned storage solutions, a distributed graph database has been chosen as the storage method for this extension. A graph database provides more flexibility over other storage solutions, such as relational data models, and does not require a predefined schema or consistent data keys across each entry of the same type. These criteria are especially important when modeling data that evolves over time, data containing complex relations, or data that has the potential for a multitude of query perspectives. The inherent graph structure retained through the usage of a graph database for persistence allows straightforward integration with other graph-based technologies (e.g. graph servers, graph visualizers, graph query languages), requiring little transformation from data structure to graph model implementation.

Titan [6] is a graph database capable of being distributed across multiple computer cluster nodes, and has flexible backend storage. Backed by Apache HBase [7], Titan provides DYGRAF with the ability to distribute an Evidentiary Graph, too large to be stored on one computer, across multiple machines, and it provides support for thousands of concurrent query transactions.

3.3 Large-scale graph exploration and analysis

Effectively analyzing massive disparate sources of data through DYGRAF's LMMNA capabilities require the graph analytics module to accommodate large-scale Evidentiary Graphs. Existing analytics within the module, which include traditional graph algorithms and social network analysis metrics, are capable of analyzing small and medium-sized Evidentiary Graphs. To mitigate the scalability shortcomings of graph analysis, DYGRAF's distributed analytical capabilities have been augmented with scalable graph technologies, namely, the TinkerPop graph technology suite, and Apache Giraph, a distributed graph processing system.

The TinkerPop graph suite [8] is an open-source collection of graph technologies which provide property graph modeling and interaction libraries, graph query and traversal capabilities, and graph exposure mechanisms. Within the distributed extensions to DYGRAF's graph analytics module, TinkerPop Blueprints (property graph API), Gremlin (graph traversal/querying), and Rexster (graph server) are valuable components that allow DYGRAF to be interoperable with other third party applications that utilize these technologies, such as Titan, which implements the blueprints API graph model for its graph representation. Additionally, the TinkerPop Blueprints API has been chosen as the common graph API throughout each of DYGRAF's previously established graph capabilities, as well as DYGRAF's extensions.

The graph querying capabilities of DYGRAF have been extended with the incorporation of TinkerPop Gremlin, a graph traversal language capable of composing highly-detailed, flexible, and sophisticated graph queries across a property graph, executed in a depth-first manner. By incorporating the ability to construct and execute Gremlin queries over the Evidentiary Graph, a user can filter the graph by entity connections, entity or relationship attributes, conditional statements, or custom functions for the data under investigation. Gremlin also provides a mechanism for users to calculate shortest paths between two entities, and derive

Table I. Graph Analysis Comparison

Data Sets	Graph Statistics			
	Vertices	*Edges*	*Local Graph Analytic Runtime (min)*	*Distributed Graph Analytic Runtime (min)*
VAST 2010	547	1064	0.339	7.73
Global Terrorism Database	230132	632238	-	13.33

subgraphs of the Evidentiary Graph, such as a *k*-hop neighborhood.

Apache Giraph [9] is leveraged to execute DYGRAF's large-scale graph and social network analysis algorithms on the Evidentiary Graph, and derived subgraphs. Giraph is an open source iterative graph processing system, modeled after Google's Pregel [10], that implements the Bulk Synchronous Parallel computing model for parallel computing [11]. Each algorithm developed using the Giraph API is designed from a vertex-centric parallel perspective, as opposed to the traditional "top-down" graph algorithm approach, which considers the entire graph throughout each step of the algorithm.

Extracting the graph topology from Titan, DYGRAF supplies custom input and output formats that instruct Giraph on how to interpret the data, and how to handle or process algorithmic results. For each distributed algorithm included in DYGRAF, a Giraph configuration and job is constructed and executed over the iterative processing framework, respectively. The results calculated by each algorithm are then made available directly as text files in HDFS, as well as written as additional attributes for each vertex/edge in Titan, so that they can be included in subsequent queries and analysis.

4 Initial analytical results

To illustrate the value provided by DYGRAF's distributed extensions, two data sets—one small, one large—were chosen to be analyzed as preliminary proof of concept exercises. The small data set was constructed from the VAST 2010 data set [12], augmented with additional synthetic data, to represent hospital admittance, financial, and communication records. The Evidentiary Graph derived from this set consisted of 547 vertices and 1064 edges, each with varying sets of attributes. The large data set consisted of records from the Global Terrorism Database (GTD) [13], which included individual/faction information, event occurrences and event descriptions. The resulting Evidentiary Graph contained 230132 vertices and 632238 edges.

To compare runtime and processing ability, both data sets were analyzed using a subset of social network analysis algorithms included in DYGRAF, as well as their counterparts introduced through DYGRAF's distributed extensions. As the objective of the exercise was in part to show a proof of concept, the distributed extensions were executed on a single VM pseudo-cluster, providing 8GB of RAM. The results are presented in Table I.

Analysis results of the VAST 2010 data set show that the non-distributed version of DYGRAF outperforms the distributed extensions with respect to runtime on small data sets that fit into memory. This disparity is mainly caused by the overhead incurred in order to configure and execute Giraph jobs. The real value of the distributed extensions is illustrated through the results of the GTD analysis. Although the runtime to execute the set of SNA metrics was 13.33 minutes, the same set of metrics did not complete using non-distributed methods. This accomplishment is valuable in that the distributed extensions allow DYGRAF to perform analyses on data sets that it had not been able to achieve previously. These initial results provide motivation for subsequent experiments with a multi-node cluster, to test scalability limits and runtime.

5 Conclusions

DYGRAF is a domain agnostic framework used to provide layered multi-modal network analysis resulting in a cohesive and comprehensive picture of the current cross network situation faced by analysts and decision makers, enabling them to correlate and investigate activity and time occurrences in related networks. DYGRAF provides the capabilities to reduce the amount of mental correlation required by the analyst, while maximizing their ability to understand and potentially predict when and where a major event or activity of interest is about to occur. By leveraging data collected from multiple disparate sources, DYGRAF is able to shed light upon the relationships between diverse and seemingly unrelated information.

As data collection techniques and storage capabilities become more advanced, the amount of data that exists describing events and activities of interest has grown considerably. Furthermore, to fully understand the information found in this data, there is a desire to analyze the sets in their entirety, with the goal of capturing any trends, patterns, or hidden relationships. In order to meet these needs and eliminate scalability restrictions, DYGRAF has introduced distributed computing extensions to its framework. Augmenting its capabilities with high-performance computing technologies allows DYGRAF to perform the large-scale data alignment, data association, and LMMNA needed to fully exploit the massive data sets that are becoming common.

6 Acknowledgment

The authors would like to thank Mr. Craig Anken and Mr. John Spina of the DYGRAF program management team for their help and support, as well as Dr. Gregory Tauer for his MapReduce data association research.

7 References

[1] P. LaMonica, T. Waskiewicz, "Layered multi-modal network analysis of textual data for improved situation awareness," *In* Proc. *International Conference of Knowledge Engineering (IKE)*, July 2011.

[2] M. Margitus, W. Tagliaferri Jr., M. Sudit, P. LaMonica, "Dynamic graph analytic framework (DYGRAF): greater situation awareness through layered multi-modal network analysis," *Proc. SPIE* 8402, Evolutionary and Bio-Inspired Computation: Theory and Applications IV, 84020E (May 1, 2012); doi:10.1117/12.920598

[3] T.R. Gruber. "A translation approach to portable ontologies," *Knowledge Acquisition,* 5(2); 199-220, 1993.

[4] *Apache Hadoop* [Online]. Available http://hadoop.apache.org

[5] G. Tauer, R. Nagi, M. Sudit, "The graph association problem: mathematical models and a Lagrangian heuristic," *Naval Research Logistics (NRL)*, vol. 60, issue 3, pp. 251-268, March 2013.

[6] *Titan Distributed Graph Database* [Online]. Available http://thinkaurelius.github.io/titan/

[7] *Apache HBase* [Online]. Available http://hbase.apache.org

[8] *TinkerPop* [Online]. Available http://www.tinkerpop.com

[9] *Apache Giraph* [Online]. Available http://giraph.apache.org

[10] G. Malewicz, M. Austern, A. Bik, J. Dehnert, I. Horn, N. Leiser, G. Czajkowski, "Pregel: a system for large-scale graph processing," *In Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data* (SIGMOD '10). AMC, New York, NY, USA, 135-146.

[11] L. Valiant, "A bridging model for parallel computation," *Commun. ACM* 33, 8 (August 1990), 103-111.

[12] *VAST Challenge 2010* [Online]. Available http://hcil2.cs.umd.edu/newvarepository/VAST%20Challenge%202010/challenges/Grand%20Challenge%202010/

[13] *Global Terrorism Database* [Online]. Available http://www.start.umd.edu/gtd/

Determining Formal and Informal Organizational Hierarchy

J.M. Brown, E.A. Benagh, C.G. Fournelle
Boston Fusion, Corp., 1 Van de Graaff Drive, Burlington, MA

Abstract—*Social hierarchy influences how information flows within groups of people, but it is not obvious how interactions within social networks reflect or differ from formal hierarchies and how well one can be used to predict the other. Using records from a business unit of ~6,000 employees, we determined formal organizational structure from the LDAP (Lightweight Directory Access Protocol) and informal structure from email communications over the course of one year. We compared features within the email network with relationships in the formal hierarchy. We then trained a SVM classifier to predict both supervisor-supervisee pairs and pairs within the same LDAP group based on email network features. The relationship from supervisee to supervisor was reflected most strongly in the email features. In all cases, the classifier found strong false positives which may reflect the cross-matrix structure of this organization and indicate mentoring or team relationships not included in the formal structure.*

Keywords—*graph analysis, social network, machine learning, hierarchy determination*

1. Introduction

Hierarchy is an integral component of human social organization, affecting how people relate to each other, how information travels within groups, and how organizations evolve over time. In organizations, hierarchies are often formally defined with supervisors, groups, and clear chains of command. However, this formal hierarchy may not reflect the actual working hierarchy of people's day-to-day interactions. Social communication networks, such as emails, reveal these day-to-day interactions but carry no explicit record of hierarchy. If we can determine the underlying connection between social networks and formal hierarchies, we might be able to infer one from the other. Further, differences between these two networks reveal discrepancies between the true organizational structure and the "effective" organizational structure.

Prior research examined social networks with respect to organizational hierarchy. In [8], the authors propose an automated detection algorithm for social hierarchy by developing a 'social score' for each user and comparing user scores to determine their rank within the levels of the organization. The algorithm demonstrated success for recognizing high levels of the organization but struggled with the lower echelons. Likewise, [9] generated a ranked list of node importance based on an entropy model. An algorithm for determining the type of organizational structure is presented in [6]. Many studies have been limited by difficulty in finding real-world email data. Several papers (e.g., [4, 8]) focus on the Enron email corpus containing ~600,000 messages between 158 senior employees (e.g., [10]).

In this paper, we focus on characterizing the links between individuals to identify the nature of the relationship. The premise is that interactions with supervisors and group members have distinct characteristics regardless of organizational level. If we can determine these small-scale connections, then we can infer the large-scale organizational structure by aggregating the connections. We use records from a business unit of ~6,000 people to provide the large volume of data suited to a machine learning approach. In the sections that follow, we discuss the data collection, individual features in the email network with respect to the organizational hierarchy, and a support vector machine (SVM) classifier for predicting formal relationships from the email communications. We then present and evaluate our results, summarize our findings, and discuss ideas for future improvements.

2. Data

In order to test the effects of hierarchy, we used real-world information for employees of a large company. All computer activity for the ~6,000 employees was tracked over the course of a year from January 1 to December 31, 2014. Table 1 contains summary statistics of the data properties. Due to the sensitive nature of the data collected, limited information was available for export and exported information was carefully anonymized.

The company maintained an LDAP (Lightweight Directory Access Protocol) for all employees to manage information access, providing a definitive guide to the formal organizational hierarchy. We recorded daily changes in the LDAP files to identify events such as an employee changing supervisor, moving departments, and joining or leaving the organization. The company hand curated the LDAP; hence, events may be offset from the record by up to a couple of days.

The email communications provide an extensive view of the social network within the organization and form the basis for building the informal organization structure. We have no record of email content due to the private nature of the information. Email features include basic metadata of the email address that was used to send the message, all recipients (as email addresses), when the message was sent, the length of the subject line, and the size of the contents and attachments. All email addresses were consistently anonymized to enable researchers to reconstruct the structure of the communication network without revealing the real identities of individuals that were using the email accounts. Some users in the LDAP do not appear in the email network, either due to a lack of records or difficulty reconstructing the email address to user connections stripped in the anonymization process.

Table 1. Properties of the LDAP and email data between Jan. 1, 2014 and Dec. 31, 2014

LDAP properties	
#LDAP Employees	6,562
#Departments	954
#Supervisors	827
#Groups	1,075

Email properties	
#Employees	4,794
#Connections	260,318
#Total emails	3.8M
#Email addresses	36,811
#Supervisor links	4,030
#Group links	36,861

3. Identifying Formal Hierarchy

To determine the formal hierarchy, we identified supervisor-supervisee pairs within the LDAP. To protect company proprietary information, some higher level employees were excluded from the sample for which email data was collected. We identified all pairs of employees that belong to a common group, where we define a group as employees that share a supervisor and the supervisor themselves. Group sizes ranged from 2 to 698 with a median of 5. The structure changed daily (although not significantly) as people were hired, quit, or changed supervisors and departments.

Figure 1 shows the organization hierarchy on a supervisor group basis (i.e., with each node representing a group, and a link indicating communications between the associated groups). We canonically identify each group node with the supervisor, and we display node size in proportion to group size. When more than one supervisor relationship existed over the time period, the figure shows the longest lasting relationship to provide an overview of the structure. The highest level supervisor is generally not tracked in the email sample (shown in the figure in blue), due to the corporate sensitivity of high level executive work. Within the data set, there are two large hierarchy structures and one group of several hundred people who have no additional subordinates within the group. Overall, there are 28 structured multi-group components plus an additional ~50 supervisor groups led by supervisors who are excluded from the email sample.

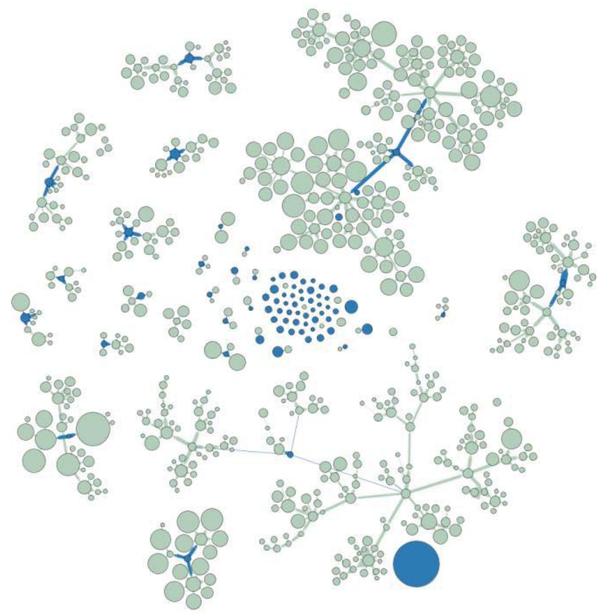


Figure 1. The organizational structure of supervisor groups. Every node represents a supervisor group with the size proportional to the size of the group. Light green nodes are groups that are led by individuals within the experimental sample, while blue nodes are led by supervisors outside of the email sample. Edge weights show the fraction of the year the connection existed.

4. Identifying Informal Network

To construct the communication network among employees, we first connected all the anonymized email addresses to individual employees. Employees generally had several addresses and more than one mail application, with each application recording email addresses in slightly different formats. The anonymization process then stripped the original information, making it more challenging to associate differently formatted addresses. The dataset also includes activity for group email addresses with large distribution lists that were used by multiple people.

The email tracking software recorded two types of email events: *send* events, when the user sent an email, and *view* events, when the user viewed an email, which included initial viewing of email messages as well as rereading events. We created a dictionary linking all user IDs to email address variants in two steps. First, we linked all the sender addresses in the send events with the user ID associated with the event. We then examined all the view events to identify other email variations. We dismissed view events that had more than one recipient, as it was too difficult to determine which of the addresses belonged to the individual associated with the event. We also required that either the sender or recipient address was already in the dictionary to determine whether the user ID was associated with the sender or recipient address. If both addresses were already in the dictionary, we added the recipient address to a list of group emails sent to multiple recipients and removed the address from the dictionary. The dictionary clearly had imperfect performance due to limitations in the data, finding that only ~50% of supervisor-supervisee pairs exhibited email links. Some relationships were short term, on the order of days and weeks, and it is possible no emails were exchanged during those transitions.

Once the dictionary was as complete as possible, we captured graphs on a monthly basis. Each node in the graph represents a user, and the directed connections indicate that an email was sent. Each connection has a weight equal to the number of emails sent. Since single emails could be counted multiple times for each read-through, it was infeasible to accurately interpret the 'viewed event' records as weighted communication links. As a result, we chose to ignore the 'view events', and focused on the cleaner 'sent events'. Because supervisor and group relationships change with time, we excluded months where the relationship was not in place when generating email features.

5. Comparison of individual features

We examined five features to identify traits that might support inference about supervisor-supervisee or group relationships. First, we examined whether email volume was higher between supervisor-supervisee and other intra-group pairs. Figure 2 shows the results for the intra-group pairs. When low total numbers of emails were exchanged, the pairs overwhelmingly crossed group boundaries. If a pair had sent hundreds of emails, over half of those pairs are within the group.

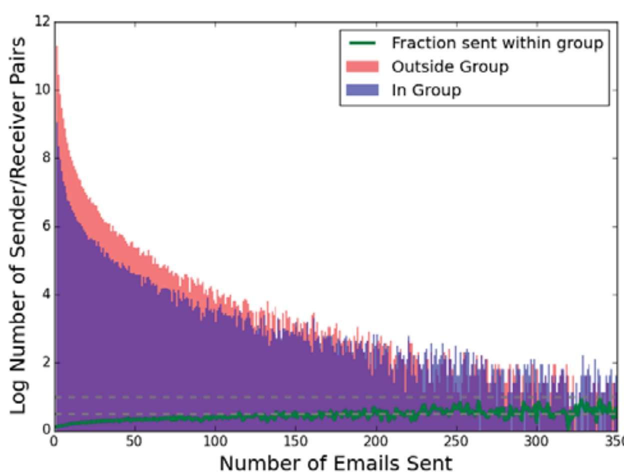


Figure 2. Comparison of emails sent within groups versus across group boundaries. The colored bars represent how many pairs received that number of emails. Note that the 'in group' and 'outside group' bars are overlaid rather than stacked. The green line shows the fraction of pairs which were within group for each number of emails sent. The two horizontal dashed grey lines mark 0.5 and 1.

The second feature we examined was the rank within the contact list. For each pair, we generated two rank scores: the first score indicates the position in the second's contact list, ordered by volume of emails, and the second score indicates the position of the second individual in the first's contact list.

As shown in Figure 3, employees generally have their supervisor as the first or second person in their contact list. Interestingly, this observation held at all levels of the organizational hierarchy. We divided the sample into those with direct reports and those without and saw little difference in the resulting distribution of rank scores. Supervisors generally contact their supervisees more frequently than others, but the effect is more diluted due to supervisors having multiple supervisees.

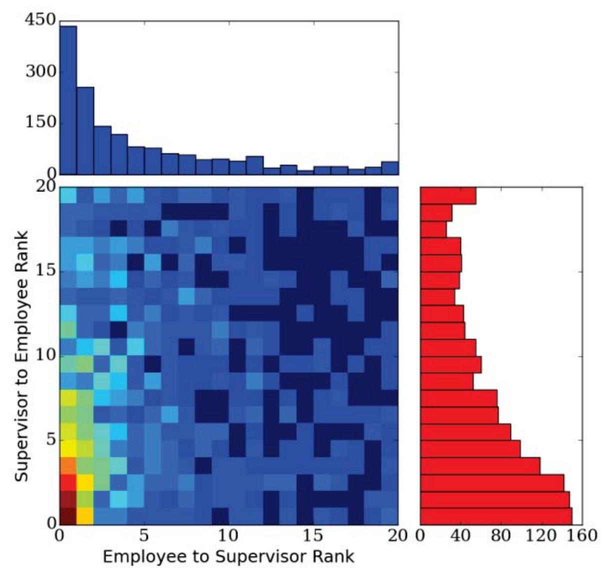


Figure 3. The supervisor to employee rank is the position of the employee in the supervisor's list of contacts ordered by number of emails sent, while the employee to supervisor rank in the supervisors position in the employee's list. The heatmap shows the 2-dimensional histogram of the two ranks, while the 1-dimensional histograms on the edges show the collapsed histogram in each direction.

Finally, we examined the similarity of the communication patterns of the pair. We found all the maximal cliques within the email graph using the algorithm first proposed by [2] and implemented by [1]. We then counted the number of cliques within which a pair co-occurs as a measure of overlapping circles of contacts. We observed that most pairs are not in group, but the ratio of in-group to out-of-group increases with number of clique memberships, as shown in Figure 4.

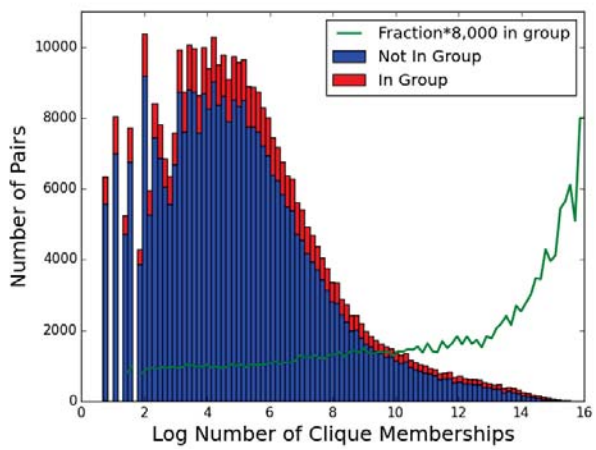


Figure 4. Histogram of the number of maximal cliques in which both individuals in a pair has membership in the same group. The histogram is stacked with the 'in group' population appears on top of the 'outside group'. The green line shows the fraction in group, multiplied by 8000 to align the scales.

For each individual in every pair of users, we computed the fraction of the user's total email volume that was sent to the other individual in that pair. The final feature was the mean size of the emails sent based on the number of characters in the email body to further characterize the significance of their interaction.

6. SVM Classifier

We used the support vector classifier (SVC) in the Python library 'scikit-learn' to combine all the individual features to produce better classification. We generated two classifiers, one to identify supervisor-supervisee from other relationships, and another to classify relationships as internal to and across group boundaries. We normalized all features to have a mean of 0 and a standard deviation of ±1 so that the different scales for the features did not influence the classification. The classes are highly unbalanced, with most relationships being neither supervisor-supervisee nor within group. In fact, supervisor links comprise only 1.5% of the connections, whereas group connections account for 14% of the total. To compensate, we weighted the classes by the inverse of frequency. The dataset was large enough that the classifiers still included instances of the underrepresented classes within the training sample. We explored using both linear and radial basis function (RBF) kernels, but achieved slightly better performance with the linear kernel. The training sample consisted of 50,000 instances with ~210,000 instances in the target population. Increasing the training size to 100,000 did not improve performance while greatly increasing runtime.

There are some limitations for classification over this set of data. First, the email structure (unsurprisingly) does not perfectly reflect the LDAP hierarchy. Second, the classifier can only evaluate relationships present in email, but some supervisors never email their supervisee, and vice versa. Finally, we are aware that the management for this organization is "matrixed", and individuals receive work assignments under managers outside of their formal LDAP structure. We therefore do not expect the classifiers to be able to achieve perfect performance.

7. Results

The classifier achieved meaningful classification of both supervisor and group relationships. Figure 5 shows the receiver operating characteristic (ROC) curve for both the supervisor-supervisee classifier and the group classifier. The area under the curve (AUC) for the supervisor classifier was 0.79 while the AUC for the group classifier was 0.72. The supervisor classifier had better performance than the group classifier, indicating more identifiable relationships between supervisor and supervisee than within groups. The group classifier is likely weakened by group members who have some communication but are not strongly tied. There is no indication that all members of a supervisor group actively worked together, and in the case of the largest 698 member group, it seems highly unlikely. That said, enough group members are connected strongly, and so the classifier can achieve meaningful performance.

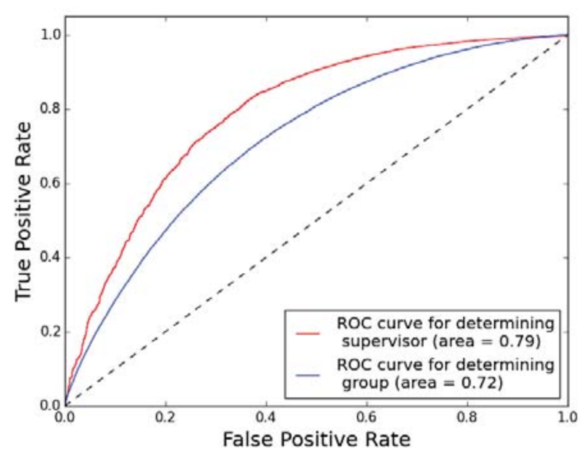


Figure 5. ROC curves for determining supervisor (red) and group membership (blue).

8. Evaluation

We present the optimal prediction results for both classifiers in Table 2. We determined the optimal threshold by maximizing the distance from the random chance line.

Of particular interest are the false positives identified by the classifiers. These may represent relationships that mimic those of supervisor-supervisee and group (such as team leaders and teams working on the same project), but are not connected in the formal hierarchy, partly due to the matrixed organization style. We have no method to check these work arrangements in this dataset, but we examined these pairs and predicted supervisors in more detail. Of the false positives from the supervisor classifier, ~12,000 are group relationships but not supervisor relationships. Of all the false positives, 90% of the pairs represent 1,969 individuals who are incorrectly predicted as supervisors in five or more pairs. As the number of instances in which a person is predicted as a supervisor increases, it becomes more and more likely that the person plays a key role in the informal hierarchy. The average number of predictions identifying an individual as a supervisor was 9 and the largest number of predictions was 144 false positive pairs.

Table 2. Classifier performance results for the test portion of the data set

Supervisor	Predicted Supervisor	Predicted Not
Actual Supervisor	*1,444*	*832*
Actual Not Supervisor	*36,367*	*171,735*

Group	Predicted Group	Predicted Not
Actual Group	*20,242*	*11,930*
Actual Not Group	*54,227*	*123,979*

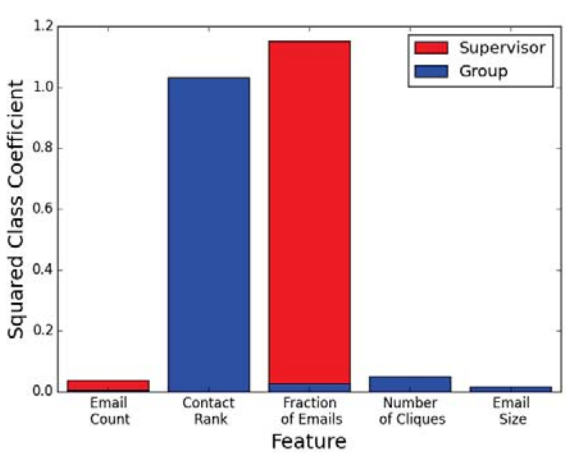


Figure 6. The feature importance for the two classifiers. Both classifiers have strong dependencies on single features.

To understand the importance of different features, we used the squared weight coefficients for each class, as determined by the linear classifier (as in [6]). Interestingly, both classifiers seem to strongly favor a single feature, but they favor different features. The supervisor classifier depends heavily on the fraction of an individual's messages that are sent to the supervisor, while the group classifier mainly uses contact rank.

9. Next Steps

In the future, we plan to utilize the insights into organizational hierarchy we gained from the email network to identify information flows through the network and to search for anomalies in communication. We will investigate the strong false positive to look for connections indicating work team relationships. We will also expand the feature sets for the classifiers to include features such as frequency of contact, response time, and number of attachments sent.

The technology we are developing will benefit operational customers who are trying to build an understanding of information and influence networks. Directed social networks are present throughout modern life, including social media such as Twitter, Facebook, and LinkedIn, as well as email, SMS, and telecommunication networks. Unlike companies, these social networks rarely come with a blueprint to the underlying hierarchical structure. It is only by inferring the hierarchy that we can understand the interpersonal connections. The relationship classifiers we developed here provide a method to locally identify rank within the large graph rather than globally trying to rank individuals.

10. Summary

We present a study of the organizational hierarchy of a corporate business unit of ~6000 people over the course of a year. We examined both the formal and informal hierarchy to determine how closely connected the two were and developed a classifier to predict formal relationships from the informal email network.

We found significant differences between the formal and informal organizational structure. Formal relationships may have no equivalent in the informal structure or may be less intense than expected. The informal structure includes many connections with no formal counterparts, as might be expected. Formal supervisor and group relationships, however, were noticeably different from other connections within the email network.

We examined five features in the email network for their relationship to the formal hierarchy: email volume, contact rank, fraction of user emails sent to contact, number of joint maximal cliques and email size. We then used a support vector machine classifier with a linear kernel to simultaneously use all the features for classification. Classification of supervisor relationships had an area under the ROC curve of 0.79 while group classification had slightly poorer performance at 0.72. The email network thus reflects supervisor-supervisee relationships more strongly than group structure, although supervisor relationships are highly underrepresented in the sample as a whole.

The results from the classifier hint at a parallel informal structure that mimics the formal hierarchy. Strong false positives from the classifier may identify relationships that are functionally similar to supervisor and group, such as team leader and team. Identifying the actual organizational structure from communications provides a more complete organizational picture than depending on the incomplete formal hierarchy.

11.Acknowledgements

The authors gratefully acknowledge support for this work from DARPA through the ADAMS (Anomaly Detection At Multiple Scales) program for funding project GLAD-PC (Graph Learning for Anomaly Detection using Psychological Context), under U.S. Army contract number W911NF-11-C-0216. We gratefully acknowledge the support of Xerox PARC and our teammates there including Dave Gunning, Gaurang Gavai, Sricharan Kumar, and John Hanley. Any opinions, findings, and conclusions or recommendations in this material are those of the authors and do not necessarily reflect the views of the government funding agencies.

12. References

1. Aric A. Hagberg, Daniel A. Schult and Pieter J. Swart, “Exploring Network Structure, Dynamics, and Function Using NetworkX”, in Proceedings of the 7th Python in Science Conference (SciPy2008), Gäel Varoquaux, Travis Vaught, and Jarrod Millman (Eds), (Pasadena, CA USA), pp. 11–15, Aug 2008.
2. Bron, C. and J. Kerbosch, “Algorithm 457: Finding All Cliques of an Undirected Graph”, Commun. ACM 16, 9, 575-577, 1973.
3. Clauset, A., M.E.J. Newman, C. Moore, “Finding Community Structure in Very Large Networks”, Phys. Rev. E (70), 066111, 2004.
4. Diesner, J., T.L. Frantz, K.M. Carley, “Communication Networks from the Enron Email Corpus”, Computational & Mathematical Organization Theory, 11, 3, 201, 2005.
5. Gupte, M., P. Shankar, et al. “Finding Hierarchy in Directed Online Social Networks”, WWW 2011, Mar. 28 – Apr 1, 2001, Hyperabad, India, 557, 2001.
6. Guyon, I., J. Weston, S. Barnhill, V. Vapnik. “Gene Selection for Classification using Support Vector Machines”, Machine Learning, 46, 389, 2002
7. Maiya, Arun S., Tanya Y. Berger-Wolf, “Inferring the Maximum Likelihood Hierarchy in Social Networks”, Proceeding of CSE ’09 (4), 245, 2009.
8. Rowe, R., G. Creamer, S.Hershkop, S. Stolfo, “Automated Social Hierarchy Detection through Email Analysis”, Joint 9th WEBKDD & 1st SNA-KDD, San Jose, CA, Aug. 12, 2007
9. Shetty, J. and J. Adibi, “Discovering Important Nodes through Graph Entropy: The Case of Enron Email Database,” LinkKDD '05, ACM, 74, 2005
10. http://www.cs.cmu.edu/~enron/

SESSION

KNOWLEDGE BASE SYSTEMS, REPRESENTATION, DISCOVERY + INFORMATION ENGINEERING METHODS AND RELATED ISSUES

Chair(s)

TBA

Using Multiple Methods to Infer Classification with An Incrementally Expanding Knowledge Base.

K.H. Cheng
Computer Science Department, University of Houston, Houston, Texas, USA

Abstract – *Many problems are solvable by several alternative methods. Each method is associated with some level of confidence, which depends on the kind of knowledge used. In this paper, we identify several methods to infer the classification of a target. We develop two algorithms, one used when the target is a category, while the other when the target is an object of a category. Our solution applies the methods in the order of non-increasing level of confidence, and for methods with the same level of confidence, the more versatile method is used first. Besides the answer, our solution also provides the certainty of the obtained answer. In the presence of incomplete knowledge, three-valued logic is involved in the reasoning of the answer. Our solution is designed to use knowledge at ancestor categories. This reduces drastically the total amount of knowledge required to be learned, thus alleviating the burden of the teacher.*

Keywords: Inference, reasoning, three-valued logic, knowledge, and artificial intelligence.

1 Introduction

It is very common for a problem to be solvable by several methods each using a different kind of knowledge. However, each method may only be appropriate for some subset of input instances. The correct methods and the corresponding kinds of knowledge used to solve the problem are determined by the specific input instances. In other words, given a specific input instance, not every method is able to provide the correct answer. We will focus our discussion on problems that will result in either a positive/true or a negative/false response. These problems correspond to the problems posed by a decision question in the English language. Decision question has the following sequence grammatical structure: main verb followed by complete subject and then by subject complement, omitting the question mark punctuation to simplify our presentation. The scope of knowledge needed to answer all decision questions encompasses all kinds of knowledge. Some decision questions correspond to undecidable problems and some answers are unknown such as “Is P=NP” [1]. The subject complement of some decision questions may be or involve an adjective such as “Is John tall” or “Is John a tall person”. It may also include adverbs such as “Is John very happy”. In this paper, we are interested in answering the subset of decision questions where the subject complement is a noun and that noun represents knowledge about a category. Specifically, we are interested in finding the correct answer to the question “Is target a classification”, which corresponds to the problem of inferring the classification of a target. Category allows human to use a name to refer to a group of objects with a common property. The ability to infer the category of a name is very useful. It may be used to determine the correct intention of a sentence [2-3] and to disambiguate multiple meanings of words in sentences [4].

One complication of the problem results from the possibility of not having the knowledge needed to find the answer since our solution is a part of a learning program in which knowledge is acquired incrementally over time. Consider the scenario that the learning program is loaded into multiple robots. Since the experience of each robot is different, knowledge known at a certain time by various robots may be different. Consequently, different answers may be obtained for the same question by different robots, and the answer to the same question may also change over time due to changes in the knowledge base. In the presence of incomplete knowledge, the answer to a question may be true, false or unknown, and our solution has to involve three-valued logic. The reasoning to arrive at an answer differs based on two different relationships among the alternative methods. If the results of the alternative methods supplement each other deficiencies, i.e., their results may contradict, then the reasoning may be formulated simply as the logical or [5] of the results obtained by the various methods. However, if the results of the alternative methods conform to each other, the reasoning is different and cannot be formulated as the logical or of the various results. The details of these two cases will be explained in Section 2.

Another complication of the problem is the need to determine the order of application of the methods since the correct answer may be obtained by many methods. Our solution applies these methods in the order of decreasing reliability of the individual method starting from the most reliable. For equally reliable methods, the one that is more versatile will be used first. A method is considered very reliable if the answer is proved mathematically or quantitatively, e.g., an answer proved by using a definition. Conversely, a method using qualitative or subjective knowledge is less reliable, e.g., the basis knowledge requirement to be an expert or knowledgeable in a subject is fuzzy. A method is considered more versatile than another method if it can be applied in more situations. For example, the method of using a definition is considered to be more versatile than the method of using a sufficient condition. The reason is that definition may be used to prove both positive and negative results, whereas a sufficient condition can only prove positive results. In addition, our solution assumes that we have perfect knowledge, i.e., no contradictory pieces of knowledge that will lead to contradictory conclusions. Consequently, if the answer obtained by any method is conclusive, then there is no need to try the next method to check for contradiction. This assumption does not pose a problem because to reasonably resolve a contradiction, the answer provided by the more reliable method should be trusted over those from less reliable methods, so we will arrive at the same conclusion no matter whether we check for contradiction or not. This assumption saves an extra step of using all available methods to check for contradiction, which will take longer to arrive at an answer.

Since each method has its own level of reliability, the certainty of an answer is determined by the reliability of the method used to arrive at the answer. Besides arriving at the logical answer, our solution also provides the certainty of the answer. There are a total of 5 levels of certainty, namely: true, highly likely, likely, slightly likely and unknown. A certainty level of true corresponds to the highest level of confidence. For instance, an answer proved or disproved by a definition has the value true or false, respectively, both with the certainty level of true. On the other hand, an answer inferred by using the hierarchical relationships among various categories will have a lower level of confidence. The reason is that many of these hierarchies are taught by a "trusted knowledgeable" teacher without the need of justification. It is possible that an answer is wrong due to a teacher's mistake. An inferred false answer may also be wrong due to incomplete knowledge, which we will explain in Section 2. Logically, there is no need to have an unknown certainty level because a logical answer of true or false with an unknown certainty level is logically equivalent to an unknown answer with a certainty level of true. However, when the answer is actually unknown and if the program is forced to provide a logical answer that can only be true or false; the program will report it as false with a certainty level of unknown. Note that there is no need to have certainty level in the negative sense such as highly unlikely because a highly unlikely true answer is the same as a highly likely false.

There are systems that focus on answering questions such as IBM's Watson [6] and PENG Light [7]. Answering decision question is a sub-problem for a computer program communicating to a person. Recently, we have developed a natural language communication capability [8] for a learning program system, ALPS [9]. After learning a small subset of the English language [10], the program is capable of understanding declarative sentences involving actions [2], forms-of-be verbs [3], pronouns [11], and time adverbs [12]. It also automatically generates the role structures that the program can use to write a sentence based on a thought [13].

Given the decision question "Is target a classification", there are two inputs to the inference problem, namely the target and the classification. When the classification is a category, we identify several methods to justify or refute the classification of the target based on the knowledge learned about the category. These methods are named based on the knowledge used: definition, condition, basis knowledge, and hierarchical relationships, respectively. The hierarchical relationships among categories are used in [14] to infer the category of a given, while definitions are used in [15] to justify the classification of numbers in number theory. In this paper, two additional methods are used. The first method uses two types of conditions: a sufficient condition is used to justify a classification, while a necessary condition is used to refute the classification. For example, all four sides are equal is necessary for a quadrilateral to be a square. So, a given quadrilateral whose four sides are not all equal is determined to be not a square. The second method uses the basis knowledge for categories. For example, assume the category knowledge subject is taught to be based on what it studies. Since the knowledge subject Zoology is the study of animals while Biology is the study of live objects, we may infer that Zoology is Biology. Each kind of knowledge provides one method to solve the inference problem, but it is possible that not all methods are suitable for a specific input. Based on the classification being asked, which affects the exact kind of knowledge known by the program, the corresponding method should be used to arrive at the answer. Instead of focusing on using only one method, our solution uses whatever methods appropriate for the classification, i.e., for the same problem with different input, the methods used may be different.

The solution to the problem is also affected by the second input, the target. If the target is a category, then we may simply use the various methods just identified. If the target is an object, we may also use its category to determine the answer, e.g., John is a mammal because John is a human and human is a mammal. However, this is not enough since objects may hold many roles involving multiple hierarchies, e.g., a person may also be a Physicist, a father, a Christian, as well as a white person. Obviously, the object's category is not the appropriate knowledge to be used for these classifications, but some sub-knowledge of the object should instead be used. The exact sub-knowledge is determined by the classification being asked. For example, it may be what John studies if the classification is Physicist; and it may be John's occupation if the classification is a medical doctor. Consequently, if the target is an instance of an object, our algorithm will first use the target's category to see whether it is the asked classification. If the answer is true, then the correct answer is obtained. On the other hand, if the answer is false or unknown, then our algorithm will determine what sub-knowledge of the target should be used based on the asked classification. The sub-knowledge, if known, will then be used to answer the question.

Due to the huge amount of knowledge required to be acquired by the learning program to answer any question, our solution is designed to minimize the total required amount. One way to achieve this is to use knowledge available at ancestor categories that descendants inherit, thus requiring only the knowledge to be known at the root rather than in all descendants. Another way is to assume that objects of a category cannot possess an aspect if that knowledge is not present. This makes he knowledge must be taught explicitly only if objects of a category can possess an aspect, which is a much smaller amount of knowledge. Although this assumption will allow wrong false answers when the program is ignorant, these mistakes will be corrected once the program becomes educated, i.e., when the appropriate knowledge has been acquired. All these minimize the amount of required knowledge, thus alleviating the burden of the teacher.

The rest of the paper is organized as follows. Section 2 presents the various methods in determining the answer to the inference problem. It also describes the two reasoning algorithms used when the target is a category and an object, respectively. Section 3 explains how the two algorithms work on several examples each under various situations of the acquired knowledge base. Section 4 discusses in detail how the design of our solution minimizes the amount of knowledge required to be learned. Section 5 concludes the paper and discusses potential future extensions of the current work.

2 Methods and Our Solution

There are multiple ways to decide whether a target is a classification or not, but only some may be appropriate for the decision based on the involved classification. The first way is to use a definition, which specifies the necessary and sufficient condition of a category. If the definition is satisfied, the target is proved to belong to the category; otherwise it is not a member of the category. If known, the decision, either true or false, obtained by using a definition is considered to have the highest level of certainty, i.e., true. If the definition is not known, the answer using this method is unknown with a certainty level of true. Using definition is always the most preferred way to decide the classification because it is most reliable and versatile.

Since the definition of many non-Mathematical ideas may not exist, we have to rely on other methods to determine their membership. For some categories, the membership criteria may be learned separately over a period of time by using sufficient conditions. Although each sufficient condition is not necessary for objects to belong to a category, satisfying one is sufficient to conclude that the given target belongs to the category. Besides sufficient conditions, some categories may have well known properties, which are necessary conditions. A target violating a necessary condition proves a negative result, i.e., the target does not belong to the category. Similar to definition, an answer obtained by using either a condition, either sufficient or necessary, is also given the certainty level of true. If there is no known condition, the answer using this method is unknown with certainty true. Although the method of using condition is as reliable as using definition, each kind of condition is only applicable to a subset of situations, and so this method is considered as less versatile than the method of using definition.

The next alternative way uses the fact that a category is classified based on certain aspects of the target satisfying some criteria. Since many of these bases are verbal that may be fussy in nature, answers obtained by this method is given a certainty level of highly likely instead of true. If the basis knowledge is known for the classification, it may be used if the target is also known to have a specific value for that aspect. The answer by this method is unknown if either no basis knowledge is known for the classification or the value for the basis of the classification is unknown for the target. If the basis knowledge of the classification is known but the category of the target is not known to have that aspect, the answer is false instead of unknown. The reason for this decision is discussed in Section 4.

Finally, the correct classification may also be inferred by using the hierarchical relationships known among the various categories. These relationships are taught by a "trusted" teacher without the need of justification. Due to the vast amount of hierarchical relationships among categories and the incremental acquisition of knowledge by our program, it is highly likely that these learned hierarchies are incomplete. If the inferred answer is false, it is possible that the answer is wrong due to incomplete knowledge. For example, if the root categories of human and live-object are animal and physical object, respectively; then the answer to the question "Is a human a physical object" is false. The reason is physical object is not an ancestor category of human because of the missing knowledge of animal is a live-object. This false answer is obviously wrong. For an answer obtained by using the method of hierarchical relationships, if it is true, the confidence in our answer is highly likely since the hierarchical relationship has been explicitly taught and inferred. If it is false, our program would infer the answer as unknown only when both the target does not have a parent and the classification does not have any children because these conditions indicate that the knowledge is probably newly learned, and has a high likelihood that the learned knowledge is incomplete. Otherwise, the inferred answer is false if either the target already has a parent or the classification already has some sub-categories. These conditions indicate that at least some and hopefully most hierarchical relationships have been learned. The certainty level of the inferred false answer is still highly likely if the target and the classification are dependent, i.e., they belong to the same hierarchy with a common root. However, if two independent hierarchies are involved, the certainty of the inferred false answer drops to likely because of the possibility that the false answer is wrong due to incomplete knowledge.

Our solution uses Object-Oriented paradigm so that the implementation of the algorithm is distributed to different classes. Given a decision question, our learning program, using the learned grammar, creates a decision question role which contains two logical objects, the target and the classification. Based on the knowledge kind of the target, different algorithms is applied from the corresponding class of knowledge. First, we will discuss the algorithm when the target of the question is an instance of a category. Each method may be used to arrive at the answer depending on what kind of knowledge has been taught to determine the classification. Our solution applies these methods in the order of non-increasing reliability starting from the most reliable method. If two methods are considered to be equally reliable, then the method with better versatility would be used first. Based on the reliability and versatility of the various methods, the order of the methods used in our algorithm is as follows: definition, condition with sufficient condition followed by necessary condition, basis knowledge, and hierarchical relationship. Since these methods are alternative solutions and we apply the more trusted methods first, if the answer obtained by any method is conclusive, i.e., either true or false, then the answer is returned with its corresponding certainty level, without the need to try the next method. The reason is that the answer is supported by one legitimate method with some confidence. If the knowledge required by a method is not available, then the answer by that method is unknown, and our algorithm will try the next alternative method to determine the answer. Specifically, if the definition knowledge is available, then it is used; and its result, whether true or false with a true certainty, will be returned. Currently, the answer is considered as unknown if no definition knowledge is known; so our algorithm moves on to use the next method. For the condition method, our algorithm will first use sufficient conditions, one at a time. If the target satisfies a sufficient condition, then a true answer with true certainty is returned. If the target fails all sufficient conditions or the classification does not have any sufficient conditions, then our algorithm will use necessary conditions, again one at a time. If the target fails a necessary condition, then a false answer with true certainty is returned. The answer using the condition method is unknown with a true certainty and our algorithm will try the next method of using basis knowledge when there are no more conditions. Once again, if a definite answer of true or false is obtained using basis knowledge, the answer will be returned. Finally, the last method of using hierarchical relationships for inference is attempted if all earlier methods result in an unknown response. The answer is unknown if none of the methods is able to provide either a true or false answer. Note that the above reasoning scheme of the solution is different from Kleene 3-valued logic [5]. Although these are alternative methods to obtain the answer, one might think that it can be formulated as the "logical or" of the results of the various methods. For both Kleene 3-valued logical or and our solution, the answer to the question is true if the result of any one method is true and the answer is unknown if the results of all methods are unknown. However, if one method yields a false answer and the others are unknown, then the combined answer is unknown in Kleene 3-valued logical or. This is because the answer will be true if one of the unknown answers turns out to be true, but the answer will be false if all the unknown answers turn out to be false. This conclusion of unknown is clearly different from the conclusion made by our solution, which is false. The reason for this difference is that the results from all the methods, although alternative, should agree with each other, i.e., if one answer is false, then none of the other answers can be true. On the other hand, the conditions in a logical or are independent and may contradict each other, i.e., if one condition is false, the others can be either true or false. Therefore, our reasoning cannot be formulated simply as a logical or of the results of the various alternative methods.

When the target is an instance of an object of a category, our solution will first use the algorithm on the category of the object. The reason is that the object may belong to the category that is or is not of that classification. For example, given a human object John, one can easily infer that John is a mammal since human is a mammal. One would also infer that human is not a robot, and would like to conclude that John is not a robot. However, if a false answer is arrived using the category of the object, it is still possible that the object is that classification. For example, for the question "Is John an engineer", human is obviously not an engineer using John's category, but John is still possible to be an engineer based on his occupation. Note that here the reasoning is the same as the Kleene 3-valued logical or. A false answer using the object's category is possible to be overturned as true if knowledge within the object is used instead. As a result, when using the algorithm on the object's category, only a true answer is conclusive and our algorithm returns that as the answer. Conversely, a false or unknown answer is not conclusive, so our algorithm will attempt to use some sub-knowledge of the object to determine the answer. Once again, if the classification has a definition or sufficient conditions or necessary conditions, it is possible that the object may have the necessary knowledge needed to prove that it belongs to the classification or not. For instance, the category quadrilateral is neither a rectangle nor a rhombus, but an object of a quadrilateral may be proved to be a rectangle but not a rhombus by using the measurements of the quadrilateral object and the respective sufficient and necessary conditions of rectangle and rhombus. Those needed knowledge of the object are specified explicitly in the conditions, and so can be determined readily. Now, if both methods of definition and condition results in an unknown answer, our algorithm will attempt to use two more pieces of knowledge that may be obtained from the object. First, if the program knows what the classification is based on, knowledge obtained most probably from the root category of the classification, then our program knows what basis knowledge to search for in the object and the classification. If both the classification and the object have the basis knowledge, then those bases knowledge will be used to determine the answer. For example, if the target is John and the classification is biologist, and suppose the root category of biologist is specialist. Assume the program knows that the basis of being a specialist is knowledgeable in a specific knowledge subject, then our algorithm will look for what subjects biologist and John are knowledgeable in. If both are known, then they are used to determine the answer. Second, if it is not known what the classification is based on, our algorithm will attempt to search within the object for the knowledge about the root category of the classification. The found knowledge may then be used to determine the answer using hierarchical relationships. For example, assume now the classification is engineer and the root category of engineer is occupation, and that the basis knowledge about engineer and occupation is not known. Our algorithm will attempt to look for the occupation knowledge within John, the target object. If the object has occupation knowledge, then it will be used to determine the answer.

3 Examples

We will first consider cases when the target is an instance of a category. Our first example is on answering questions when knowledge of all methods is unknown except hierarchical relationships among categories. For example, if human is known to be a mammal and a mammal is an animal, it is easy to infer that human is an animal [14]. However, when human does not have a parent category and animal does not have any children, the answer is unknown. If some but incomplete hierarchical knowledge about them has been taught, then the answer is false which is obviously wrong. However, the false answer may actually be correct if the target is not an animal such as Hibiscus.

Our second example is on answering questions for cases when both definition and conditions are unknown but basis knowledge is known. For instance, for “Is Zoology Biology”, assume knowledge subject is classified based on what it studies, and both Zoology and Biology are knowledge subjects. Further assume the following has been taught also: Zoology is the study of animals, Biology is the study of live objects, and the hierarchical relation animal is a live object. By the time our algorithm tries the basis knowledge method, it discovers that knowledge subject, an ancestor category of Biology, is classified based on what it studies. Since what Biology and Zoology study is known, the basis knowledge method can be used. Finally, our algorithm concludes that Zoology is Biology since animal is a live object is confirmed by the same algorithm using hierarchical relationship. On the other hand, using similar assumptions, our algorithm concludes that Sociology is not Biology because Sociology is about the study of society.

Our third example involves answering questions when the target belongs to multiple hierarchies each classified by a different basis. Consider the question “Is Zoology a Science”. As in our first example, it is easy to infer that Zoology is a Science if complete hierarchical relationships are known. However, let us assume that only Biology is a Science is known, but Zoology is Biology may be decided instead by the method of basis knowledge as in our second example. One expensive way to infer that Zoology is a Science is to find out that Zoology is Biology by going through exhaustively all the descendents of Science. However, assume Science is classified based on its method of learning and that method is by experiment. If Zoology also has the knowledge that its method of learning is by experiment, we can easily infer that Zoology is a Science using the basis knowledge of Science. Currently our algorithm will not go through the descendents of a category. It can arrive at the correct answer as long as the appropriate basis knowledge has been learned.

Our fourth example involves answering questions that may use our recursive algorithm on multiple levels of basis knowledge such as “Is zoologist a scientist”. Assume each is a specialist, the basis to be a specialist is his/her knowledgeable subject, and each specialist’s subject is known, e.g., zoologist is knowledgeable in Zoology. Assume again the definition and conditions of scientist are unknown, our algorithm infers that zoologist is a scientist using the basis knowledge of scientist and the answer of the previous example, also by basis knowledge.

Next, we consider cases when the target is an instance of an object. We assume that the answer inferred by using the object’s category is not true since our algorithm finishes if the answer is true. Our first example involves questions when the classification is known to have basis knowledge, but no definition and condition. For instance, in “Is John a biologist”, the target is John and the classification is biologist whose root category is specialist. Obviously human, the category of John, is not a biologist. Assume the program knows that the basis to be a specialist is knowledgeable in a specific subject, then our algorithm looks for the subject biologist and John are knowledgeable in, respectively. We further assume that the program knows that biologist is knowledgeable in Biology and John is knowledgeable in several subjects. If at least one of those subjects is Biology such as Zoology, then the true answer can be inferred, otherwise the answer is false. Finally, if the program does not know which subject John is knowledgeable in, then the answer is unknown.

Our next example involves questions when definition, condition and basis knowledge are all unknown for the classification, so our algorithm will search within the target for the root category of the classification. For instance, for “Is John an engineer”, assume occupation is the root category of engineer and all the above knowledge are unknown for both engineer and occupation. So our algorithm searches within John for occupation, the root category of engineer. If the occupation of John is unknown, then the answer is unknown. If the occupation is civil engineer or teacher, the answer will be true or false, respectively. Similarly, if basis knowledge is unknown for biologist in “Is John a biologist”, our algorithm will search within John for his specialist information. If John is known to be a zoologist, then the answer is true. If John is a non-biologist specialist, then the answer is false, and if John's specialist knowledge is unknown, the answer is unknown.

4 Discussion

To provide intelligent answers, an enormous amount of knowledge needs to be taught to the learning program. We design our solution so that the total amount required to be taught is minimized. For example, all descendent categories must also satisfy the necessary condition of their ancestors, so besides using its own, the necessary conditions of all its ancestors are also used. This requires the condition to be taught only at the root and eliminates the need to learn them at each descendent, thus greatly reduces the amount of knowledge required to be taught. For the same reason, our solution also searches ancestors for basis knowledge.

Due to the vast amount of relationships and the incremental acquisition of knowledge, it is highly likely that learned hierarchies are incomplete. As explained in Section 2, the inferred false answer may be wrong due to incomplete knowledge. Ideally, in order to avoid incorrect false answers, the answer should be unknown if two independent hierarchies are involved, i.e., their roots are different knowledge objects. Whereas, if both categories belong to one single hierarchy, then the answer is true if the target is a descendent of the classification, otherwise, the answer is false. This would require teaching the program so that all categories are linked in one giant hierarchy with a single root. However, teaching all these is very tedious and error prone. It is very easy to miss out some relationships because of the vast amount of hierarchies and that many categories have no apparent relationships among them. Artificial categories and relationships may have to be created and taught to the program in order to link unrelated hierarchies together. More importantly, this poses a major problem when answering questions about an aspect of an object such as "Is John an engineer". If all hierarchies have the same root, it would be very difficult for the program to know what to look for in the knowledge object of John since occupation is only one of the many ancestors of engineer. Even though the program may know the occupation of John, the answer cannot be easily found. Whereas, if occupation is an independent hierarchy, then the program knows to look for occupation in John since the root of engineer is occupation. Consequently, we would not require nor want the program to be taught with one huge hierarchy, but instead with many hierarchies independent of one another.

Another issue arises on what needs to be learned when our solution is searching for an aspect of an object. It may make sense to find the occupation of John, but not sensible when the target is a building, so the answer to the decision question should be false. For example, it is false that the Empire State building is an engineer or a biologist simply because a building cannot have an occupation nor can it be knowledgeable in a specific subject. There are three situations when our algorithm has to find an aspect about an object of a category. First, if the category of the object is unknown to possess such knowledge, then the answer is unknown. For example, if human have not been taught to have an occupation yet, then it will be unknown whether John is an engineer or not. Second, if the object belongs to a category known to be unable to possess such knowledge, then the answer can be determined to be false. For example, an object of car is not an engineer if car is known to be impossible to have an occupation. Third, if the category is known to possess such knowledge, once again there are three possibilities. If the object instance does not possess such knowledge, then the answer is unknown. For example, if human is known to have an occupation but John's occupation is unknown, then it is unknown that John is an engineer. Otherwise, the known occupation of John can be used to determine whether he is an engineer or not. For example, the answer is true if John is a mechanical engineer, the answer is false if John is a carpenter, and the answer is unknown if both occupation has no descendent and the occupation of John has no parent, which is highly unlikely. The above solution, although is logically correct, will require the program to learn what each category can and cannot have; which imposes a huge learning burden on the program. We observe that although human can have a lot of different aspects, objects of most other categories are impossible to have each corresponding aspect. Teaching each category cannot have each aspect is ridiculously tedious since there are a lot of categories. The amount of learning can be greatly reduced if that knowledge is only taught to ancestor categories similar in idea for necessary conditions and basis knowledge. For each aspect human or some category can have, the teacher have to find several appropriate ancestors of all the other categories that cannot have that aspect to teach such knowledge. Since there are many independent hierarchies, it is highly likely that many of these impossibilities are still unknown to the program even after many years of learning. Without the appropriate knowledge, a lot of questions that obviously have the false answer such as "Is Golden Gate Bridge a physician" will result in an unknown answer, which is unreasonable for a well educated robot. This lack of common sense knowledge is a common problem for all learning programs. To ease the learning burden on the program, our solution assumes that a category cannot possess an aspect if it is not known that a category and its ancestors can. Thus the program is only required to be taught when objects of a category can possess an aspect. This assumption allows the program to correctly give the answers as false for a lot of common sense situations without the need to learn the knowledge explicitly. This is desirable because when the learning program is old, it should not give the answer as unknown since it is supposed to be educated. On the other hand when the learning program is young, if it has not learned that objects of a category can possess an aspect, our assumption will lead our solution to answer false, which may be a wrong answer. This is acceptable since the learning program is considered as naïve, and these mistakes will be corrected once the program acquires the appropriate knowledge.

5 Conclusion

In this paper, we have presented several methods that may be used to infer the classification of a target. We then develop two algorithms based on the kind of knowledge of the target, one used when the target is a category and the other when it is an object of a category. These methods are applied in the order of non-increasing reliability of the individual method starting from the most reliable; and for equally reliable methods, the one that is more versatile is applied first. Besides providing the answer, our solution also gives the certainty of the answer based on the method used to arrive at the answer. In the presence of incomplete knowledge, the answer may be unknown and the reasoning has to involve three-valued logic. Although these are alternative methods, if the results obtained by the various methods must conform to one another, i.e., cannot contradict each other, the reasoning cannot be formulated as the logical or of the various results. On the other hand, if the results supplement each other deficiency, i.e., may contradict each other, then the reasoning can be formulated as the Kleene logical or of the results. Our solution is designed to use knowledge available at ancestor categories. This design keeps the total amount of knowledge required to be learned as minimal as possible, and thus does not impose an undue burden on the teacher. One assumption made by our program to reduce the required amount of learning may cause wrong false answers if the program is ignorant. These mistakes are acceptable when the learning program is young and naïve, and will be corrected when the program is educated, i.e., once the appropriate knowledge has been learned.

In the future, each method may all be tried to either double check an answer or to find out a contradiction. This is especially useful because the learning program can then accept or reject newly learned knowledge based on whether there is no contradiction or has contradiction to existing knowledge, respectively. However, even with this error checking approach, it is still possible to learn wrong knowledge. Consequently, similar to how human is taught, learning the correct knowledge initially is important for our learning program. Since the current work corresponds to answering decision question when the classification is a noun, one extension is to work on cases when the classification involves an adjective such as “Is John tall” and “Is John a smart person”. Further work is also needed when adverbs are involved such as “Is John very happy”.

6 References

[1] J. Hopcroft, R. Motwani, J. Ullman. “Introduction to Automata Theory, Languages, and Computation”; Prentice Hall, 3rd Edition, 2006.

[2] E. Ahn, W. Faris, & K. Cheng. “Recognizing the Effects caused by an Action in a Declarative Sentence”; Proceedings of the International Conference on Artificial Intelligence, 149-155, 2009.

[3] W. Faris & K. Cheng. “Understanding and Executing a Declarative Sentence involving a forms-of-be Verb”; Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 1695-1700, 2009.

[4] W. Faris & K. Cheng. “A Knowledge-Based Approach to Word Sense Disambiguation”; Proceedings of the International Conference on Aritificial Intelligence, 111-117, 2013.

[5] S. Kleene. “On Notation For Ordinal Numbers”; The Journal of Symbolic Logic, 3, 4, 150-155, 1938.

[6] D.A. Ferrucci. “This is Watson”; IBM Journal of Research and Development*, 56*, 3.4, 1-1, 2012.

[7] R. Schwitter. “Working for Two: A Bidirectional Grammar for a Controlled Natural Language”; In W. Wobcke, & M. Zhang (Eds.), AI: Advances in Artificial Intelligence*,* Springer, 168-179, 2008.

[8] W. Faris & K. Cheng. “Performance of a Natural Language Framework”; Proceedings of the International Conference on Artificial Intelligence, 98-104, 2014.

[9] K. Cheng. “An Object-Oriented Approach to Machine Learning”; Proceedings of the International Conference on Artificial Intelligence, 487-492, 2000.

[10] W. Faris & K. Cheng. “An Object-Oriented Approach in Representing the English Grammar and Parsing”; Proceedings of the International Conference on Artificial Intelligence, 325-331, 2008.

[11] W. Faris & K. Cheng. “Understanding Pronouns”; Proceedings of the International Conference on Artificial Intelligence, 850-856, 2010.

[12] N. Leach & K. Cheng. “Recognizing and Understanding Time-terms in Communication”; Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 3023-3028, 2011.

[13] W. Faris & K. Cheng. “Generating a Sentence from a Thought”; Proceedings of the International Conference on Artificial Intelligence, 554-560, 2011.

[14] K. Cheng. “The Representation and Inferences of Hierarchies”; Proceedings of IASTED International Conference on Advances in Computer Science and Technology, 269-273, 2006.

[15] K. Cheng. “Representing Definitions and Its Associated Knowledge in a Learning Program”; Proceedings of the International Conference on Artificial Intelligence, 71-77, 2007.

Moving Towards an Adaptive Enterprise Intrusion Detection and Prevention System

T. Amick[1], L. Soles[1], and D. Snider[1]
[1]Department of Computer Science, University of West Florida, Pensacola, FL, USA

Abstract - *In this paper, we describe our plans to create a smarter network defense system through the collection and analysis of network signatures generated by real security threats. To meet this goal, we plan to create software agents interconnected to a central behavior analysis database service where each software agent records attack meta-information collected during previous intrusion attempts. The central database warehouses and analyzes the meta-information collected by the interconnected agents. The agents can then utilize both instantaneous and historical data by integrating rules derived from the data collection and analysis process into intrusion prevention policies. The result is a modular and scalable network defense system that should be more responsive and adaptable to imminent threats.*

Keywords: Decision Support Systems; Intelligent Networks; Distributed AI Systems and Architecture; Knowledge Discovery; Cyber security

1 Introduction

With continuously evolving threats coming across the Internet, intrusion prevention systems must respond and adapt. Many current systems lack the ability to record data about the attack and to make decisions on how to defend against the attack. These systems also lack the capabilities to share knowledge with other intrusion prevention systems on their organization's network, leaving each machine with the task of determining threats independently which increases the time in which the network is vulnerable.

Our goal is to create a modular enterprise intrusion detection and prevention system that is adaptive and responsive to new threats. We plan to create software agents interconnected to a central, shareable behavior analysis database service where each software agent records attack meta-information collected during intrusion attempts. The central database warehouses and analyzes the meta-information collected by the interconnected agents. The agents can then utilize both instantaneous and historical data by integrating rules derived from the data collection and analysis process into intrusion prevention policies.

As an institution in the .edu domain, we are defending against attackers who are trying to exploit our domain with the intention to steal intellectual property or the personal information of students, faculty or staff. It is our belief that these attackers are attempting to gather the same information at all universities; therefore collaboration with other universities should allow for easier prediction of network traffic and behaviors that do not fit the norm.

In Section 2 of this paper, we present the problem and related research. Section 3 is a discussion of our proposed method and in Section 4 we conclude with the explanation of what makes our proposal unique.

2 Background

In February 2015, the authors' network and servers were subject to an attack originating from outside the network. Once this intrusion was detected it would have been convenient for the other systems on the network to automatically update each other with information about the attack. This desire to improve the abilities to defend our own network are the impetus for starting this research project.

There has been much research on implementing machine learning algorithms to build intelligent intrusion detection systems. Intrusion detection using genetic-based learning algorithms have been explored in [1]. A hybrid intrusion detection system was demonstrated that incorporated anomaly detection through self-organizing maps and misuse detection using decision trees [2]. Another hybrid intrusion detection system utilized decision trees and support vector machine classifiers [3]. The utilization of clustering algorithms to detect intrusions without the need for training data is demonstrated in [4]. In [5], the use of mobile agents on the network to create an intelligent intrusion detection system framework was explored.

Commercially available intelligent intrusion prevention systems such as McAfee's Network Security Platform are standalone systems that do not communicate with each other and are dependent on feeds from McAfee [6]. These systems do come with a cost which might be difficult for smaller organizations to accommodate.

There are still many areas of improvement for intrusion detection. The first of these is the need to increase detection efficiency and decrease the false positive rate. Second, there is a demand to increase throughput while decreasing the associated costs. Third, there are opportunities to improve performance metrics and assessment methodologies. Fourth, research is needed to improve the capabilities of intrusion detection systems to defend themselves from attacks. Fifth, there are opportunities to refine the analysis of network data [7].

3 Proposed method

3.1 Overview

A high-level diagram of our proposed method is shown in Fig. 1. The data acquisition agents in the top left of the diagram are tasked with collecting data about the network. The output from the data acquisition agents are fed to a real-time analysis engine. From the real-time analysis engine the analyzed data is sent to a collector database and to a policy arbiter. The collector database houses data from multiple locations on the organization's network. The policy arbiter will create firewall rules from the real-time analytics and the host's policy database. The data warehousing, machine learning and decision support functions of our system are shown on the right side of Fig. 1 as our threat analysis clearinghouse. The data from the collector database is fed to an analytics system where various machine learning algorithms can be utilized to extract knowledge about threats to the network. Raw data from the collector database and from the analytics system can be visualized to help in the knowledge discovery process. Any actionable decisions that are made are stored in the analytics publisher database which then publishes its threat analysis data to the distributed policy databases on the system.

3.2 Data acquisition

We have several potential sources of data. Along with system logs produced by various services running on a system we can also collect data from software applications. Packet capture (pcap) files generated by packet capture and protocol analysis tools like Wireshark [8] could be utilized. Output from open source network intrusion detection systems such as Snort [9], Bro [10] and Suricata [11] can also be used as input to our threat analysis clearinghouse. We can also utilize Kismet for the acquisition of data about wireless network traffic [12]. Any information regarding port scanning or brute force attacks will also be valuable for analysis.

As one of our data acquisition agents, we propose to modify the open source intrusion prevention software Fail2ban. Fail2ban works on a single machine to take both instantaneous information and scans of the server log files to look for nefarious behaviors and the IP addresses from which these behaviors emanate [13]. Once this nefarious behavior and the IP address is identified, Fail2Ban can update the firewall rules to block the IP address for a specified amount of time to protect the machine. However, this intrusion prevention system lacks the vision of the whole network; therefore, it does not have the capability to make decisions that will benefit the network. We propose to record Fail2Ban

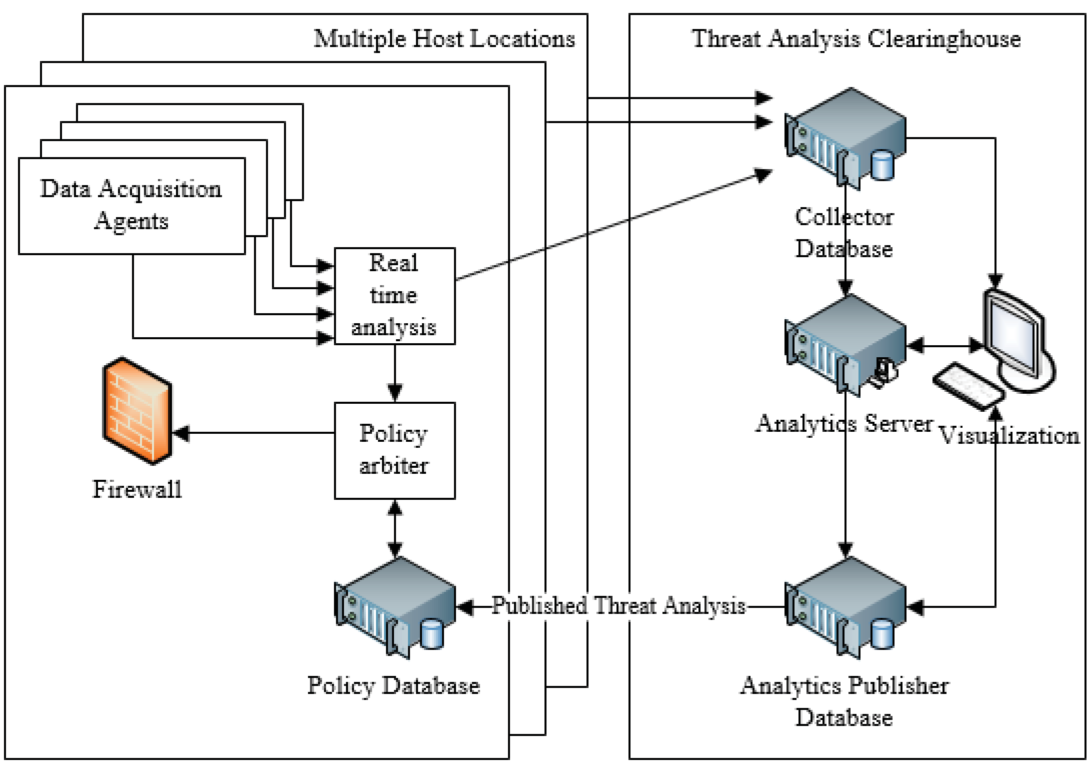


Fig. 1. High-level diagram of the proposed solution.

activity into a central database to facilitate all machines on a network to "instantaneously" update their firewall rules as well as be able to do offline mining of the attack data to develop permanent rule sets. Any new features we add to Fail2ban will be contributed back to the open source community. In our vision for this system, if someone were to develop a widget that produces the same output as our modified Fail2ban, then this widget could be used with our system.

The policy arbiter will build firewall rules from the real-time analytics supplied by the distribution agents and the host's policy database. Additionally, the policy arbiter applies local policies and filters. While the policy arbiter could be consistently configured for simple configurations, more likely it would need to be customized for each type of node (e.g. database server, web server, file server, etc.) and the traffic it should experience.

3.3 Knowledge discovery

The knowledge discovery process defined in [14] will be utilized in this project to extract knowledge from the data. As shown from left to right in Fig. 2, we will narrow the amount of data to be analyzed, pre-process this data by cleaning and integrating, transform the data to formats and features that are suitable for machine learning algorithms, and then mine for patterns that can yield knowledge about the threats. The data warehousing and decision support functions of our system will follow many of the principles established in [15]. Several machine learning algorithms will be available for implementation. These algorithms include classification, clustering and rule set generation [16] for creating models and allowing for prediction based on a priori and current conditions. For example, through our analysis we should be able to detect patterns of activity and predict there is about to be a denial of service attack and here is its target. So if we know the when and where, then we can also make the decision of what to do.

We propose to visualize both historical information collected by the database and the results of the machine learning algorithms. This information visualization will help with the analysis of the attack patterns by security experts and to explain attacks to stakeholders with a non-technical background.

The output from this decision support system will be a set of actions that characterize what we need to do, such as what IP addresses need to be blocked through the IP tables in Linux. The goal is to be firewall manufacturer agnostic, with the intention to disseminate threat knowledge quickly. Once the algorithm is designed for the overall system, we can use it internally for abhorrent behavior on the local area network.

3.4 Assessment

Most of the research cited in Section 2 of this paper used the 1999 KDD Cup dataset [17]. This dataset has been widely criticized for misleading classification accuracy based on the data content [18] [19]. It is our plan to validate and assess our data mining approaches against real, synthetic and publicly available data. The goal in our assessment is to increase the true positive and true negative prediction rate while decreasing the false positive and false negative rates. As a university in the .edu domain, we detect frequent attacks from those who are trying to exploit our domain with the intention to purloin intellectual property or personal information. Therefore we can set up honey pots to collect real data on these attackers for use in our testing and assessment of the system. In addition to real data, we can generate synthetic data to model different scenarios and we can use publicly available data sets such as those described in [20].

3.5 Risks

At the outset of this project, we have defined two risks. The first risk is our system missing a threat, whereas doing nothing would allow the threat to pass. The second risk is the misidentification of benign activity as a threat (a false positive). This second risk can be mitigated by properly defining critical services on a system. This system is not a panacea for all network intrusion ills. For instance, the system is not designed to enforce security policies such as

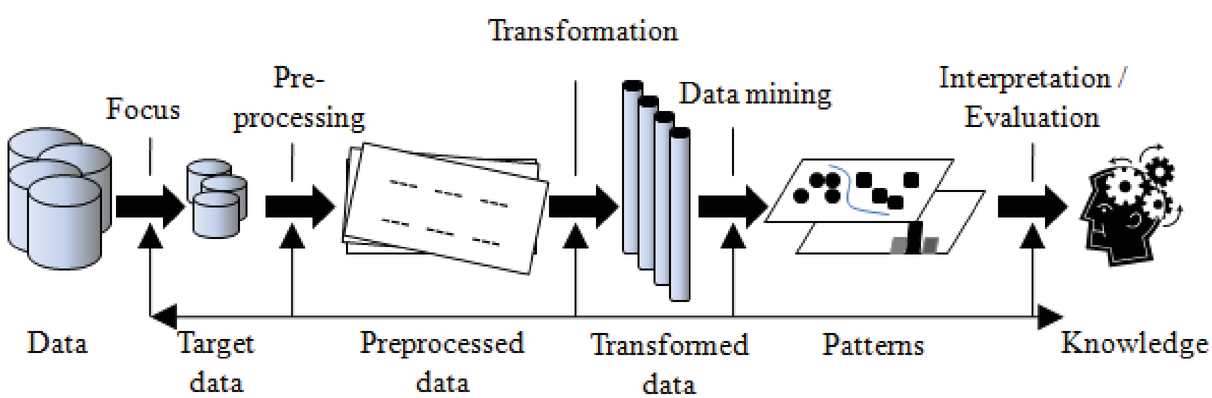


Fig. 2. The Knowledge Discovery Process [14].

strong passwords or detect misconfigured software.

4 Conclusions

The uniqueness of this solution lies in the sum of its parts. Its modular, open-source design allows for lower costs, scalability, and for the system to be tailored to meet an organization's needs. This will allow for the creation of an open alliance to change how threats are mitigated through a trusted and shared knowledge about attacks. In an ideal situation, our planned system would be used by utilities, Internet Service Providers, and the Internet backbone to communicate new threats and take appropriate countermeasures. As we initiate this research, we seek to make improvements across the board and try not to focus on one area.

5 References

[1] M. S. Abadeh, J. Habibi and C. Lucas, "Intrusion Detection Using a Fuzzy Genetics-based Learning Algorithm," *Journal of Network and Computer Applications*, vol. 30, pp. 414-428, 2007.

[2] O. Depren, M. Topallar, E. Anarim and M. K. Ciliz, "An Intelligent Intrusion Detection System for Anomaly and Misuse Detection in Computer Networks," *Expert Systems with Applications*, vol. 29, pp. 713-722, 2005.

[3] S. Peddabachigari, A. Abraham, C. Grosnan and J. Thomas, "Modeling Intrusion Detection System Using Hybrid Intelligent Systems," *Journal of Network and Computer Applications*, vol. 30, pp. 114-132, 2007.

[4] L. Portnoy, E. Eskin and S. Stolfo, "Intrusion detection with unlabeled data using clustering," in *Proceedings of ACM CSS Workshop on Data Mining Applied to Security 2001, DMSA-2001*, Philadelphia, PA, 2001.

[5] N. Jaisankar, R. Saravanan and K. D. Swamy, "Intelligent Intrusion Detection System Framework Using Mobile Agents," *International Journal of Network Security and Its Applications*, vol. 1, no. 2, pp. 72-88, 2009.

[6] McAfee, Inc., "Intrusion Prevention System - Network Security Platform," 2014. [Online]. Available: http://www.mcafee.com/us/products/network-security-platform.aspx. [Accessed 27 Feb 2015].

[7] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Macia-Fernandez and E. Vazquez, "Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges," *Computers & Security,* vol. 28, pp. 18-28, 2009.

[8] wireshark.org, "tshark - The Wireshark Network Analyzer 1.12.2," [Online]. Available: https://www.wireshark.org/docs/man-pages/tshark.html. [Accessed 12 Nov. 2014].

[9] Snort.org, "Snort.org," [Online]. Available: www.snort.org. [Accessed 28 Feb. 2015].

[10] The Bro Project, "The Bro Network Security Monitor," [Online]. Available: www.bro.org. [Accessed 10 Jan. 2015].

[11] Suricata, "Suricata Open Source IDS/IPS/NSM Engine," [Online]. Available: suricata-ids.org. [Accessed 27 Jan. 2015].

[12] Kismet, "Kismet," [Online]. Available: http://www.kismetwireless.net/. [Accessed 15 Jan. 2015].

[13] Fail2ban, "fail2ban," 6 Nov. 2014. [Online]. Available: http://www.fail2ban.org/wiki/index.php/Main_Page. [Accessed 25 Mar. 2015].

[14] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy, *Advances in Knowledge Discovery and Data Mining*, Cambridge: MIT Press, 1996.

[15] R. Kimball, M. Ross, W. Thornthwaite, J. Mundy and B. Becker, *The Data Warehouse Lifecycle Toolkit, 2nd ed.*, Indianapolis: Wiley Publishing, Inc., 2008.

[16] J. Han, M. Kamber and J. Pei, *Data Mining: Concepts and Techniques, 3rd ed.*, Waltham, MA: Morgan Kaufmann, 2011.

[17] KDD Cup 1999, "KDD Cup 1999 Data," 28 Oct. 1999. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. [Accessed 2014 15 Aug].

[18] J. McHugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory," *ACM Transactions on Information and System Security*, vol. 3, no. 4, pp. 262-294, 2000.

[19] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A Detailed Analysis of the KDD Cup 99 Data Set," in *Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications, CISDA 2009*, Ottawa, ON, 2009.

[20] A. Shiravi, H. Shiravi, M. Tavallaee and A. A. Ghorbani, "Toward Developing a Systematic Approach to Generate Benchmark Datasets for Intrusion Detection," *Computers and Security*, vol. 31, no. 3, pp. 357-374, May 2012.

Controlled Information Maximization for SOM Knowledge Induced Learning

Ryotaro Kamimura
IT Education Cente and Graduate School of Science and Technology, Tokai Univerisity
1117 Kitakaname, Hiratsuka, Kanagawa 259-1292, Japan
ryo@keyaki.cc.u-tokai.ac.jp

Abstract— *The present paper aims to control information content in multi-layered neural networks to improve generalization performance. Following Linsker's maximum information principle, information should be increased as much as possible in multi-layered neural networks. However, it is needed to control information increase appropriately to improve the performance. Thus, the present paper proposes a method to control information content so as to increase generalization performance. Experimental results on an artificial data and the spam data set showed that improved generalization performance could be obtained by appropriately controlling information content. In particular, better performance could be observed for complex problems. Compared with the results by the conventional methods such as the support vector machine, better performance could be obtained when the information was larger. Thus,, the present results certainly show a possibility of SOM knowledge in training multi-layered networks.*

Keywords: Maximum information, controlling information, multi-layered networks, SOM, deep learning

1. Introduction

1.1 Maximum Information

Information-theoretic methods have received due attention since Linsker [1], [2], [3], [4] tried to describe information processing in living systems by the maximum information principle. In this principle, information content in multi-layered neural networks should be increased as mush as possible for each processing stage. Linsker demonstrated the generation of feature detecting neurons by maximizing information content for simple and linear neural networks. However, because difficulty have existed in training multi-layered neural networks, few results on this performance of fully multi-layered neural networks have been reported.

Recently, multi-layered neural networks has received much attention because several methods to facilitate the learning of multi-layered neural networks have been proposed in the deep learning [5], [6], [7], [8]. Thus, the time has come to examine the effectiveness of the maximum information principle in training multi-layered neural networks. In the deep learning, unsupervised feature detection is realized by the auto-encoder and the restricted Boltzmann machines. However, they are not necessarily good at detecting main features of input patterns, because they have not been developed as feature detectors. Thus, it is needed to use more efficient feature detecting methods for multi-layered neural networks.

1.2 SOM Knowledge

In training multi-layered neural networks, it is important to extract the main features of input patterns. In the present paper, the self-organizing maps (SOM) is used to detect the features for training multi-layered neural networks. As is well known, the SOM has been developed to extract important features and in addition to visualize those features. If it is possible to use the features detected by the SOM for training multi-layered neural networks, the training can be more facilitated, and in addition, final results can be visualized for easy interpretation.

Recently, the SOM was found to be effective in training multi-layered neural networks under the condition that information content of each hidden layer is maximized or increased as much as possible [9]. This means that Linsker's principle of maximum information preservation is effective in training multi-layered neural networks with the SOM. Meantime, it has been observed that information should not be simply increased. The information increase or maximization should be appropriately controlled to have better performance, in particular, better generalization performance. Thus, the objective of the paper is to control appropriately the process of information maximization and to explore to what extent generalization performance can be improved.

1.3 Outline

In Section 2, the SOM knowledge induced learning is introduced, which is composed of SOM and supervised multi-layered neural networks. Then, the information content is defined as decrease of uncertainty of neurons. This information is controlled by using the number of layers multiplied by the other parameter r. The parameter r is introduced to adjust the information content for given problems. In Section 3, the artificial and spam data are used to examine to what extent information can be increased and generalization performance can be improved. Experimental results show that information can be increased and correspondingly generalization errors can be decreased by the present method.

2. Theory and Computational Methods

2.1 SOM Knowledge Induced Learning

The SOM knowledge induced learning is a method to use the knowledge by the SOM to train multi-layered neural networks. Figure 1 shows a network architecture for the learning. As shown in the figure, the learning is composed of two phases, namely, the information acquisition (a) and use (b) phase. In the information acquisition phase in Figure 1(a), each competitive layer is trained with SOM to produce weights. These weights are used to train multi-layered neural networks in Figure 1(b). In the information use phase, the ordinary back-propagation learning is applied with the early stopping criteria. The problem is whether the weights by the SOM are effective in improving generalization performance.

2.2 Information Content

The SOM knowledge is effective only with the maximum information principle. Thus, this section deals with how to increase information content. As shown in Figure 1, a network is composed of the input layer, multiple competitive layers and an output layer. Let us explain how to compute output from competitive and output neurons. Now, the sth input pattern can be represented by $\mathbf{x}^s = [x_1^s, x_2^s, \cdots, x_L^s]^T,\ s = 1, 2, \cdots, S$. Connection weights into the jth competitive neuron are denoted by $\mathbf{w}_j = [w_{1j}, w_{2j}, \cdots, w_{Lj}]^T,\ j = 1, 2, \ldots, M$. The output from an output neuron is computed by

$$v_j^s = \exp\left(-\frac{\parallel \mathbf{x}^s - \mathbf{w}_j \parallel^2}{\sigma^2}\right), \tag{1}$$

where σ denotes a spread parameter or Gaussian width. The output from the jth neuron is defined by

$$v_j = \frac{1}{S}\sum_{j=1}^{M} v_j^s. \tag{2}$$

The firing probabilities are computed by

$$p(j) = \frac{v_j}{\sum_{m=1}^{M} v_m}. \tag{3}$$

The uncertainty or entropy of this neuron is

$$H = -\sum_{j=1}^{M} p(j)\log p(j). \tag{4}$$

The information content is defined by difference between maximum and observed uncertainty

$$\begin{aligned} I &= H^{max} - H \\ &= \log M + \sum_{j=1}^{M} p(j)\log p(j). \end{aligned} \tag{5}$$

2.3 Controlled Information Maximization

This information can be increased by decreasing the Gaussian width σ. The width is here defined by

$$\sigma(t) = \frac{1}{t^r}, \tag{6}$$

where t denotes the layer number. When the number of layers increases, the spread parameter σ decreases and the corresponding information tends to increase. In addition, the parameter r is needed to control the spread parameter. When the parameter r increases, the spread parameter σ decreases and correspondingly the information tends to increase.

Figure 2 shows the spread parameter σ as a function of the number of layers t when the parameter r increases from 0.1 to 2.5. As shown in the figure, the spread parameter decreases when the the number of layers increases. In addition, the spread parameter decreases when the parameter t increases. When the layer number is higher, the spread parameter gradually decreases and information increases. In this case, the number of strongly firing neurons in black gradually diminishes as shown in Figure 1. This means that the number of effective competitive neurons gradually diminishes and features can be gradually compressed into a smaller number of competitive neurons.

3. Results and Discussion

3.1 Application to Artificial Data

3.1.1 Experimental Outline

To show the effectiveness of the information maximization, an artificial data set was made, which could be divided into two classes as shown in Figure 3(a). The total number of input patterns was 2000. Among them, only 100 patterns were for training ones. Even if the number of training pattern increased, the tendency here reported was observed. The remaining 900 and 1000 patterns were for the validation and testing ones, respectively. The number of input, competitive and output neurons were 2, 25 (5 by 5) and 2, respectively.

Then, to make the problem more complex, the standard deviation of the data increased gradually. When the standard deviation increased from one in Figure 3(a) to five in Figure 3(b), the boundary between two classes became ambiguous and the problem of classification became more difficult.

3.1.2 Weights by SOM

The SOM trys to imitate input patterns as much as possible. This means that connection weights tend to be expanded to include all input patterns. Figures 4(a) and (b) show connection weights in blue and data in green by the self-organizing maps. In Figure 4(a) and (b), weights in blue were expanded to cover all data points in green. This means that the SOM tried to acquire information over connection weights on input patterns as much as possible. The problem is whether these weights are effective in training multi-layered neural networks.

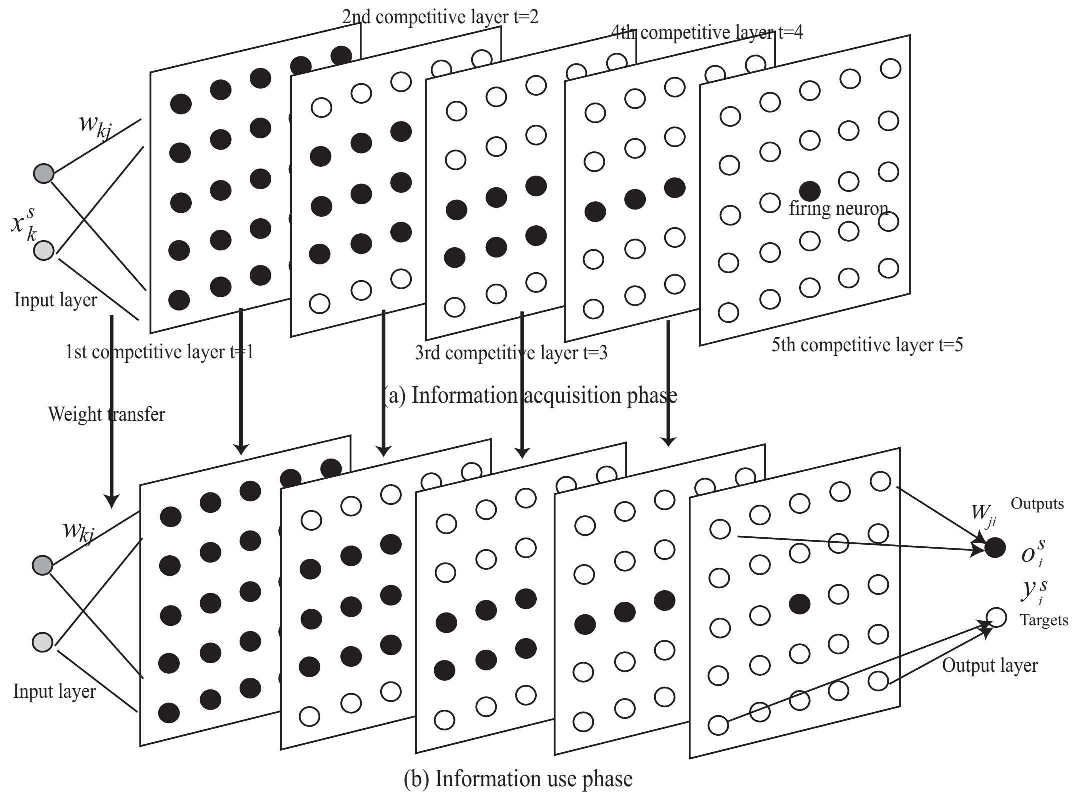


Fig. 1: Network architecture with two components of SOM knowledge induced learning where black neurons fire strongly.

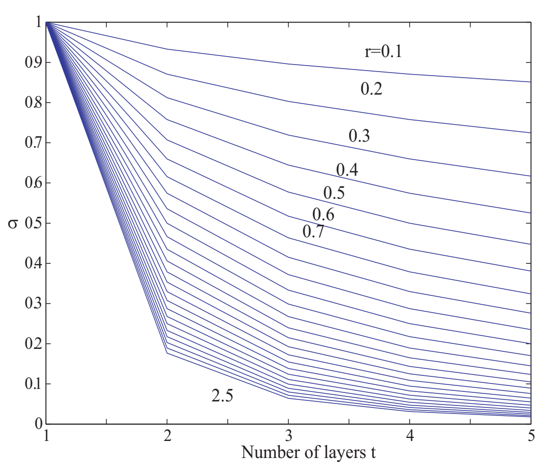


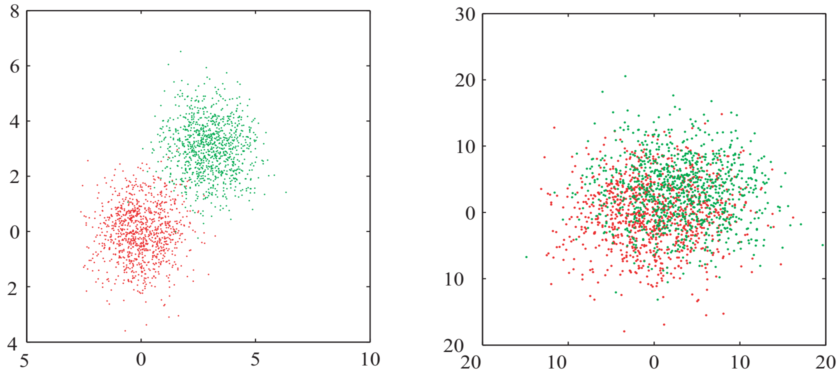
Fig. 2: Spread parameter σ as a function of the number of layers t for different valued of the parameter r.

3.1.3 Results with Information Maximization

Figure 5 shows information and generalization errors when the parameter r increased from 0.5 to 2.5. As shown in Figure 5(a), when the standard deviation was one, information increased when the parameter increased. However, the generalization errors did not decrease and in the end, they increased rapidly. Figure 5(b) shows the results when the standard deviation was three. As shown in Figure 5(b1), information increased gradually. Then, generalization errors decreased gradually as shown in Figure 5(b2). Figure 5(c) shows the results when the standard deviation was five. As shown in Figure 5(c1), information increased and the generalization errors decreased, though some fluctuations could be seen in Figure 5(c2).

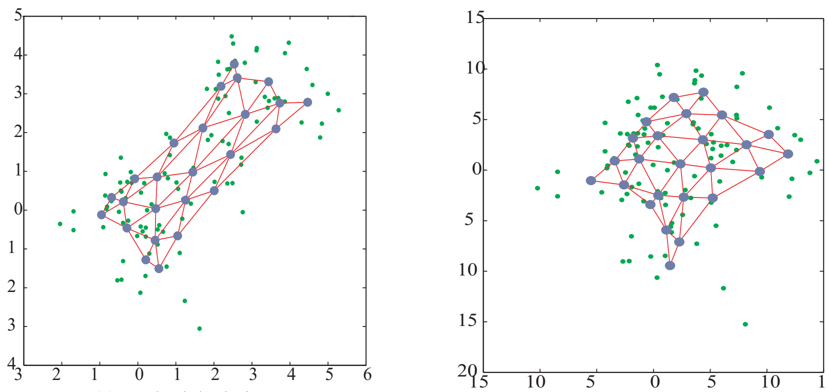
3.1.4 Summary of Results on Generalization

Table I shows the summary of generalization errors. The best average errors in bold faces were obtained by the information maximization. Only when the standard deviation was one, the support vector machine (SVM) showed the performance equivalent to that by the information maximization. When the standard deviation was one, the best error of 0.017 by information maximization was obtained for $r = 2.0$. When the standard deviation was two, the best error was obtained with $r = 1.6$. When the standard deviation was three, the best error was obtained for $r = 2.4$. When the standard deviation was four, the best error was with $r = 2.0$. Finally, when the standard deviation was five, the best error was obtained for $r = 1.7$. Thus, when the parameter r was higher, and correspondingly information was higher, the better performance could be obtained.



(a) Standard deviation=1 (b) Standard deviation=5

Fig. 3: Data with five different values of standard deviation.



(a) Stadard deviation=1 (b) Stadard deviation=5

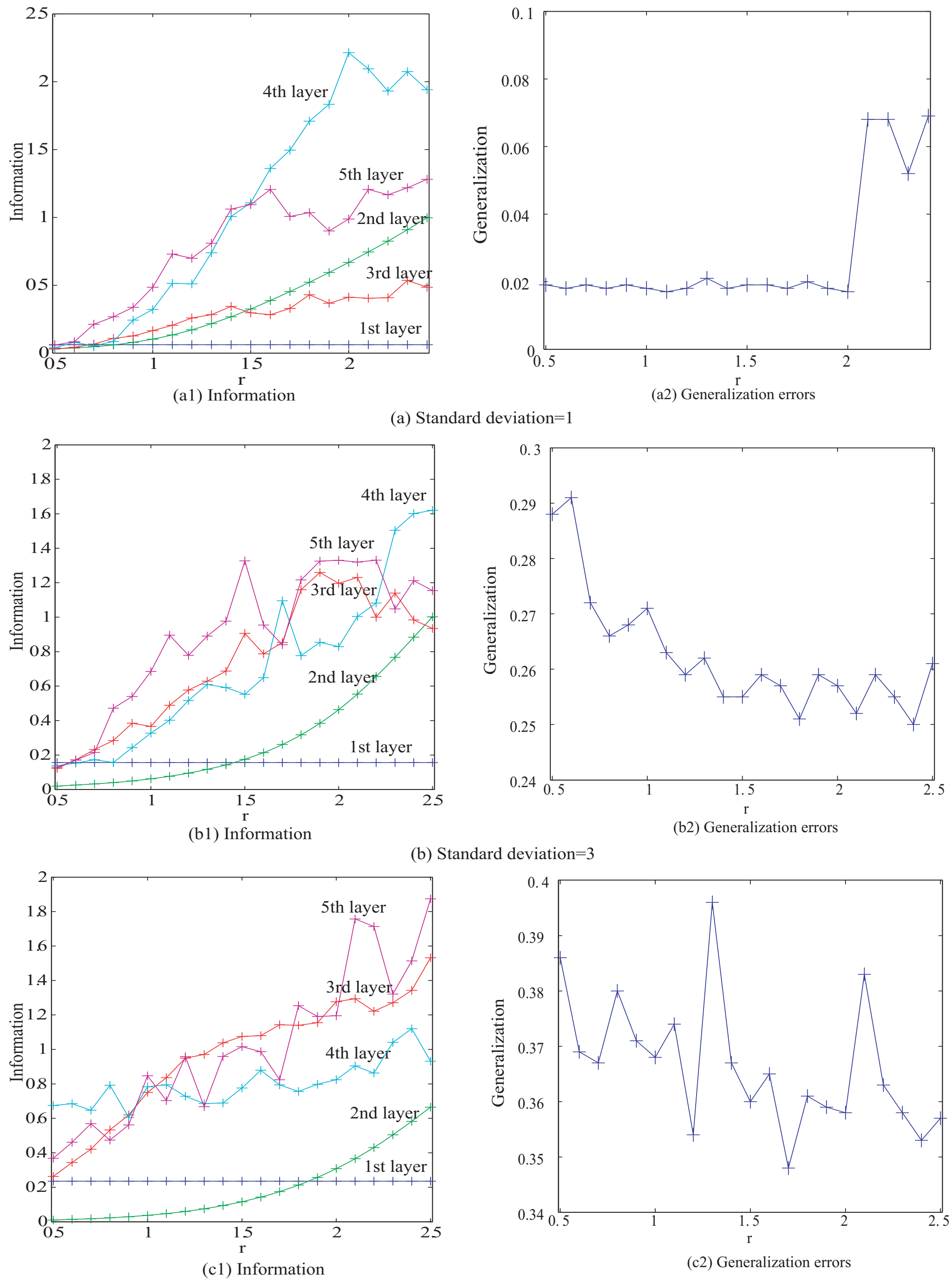
Fig. 4: Weights with 100 epochs by SOM when the standard deviation was one and five.

Table 1: Summary of experimental results for the artificial data, where "Conv", "With" and "Without" represent the conventional multi-layered networks and networks with and without information maximization, respectively. The values of the parameter r denotes networks with the best generalization performance.

			SOM induced learning			
Std dev		Conv	Without	With	r	SVM
1	Avg	0.022	0.145	**0.017**	2.0	**0.017**
	Std dev	0.007	0.166	0.004		0.005
2	Avg	0.172	0.338	**0.153**	1.6	0.160
	Std dev	0.021	0.112	0.010		0.012
3	Avg	0.272	0.432	**0.250**	2.4	0.266
	Std dev	0.025	0.098	0.022		0.017
4	Avg	0.329	0.483	**0.312**	2.0	0.326
	Std dev	0.024	0.090	0.021		0.015
5	Avg	0.365	0.466	**0.348**	1.7	0.370
	Std dev	0.027	0.061	0.017		0.016

3.1.5 Results without Information Maximization

Figure 6 shows information as a function of the number of layers by the method without information maximization. As can be seen in the figure, information tended to increase gradually when the number of layers increased, though the amount of information was smaller than that by the information maximization. The present method is successfully used to increase the information, because this natural tendency of information increase can be accentuated by the present method. However, when the layer number was three, the information decreased in Figure 6. In Figure 5(b), the information increased when the standard deviation was three. Thus, the present method can increase the information in spite of the absence of natural tendency of information increase. In addition, in Figure 5, information increase seems to be correlated with improved generalization when the standard deviation is larger. This means that when the problem becomes more complex, the present method will be more effective in increasing generalization performance.



(a1) Information (a2) Generalization errors

(a) Standard deviation=1

(b1) Information (b2) Generalization errors

(b) Standard deviation=3

(c1) Information (c2) Generalization errors

(c) Standard deviation=5

Fig. 5: Information and generalization errors with the information maximization component when the standard deviation increased from one to five.

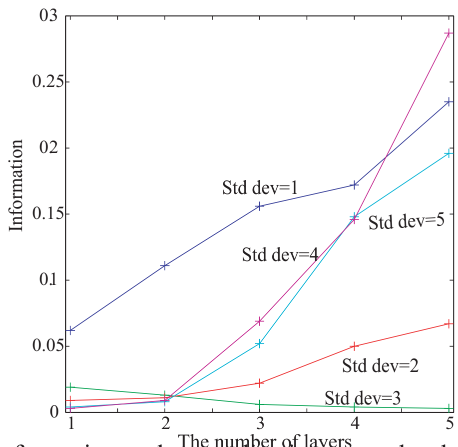


Fig. 6: Information and generalization errors by the method without the information maximization component.

3.2 Application to Spam Data Set

3.2.1 Experimental Outline

The spam data set from the machine learning database [10] was used to examine the performance of the present method. The number of patterns was 4601 with 57 variables and 1000 of them were for training data. The number of validation data set was 1000 and the remaining ones were for testing. The number of input, competitive and output neurons were 57, 25 (5 by 5) and 2, respectively.

3.2.2 Information and Generalization

Figure 7 shows information and generalization errors by the present method. Information content increased gradually when the parameter r increased from 0.5 to 2.0 in Figure 7(a), though in the fourth layer information decreased. Figure 7(b) shows generalization when the parameter r increased from 0.5 to 2.0. The generalization errors decreased gradually and the lowest error was obtained when the parameter r was 1.5. Those results show that when information increased, generalization errors tended to decrease accordingly.

As mentioned, for the fourth layer, the information increased when the parameter r increased from 0.5 to 1.4 in Figure 7(a). However, the information then decreased when the parameter r increased from 1.5 to 2.0 in Figure 7(a). As shown in Figure 7(b), the generalization errors fluctuated when the parameter r increased from 1.5 to 2.0. This fluctuation may be explained by the decrease in information for the fourth layer.

3.2.3 Summary of Generalization Performance

Table II shows the summary of generalization errors. The lowest error of 0.142 was obtained by the present method. The second best error of 0.150 was by the support vector machine. Then, the conventional multi-layered networks shows the third best error of 0.183. The worst error of 0.363 was by the method without the maximum information component.

Table 2: Summary of experimental results for the spam data, where "Conv", "With" and "Without" represent the conventional multi-layered networks and networks with and without information maximization, respectively. The values of the parameter r denotes networks with the best generalization performance.

		SOM induced learning			
	Conv	Without	With	r	SVM
Avg	0.183	0.363	**0.142**	1.5	0.150
Std dev	0.019	0.089	0.035		0.010

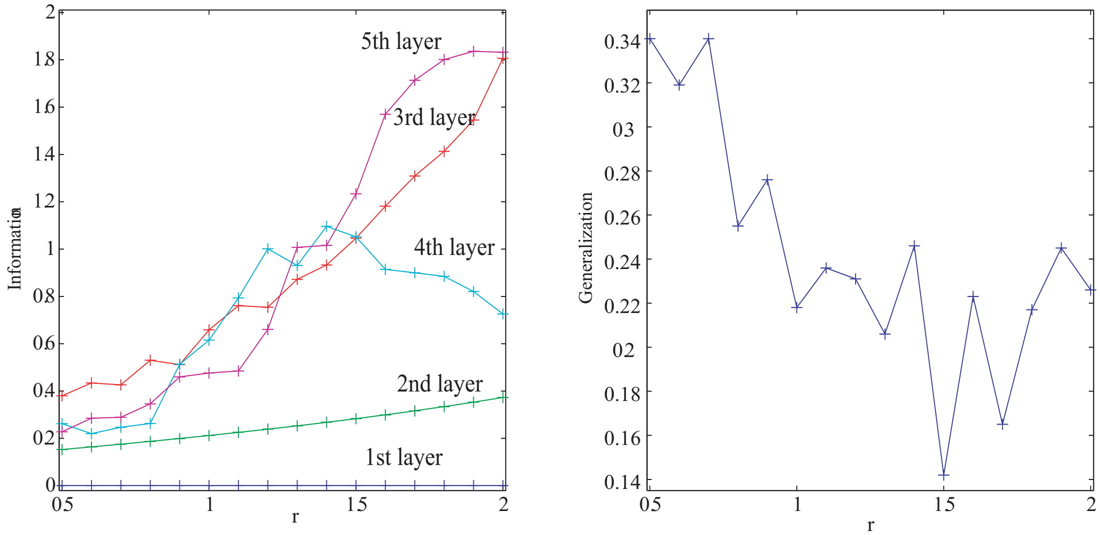
As shown in the table, the largest standard deviation of 0.089 was obtained by the method without the maximum information component. By the maximum information component, the standard deviation decreased from 0.089 to 0.035, which was however the second largest value. Thus, the present method produced results with larger standard deviation and these large values can be decreased by the information maximization component. However, by the present method, the standard deviation was still larger. Thus, it is necessary to examine why such large standard deviation is produced and to develop a method to stabilize the learning by multi-layered neural networks with SOM knowledge.

3.2.4 Comparison of Information Increase

Figure 8 shows the information increase by the method without the maximum information component. The information content increased when the layer number increased from one to three, and then it decreased when the layer number increased from four and five. The information maximization component could increase information in spite of the tendency of information decrease for the higher layers.

3.3 Conclusion

The present paper has shown that it is important to control information content in training multi-layered neural networks. Linsker stated that information content should be maximized for each processing stage. However, simple information maximization does not necessarily imply better performance in multi-layered neural networks. Information content is increased appropriately for each processing stage. Experimental results on the artificial data and spam data set showed that the appropriate control of information increase was essential in increasing better generalization performance. One of the main problems is that the present method sometimes tended to produce the larger variances of results. Thus, it is needed to develop a method to stabilize learning. Though there are some problems to be solved, the present results certainly show that the appropriate control of information content is one of the most important



(a) Information (b) Generalization

Fig. 7: Information and generalization errors by the method with the information maximization component for the spam data set.

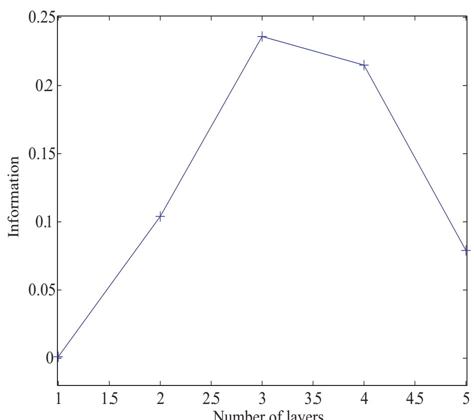


Fig. 8: Information by the method without the information maximization component.

factors in training multi-layered neural networks with SOM knowledge.

References

[1] R. Linsker, "Self-organization in a perceptual network," *Computer*, vol. 21, pp. 105–117, 1988.

[2] R. Linsker, "How to generate ordered maps by maximizing the mutual information between input and output," *Neural Computation*, vol. 1, pp. 402–411, 1989.

[3] R. Linsker, "Local synaptic rules suffice to maximize mutual information in a linear network," *Neural Computation*, vol. 4, pp. 691–702, 1992.

[4] R. Linsker, "Improved local learning rule for information maximization and related applications," *Neural Networks*, vol. 18, pp. 261–265, 2005.

[5] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[6] G. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[7] G. E. Hinton, "Learning multiple layers of representation," *Trends in cognitive sciences*, vol. 11, no. 10, pp. 428–434, 2007.

[8] Y. Bengio, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[9] R. Kamimura and R. Kitajima, "Som knowledge induced learning with maximum information principle to improve multi-layered neural networks," in *Proceedings of computational intelligence conferences*, 2015.

[10] K. Bache and M. Lichman, "UCI machine learning repository," 2013.

The Role of Planning in Object-Oriented Programming for Beginners

Christina Schweikert [1]
[1] Division of Computer Science, Mathematics and Science, St. John's University, Queens, NY USA

Abstract - *Programming languages, environments, and tools have evolved over time and various programming paradigms, including procedural, functional, object oriented, and scripting languages, have been developed. Despite the advancements, programming remains a difficult task for some novices. Learning an object-oriented language, such as C++ or Java, as a first language presents additional challenges for students and instructors. Incorporating the concept of planning into the learning of object-oriented programming may facilitate beginners' understanding of implementing the solution to a problem, as well as the design and implementation of objects. Focus is also placed on properly integrating objects into a problem solution.*

Keywords: object-oriented programming, knowledge representation, planning, computer science education

1 Introduction

Learning object-oriented programming continues to be a challenging task for many students in introductory programming courses. Object-oriented languages, such as C++ or Java, have additional layers of abstraction, due to the use of objects, that may be more easily grasped by some students than others. There has been ongoing discussions within the computer science community about how objects should be presented. [3,4,5] Many textbooks that attempted the "early objects" approach have released "late objects" versions due to demand. In order to understand how to implement classes, one must first understand data types, methods, parameters, return values, among other concepts. A professor could take a mixed approach in which programming is taught starting from the basics, but demonstrates built-in classes early on - for example, from the Java library. It is becoming more critical that, in addition to learning the programming language, students develop strong problem-solving skills. There have been several visual and interactive environments for exposing students to programming, including Scratch [7], Alice [2], and BlueJ [1], to name a few.

2 Plan Knowledge Representation

Utilizing the concept of plans in programming is rooted in the idea that "experts" in a field develop specialized knowledge through their experiences, which could include a set of actions needed to accomplish a goal. For example, a master in the game of chess has built up years of "experiential knowledge" that is drawn upon when a chess master faces a new, but similar, situation to one encountered in the past. It is then that the expert draws upon their experience, or set of plans, to make the moves necessary to achieve the goal – in this example, winning the game. We can represent this experiential knowledge as a plan. The concepts of scripts and plans for knowledge representation originated in the area of natural language processing (NLP). A script is a structure that contains a predetermined sequence of events that applies within a particular context. Plans account for general knowledge that can be used in new situations. Plans contain a set of choices needed to accomplish a goal. For example, when discovering a plan while reading a book, one can make guesses about the intentions of an action in an unfolding story and use these guesses to make sense of the story. [10] Experienced programmers can remember programs better than beginners if the programs have some meaningful structure, or plan. [11] Experts can recognize plans they have become familiar with through experience, such as a program that searches for a value in a list. Concepts from the research area of planning in the field artificial intelligence can be applied when dealing with applications on a larger scope and scale [6, 8]; however, here the focus is strictly on novice programming.

3 Class Design using Plans

When learning to program, beginners often have difficulty in properly designing and implementing classes, and using objects. This includes errors in determining the appropriate attributes and methods for a class, as well as failure to properly utilize objects in a problem solution. To get learners accustomed to class design, we can utilize a "plan" to design a class, along with component plans for creating attributes and methods, as well as inheritance relationships. Plans can also be used to integrate resulting objects into a program. The idea is to help beginners design classes and incorporate objects as part of a larger solution to a problem. Plans, and their components, are then implemented in the chosen object-oriented programming language. As an example, a simple Inventory Plan will necessitate an Item Object. Based on the specification of the Inventory Plan, the proper attributes and methods will be created and integrated into the Item Object. Here, the Item Object would require itemID, itemName, and price attributes, as well as a displayItem() method.

4 Learning with Plans and Objects

A prototype system, Web Plan Object Language (WPOL), which uses the concept of programming plans within an object-oriented paradigm has been designed. [9] Representing programming knowledge, even simple tasks such as computing an average, as plans can help develop students understand how the solution to a problem is translated into code. Most problems are complex and involve multiple plans that need to be integrated together. Plan integration refers to the relationship between plans, such as plans that are sequential (appended), branched, embedded, or interleaved. The system consists of 3 phases of learning: Plan Observation, Integration, and Creation. The phases are described in the following sections.

4.1 Plan Observation

In this phase, solutions to sample programs are visually demonstrated, step by step, using plans to design necessary objects and to design other program tasks. Plans are integrated to form the final problem solution, and then the plans are transitioned into code. The Observation Phase begins with the problem description and identification of the major components. The Student Average Plan is used as an introductory example.

Student Average Plan Description: Compute the average of a student's assignment, midterm, and final scores. A student's id, assignment, midterm, and final scores will be input. The student's id, along with the computed average, will be displayed.

A portion of the Observation Phase of a sample Student Average Plan is illustrated in Fig. 1; this screen shows the implementation of the computeAverage() method of the Student class. The Student Object Plan contains embedded plans for attributes (Data Member Plan), and methods (Member Function Plan). In this case, the language of choice is C++. If using Java, the corresponding terminology would be used. Fig. 2 shows the integration of the Student Object Plan into the Student Average Main Plan and a sample running of the program. "Object Utilities" is included here and consists of a Set Plan and Get Plan, which create set and get methods for the attributes. Constructors (and destructor if needed) would also belong here. (Note: Only a portion of the entire solution process is shown due to space constraints.)

The second example provided is a Sort Students Plan; this program builds on the previous Student Average Plan and extends it by incorporating concepts of inheritance, arrays, loops, decision-making, and sorting. The Student class inherits a newly created Person class, an array of Students is created, Students are sorted by their computed average using Bubble Sort, and the sorted array of objects is displayed. Snapshots from this example are included in Fig. 3, 4, and 5.

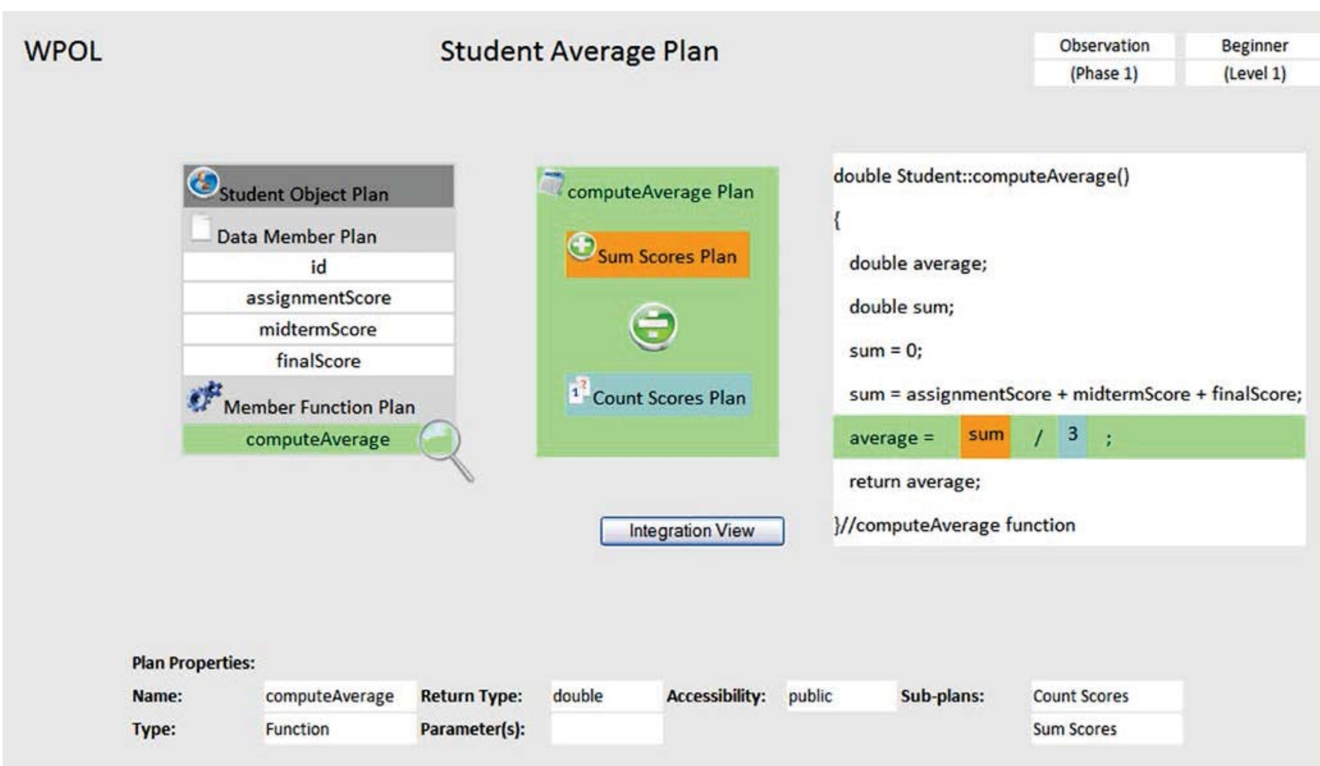


Fig. 1 – Code View for computeAverage Plan

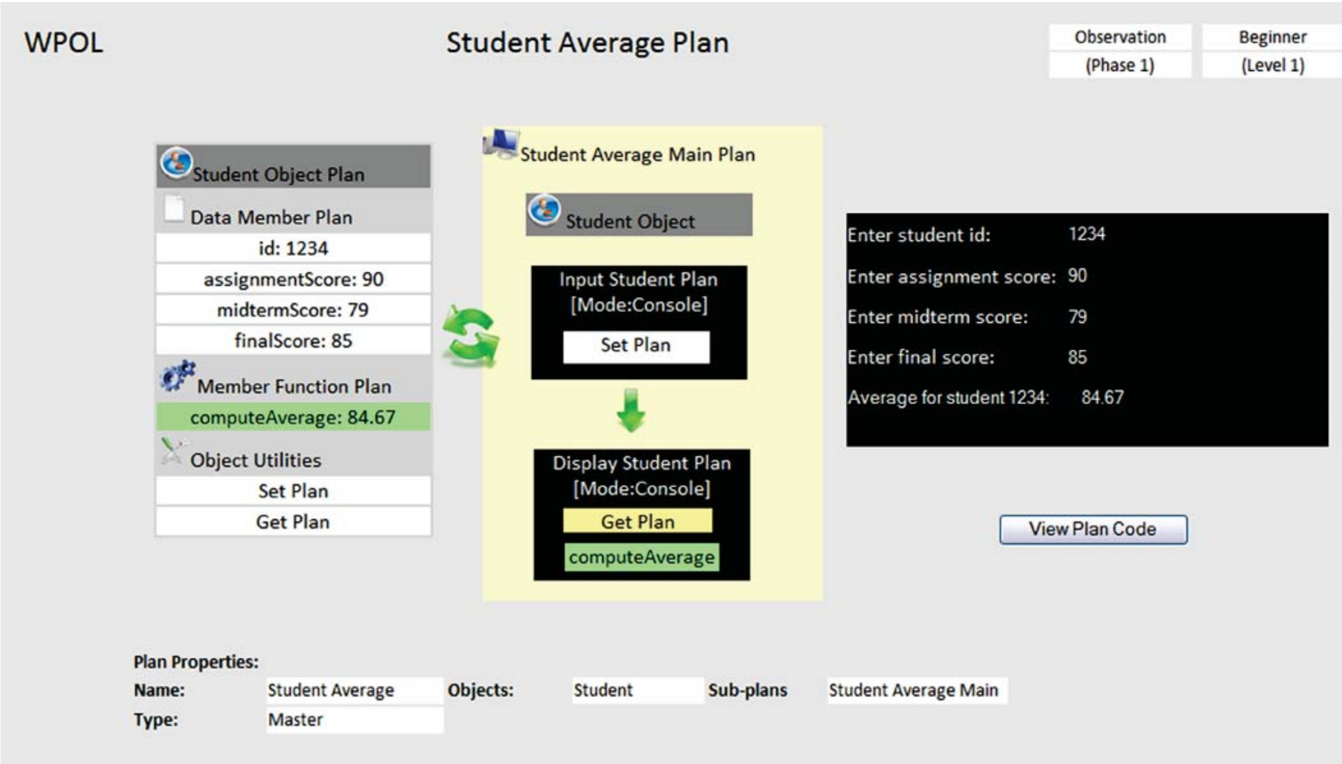


Fig. 2 - Integration View for Student Average Main Plan

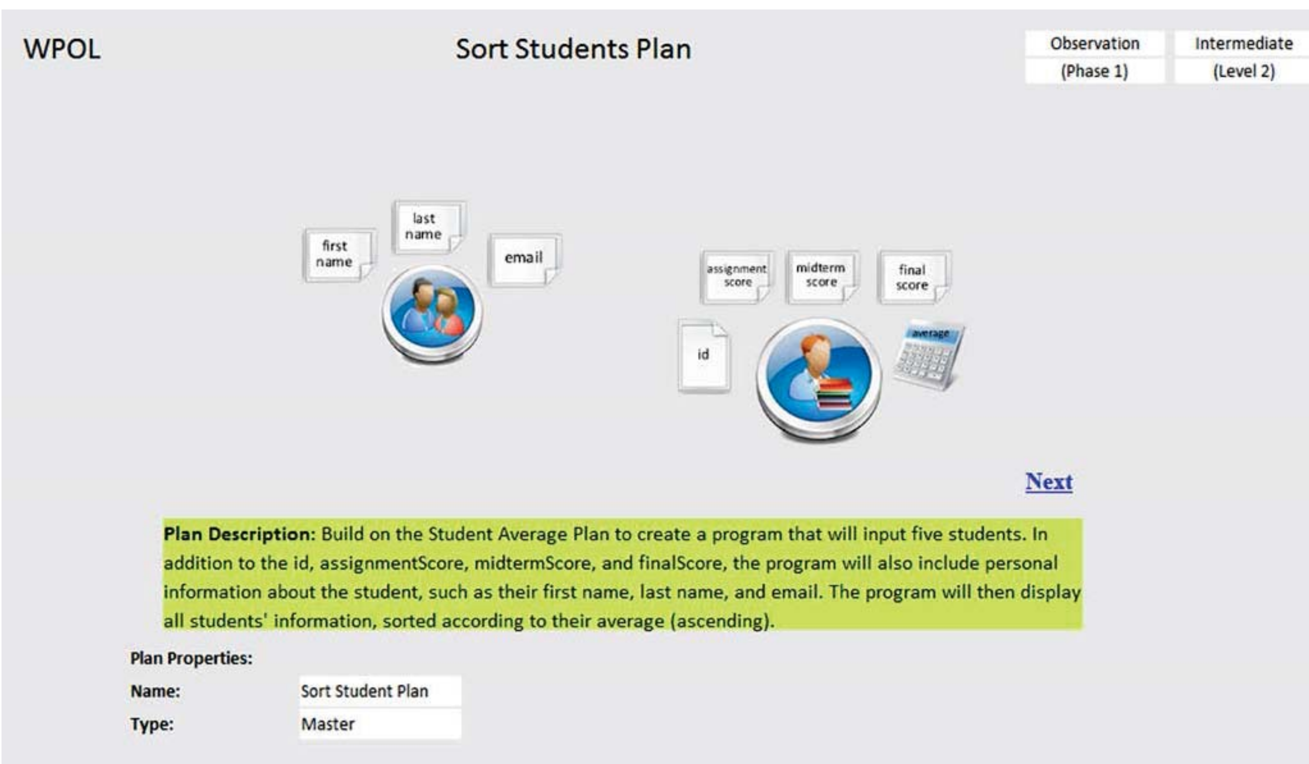


Fig. 3 - Sort Students Plan Description

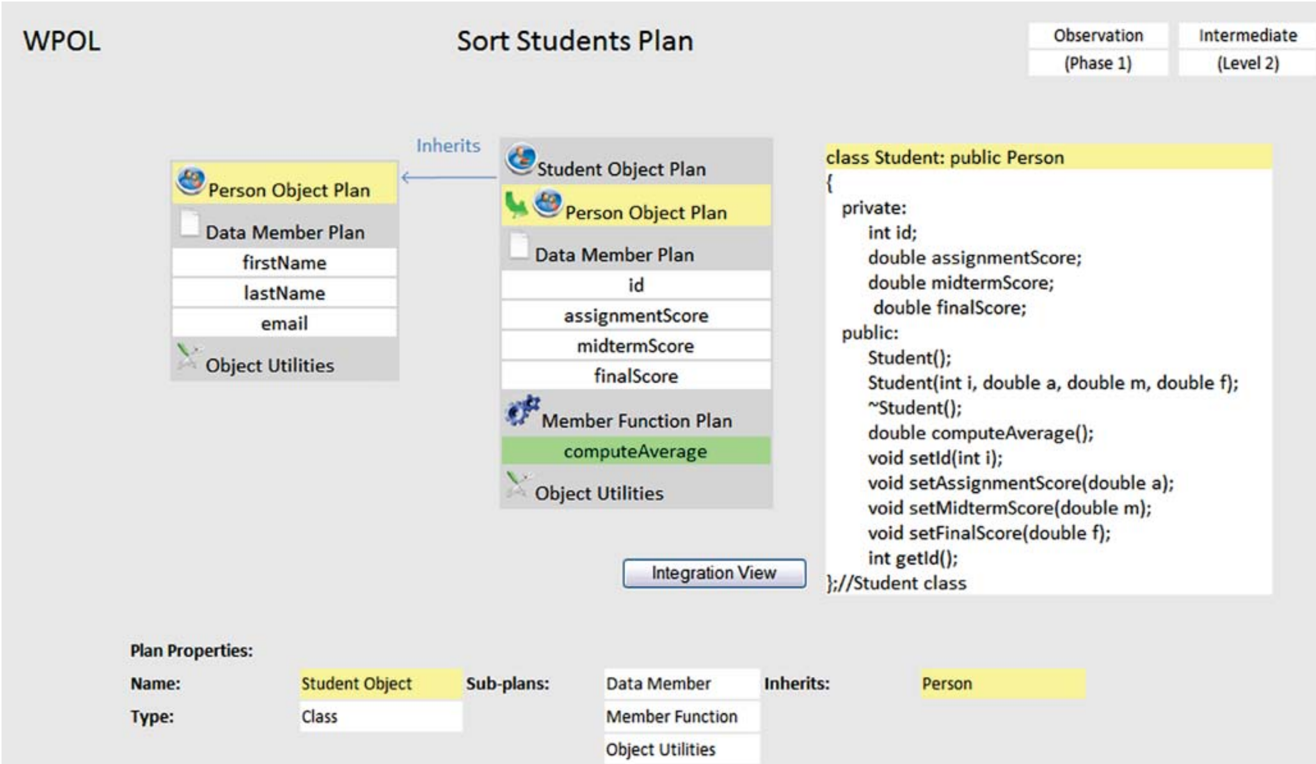


Fig. 4 - Implementation of Inheritance (IS-A) Relationship

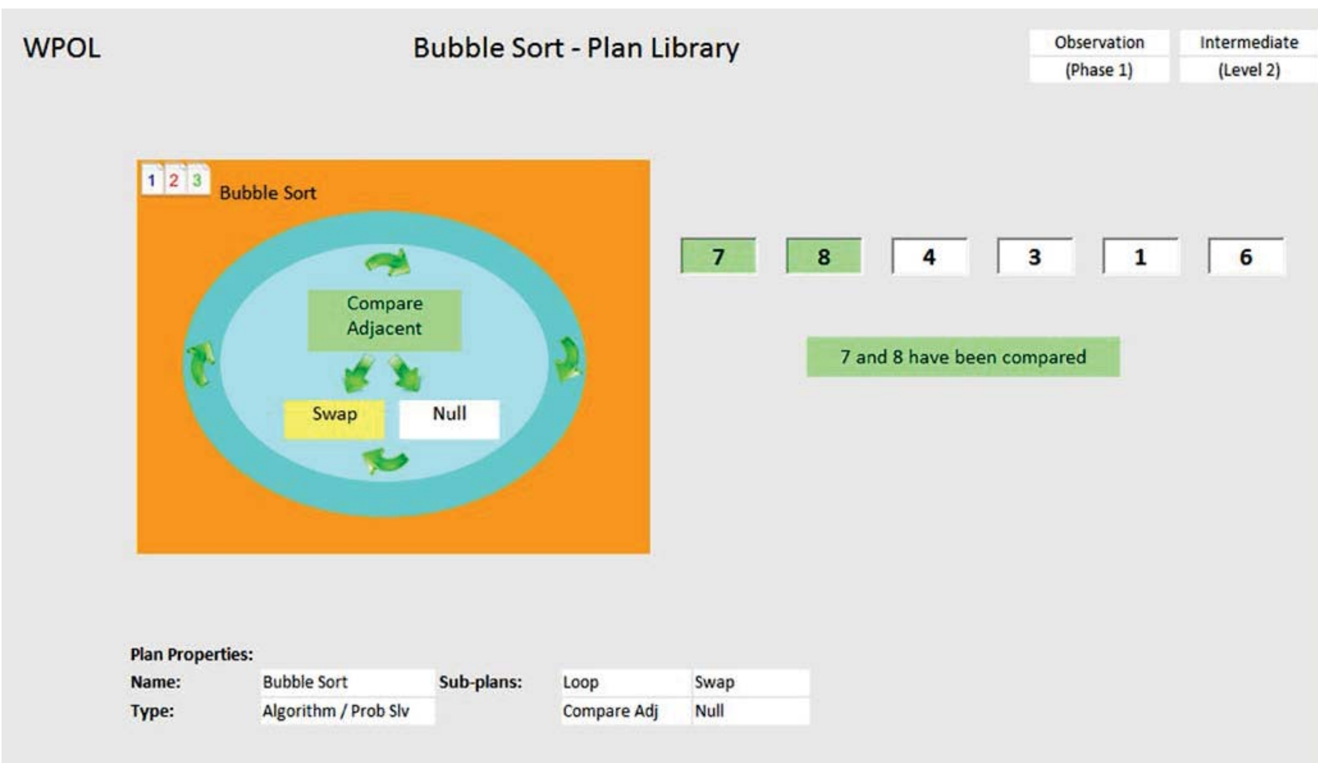


Fig. 5 - Compare Adjacent Plan Demonstration

4.2 Plan Integration

In the Integration Phase, a student's ability to properly integrate plans to form a solution is tested. The purpose of this phase is to reinforce concepts of plan integration and object design. The student is provided with a description of a program to be completed. Then, the student is presented with plans and asked to select which plan(s) should be integrated. The correct integration mode (Appended, Branched, Embedded, or Interleaved) must also be selected. This reinforces the students' understanding since they are taking an active role in creating the solution, and incorrect solutions are explained. A sample incorrect plan integration is shown in Fig. 6.

4.3 Plan Creation

In the Creation Phase, students can customize plans and design new objects (classes). Plans are customized by setting plan properties. This phase facilitates students in creating a program template. For example, an Object can be created by setting the properties of an Object Plan, as well as the properties of its sub-plans. This includes setting class attributes and methods. A screenshot from the creation of an Object Plan for a Book class is demonstrated in Fig 7.

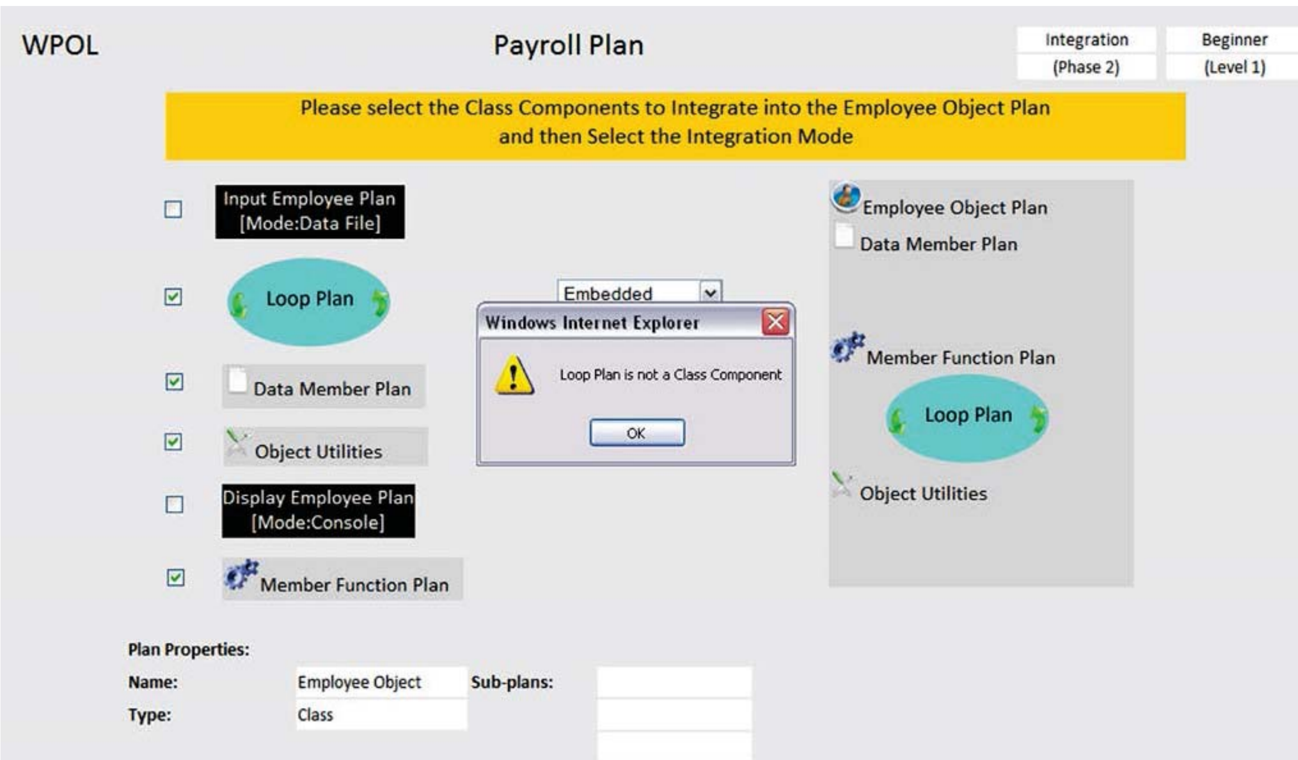


Fig. 6 – Selection of Incorrect Plan

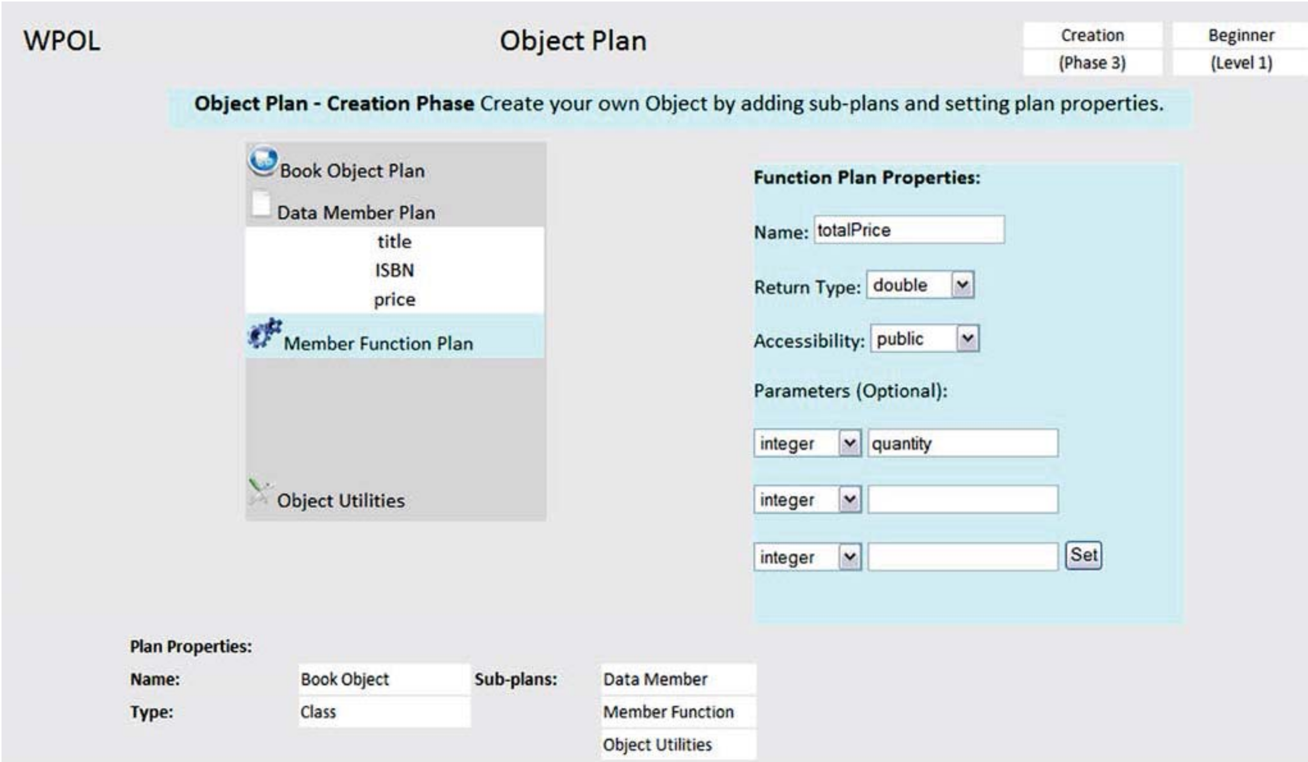


Fig. 7 – Function Plan Properties

5 Conclusions and implications

As computer science educators, we are constantly seeking ways to enhance students' experience of learning programming, and to enable better assimilation of programming concepts, such as object oriented programming. This project seeks to capture the way expert programmers represent programming knowledge and visualize this knowledge representation for novices to enhance their learning of programming in the object-oriented paradigm. Objects can be introduced early with a visual environment and plan representation that reinforces object design and object oriented concepts. A contribution of this research project is using the concept of plans to teach object-oriented programming and problem solving. This approach enhances novice programmers' ability to design, implement, and integrate objects into their programs. Another contribution of this work is a proposed learning environment that utilizes the planning approach with three phases of learning: plan observation, plan integration, and plan creation. The proposed environment is easily adapted to any object-oriented language, such as Java and C#.

6 References

[1] D. J. Barnes & Michael Kölling, Objects First with Java: A Practical Introduction using BlueJ, Fifth edition, Prentice Hall / Pearson Education, 2012.

[2] Conway, M. et al. Alice: Lessons Learning from Building a 3D System for Novices. in Proceedings of CHI (2000). ACM Press, 486-493.

[3] C. Hu, Rethinking of Teaching Objects-First. *Education and Information Technologies* 9, 3 (September 2004), 209-218, 2004.

[4] C. Hu, Just say 'A Class Defines a Data Type'. *Commun. ACM* 51, 3 (March 2008), 19-21, 2008.

[5] A. Ehlert and C. Schulte, Empirical comparison of objects-first and objects-later. In *Proceedings of the fifth international workshop on Computing education research workshop* (ICER '09). ACM, New York, NY, USA, 15-26, 2009.

[6] S. Lucci and D. Kopec, Artificial Intelligence in the 21st Century, 2nd ed. Dulles, Virginia: Mercury Learning & Information, 2015.

[7] J. Maloney, M. Resnick, N. Rusk, B. Silverman, & E. Eastmond. The Scratch Programming Language and Environment. Trans. Comput. Educ. 10, 4, Article 16 (November 2010), 15 pages.

[8] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed. Upper Saddle River, New Jersey: Prentice Hall, 2009.

[9] C. Schweikert, Study of novice programming: Plans, object design, and the Web Plan Object Language (WPOL). Ph.D. dissertation, The Graduate Center, City University of New York, 2008.

[10] R. Schank and R. Abelson, Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures. Hillsdale, NJ: Erlbaum, 1977.

[11] E. Soloway, K. Ehrlich, J. Bonar, Tapping into Tacit Programming Knowledge. Proceedings of the Conference on Human Factors in Computing Systems, NBS, Gaithersburg, Md, 1982.

Epistemology Under Schematization:

Defining Knowledge

Sabah Al-Fedaghi
Computer Engineering Department, Kuwait University, Kuwait

***Abstract*—Many scientific fields have used diagrams to depict knowledge and to assist in understanding problems. This paper proposes to apply a schematization method in epistemology. The method is used in three examples that challenge the definition of propositional knowledge as justified true belief. The resultant drawings seem to contribute to increasing the comprehension of this issue. Initially, such a description can be used as a flowchart that facilitates discussion.**

Keywords—knowledge; justified true belief; diagrams; diagrammatic representation; Gettier problem

I. Introduction

Diagrams probably rank among the oldest forms of human communication. Natural sciences, physical sciences, and mathematics, as well as applied sciences such as engineering, have used diagrams to depict knowledge and to assist in understanding problems [1-4]. Traditional logic diagrams have been utilized as conceptual representations [5-6], and it has been claimed that these diagrammatic representations, in general, have advantages over linguistic ones [7-9].

> Diagrammatic inscriptions, …, are media that provide a point of linkage between thinking and intuiting, ... Today, images are recognized as a legitimized object of research in epistemology and philosophy of science. They are considered not merely a means to illustrate and popularize knowledge but rather a genuine component of the discovery, analysis and justification of scientific knowledge. [10]

Several well-known diagrammatic systems have been available as a heuristic tool. Euler circles [11] embrace circles to illustrate syllogistic reasoning. A drawback of these diagrams is a failure to represent certain pieces of information in a single diagram [5]. Venn diagrams [12] developed to produce an expressive power so that partial information could be represented. "The solution was [the] idea of 'primary diagrams'. A primary diagram represents all the possible set-theoretic relations between a number of sets, without making any existential commitments about them [5]. Peirce [13] introduced his diagrams to extend Venn's system in expressive power with respect to the existential and disjunctive statements [5].

Recently, many researchers have become increasingly aware of the importance of diagrammatic representation systems.

> Diagrams are usually adopted as a heuristic tool in exploring a proof, but not as part of a proof. It is a quite recent movement among philosophers, logicians, cognitive scientists and computer scientists to focus on different types of representation systems, and much research has been focused on diagrammatic representation systems in particular. [5]

In philosophy, images and diagrams are old subjects. Plato's allegory of the cave visualizes situations and pictures knowledge configurations. "The diagram functions as an instrument of making evident the structure of ontology and epistemology" [10]. Descartes made "two-dimensional geometric figures and linear algebraic equations mutually transferable" [10].

Nevertheless, the current state of diagrammatic representation of logical and philosophical problems is susceptible to propose new methodologies. Specifically, this paper explores the diagrammatic representation, called the Flowthing Model (FM), that has been utilized in representing fundamental structure logic [14-16]. This paper applies FM to some epistemic problems to provide models, where a model is to an abstract schematization that expresses ordered sequence flows and structured events in the problem. Schematization, here, refers to a diagrammatic description of events, actions, and operations that assists in conveying the embedded structure of the involved situation.

Advantages of the resultant diagrams for epistemic problems include providing an explicit depiction represented only implicitly in these problems, conveying a better understanding of the problems, especially in the environment of human/computer interactions (e.g., learning), and presenting some new variations of considering these problems and how to reflect about them.

For the sake of completeness, and because that FM is not a well-known methodology, the model will be briefly described in the next section. The example in the section is a new contribution.

II. Flowthing Model

A flow model is a uniform method for representing things that "flow," i.e., things that are created, processed, released, transferred, and received [17-18]. "Things that flow" include information, materials (e.g., goods), and money. They flow in *spheres*, i.e., their environments. A sphere is different from a set in the sense that a set is a static structure, whereas a sphere includes flowthings (current members) at different stages in a progression and possible directions (lines) of movement from one stage to another, or movement from/to the spheres of the flowthings. A sphere may have subspheres.

An FM representation is a depiction of the structure of a scheme resembling a road map of components and conceptual flow. A *component* comprises *spheres* (e.g., those of a company, a robot, a human, an assembly line, a station) that enclose or intersect with other spheres (e.g., the sphere of a house contains rooms which in turn include walls, ceilings). Or, a sphere embeds flows (called *flowsystems*; e.g., walls encompass pipes of water flow and wires of electrical flow).

Things that flow in a flowsystem are referred to as *flowthings*. The life cycle of a flowthing is defined in terms of six mutually exclusive *stages*: creation, process, arrival, acceptance, release, and transfer.

Fig. 1 shows a flowsystem with its stages, where it is assumed that no released flowthing flows back to previous stages. The reflexive arrow in the figure indicates flow to the Transfer stage of another flowsystem. For simplicity's sake, the stages Arrive and Accept can be combined and termed *Receive*.

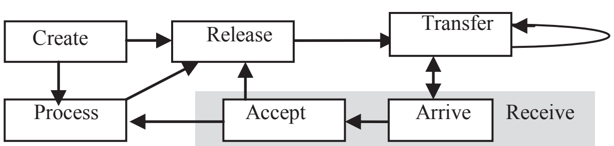


Fig. 1. Flowsystem

The *stages* of the life cycle of a flowthing are mutually exclusive (i.e., a flowthing can be in one and only one stage at a time). All other states or conditions of flowthings are not generic stages. For example, we can have *stored* created flowthings, *stored* processed flowthings, *stored* received flowthings, etc.; thus, *stored* is not a generic stage. In contrast, there are no such stages as, e.g., *created-processed, received-transferred*, or *processed-received* stages. Flowthings can be released but not transferred (e.g., the channel is down), or arrived but not accepted (wrong destination), …

In addition to flows, *triggering* is a transformation (denoted by a dashed arrow) from one flow to another, e.g., a flow of electricity triggers a flow of air.

Example: A proposition p indicates a *sphere* with two subsystems (Body (symbolic expression), and Truth value). Let p be the statement *Maria learns discrete mathematics* and q the statement *Maria will find a good job*. Consider expressing the statement p → q in FM (this example is based on a problem in [19]).

Fig. 2 shows the FM representation for p → q. Each formula p, q, and p → q (circles 1, 2, and 3, respectively) is represented by two flowsystems, body and truth value. If truth values are assigned (created) for p and q (4 and 5, respectively), then they flow to p → q (6 and 7, respectively). There, they are processed according to the truth table of p → q (8) to create a truth value (9).

Application of this method in logical proofs can be found in [14-16]. Note that the description in the body of the proposition can also be represented in FM See Fig 3).

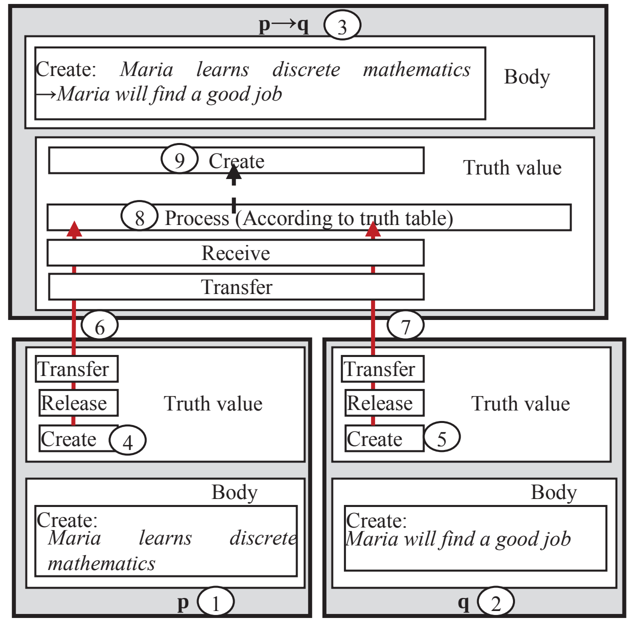


Fig. 2. FM representation of p → q

Fig. 3. FM representation of *Maria learns discrete* mathematics

III. Applying FM to Epistemology

The so-called propositional knowledge refers to what is expressed by a proposition (roughly) sentence. In epistemology, *knowledge* is usually said to entail *true belief*; i.e., knowing a proposition, p, involves believing p, and p is true. Traditionally, to resolve what knowledge *is* (not just entails), a good *justification* is required. Accordingly, knowledge is said to be a justified true belief. There are three components to the traditional ("tripartite") analysis of knowledge. According to this analysis, justified, true belief is necessary and sufficient for knowledge. In the so-called *tripartite analysis of knowledge*, an agent S *knows* that proposition p iff

- S believes that p;
- p is true;
- S is justified in believing that p.

A. Example 1

Consider the following example given by Pritchard [20]. Suppose that an agent called Edmund believes that it is 8:20 a.m. by looking at the clock in his hall. This clock has been very reliable in the past, and Edmund has grounds for believing that the time is 8.20 am (e.g., it's light outside). "Finally, let us stipulate that Edmund's belief is true, it is 8.20 am" [20]. Accordingly, Edmund has a true belief in this proposition, and he has excellent justification.

Suppose that the clock stopped working twenty-four hours previously and is stuck at the time 8.20 am. "The moral of the story is thus that whatever knowledge is, it is not justified true belief" [20].

Fig. 4 shows the FM schematization of this example. It has two time spheres (periods/frames):

- Time sphere 1, called *Twenty-four-hours-ago* (24h, for short, circle 1).
- Time sphere 2: called *Now* (circle 2)

Time sphere 1: Twenty-four-hours-ago (circle 1)

Twenty-four-hours-ago includes one sub-sphere: Clock. The clock receives time (3) that flows from the outside and is processed (4). Process, here, refers to "consuming" the time and triggering (5) the generation (creation) of its clock display (6). As the processing (4) indicates, the process stops at 8: 20 a.m., triggering the display "8:20." The displayed time flows to the *Now* sphere (7).

Time sphere 2: Now (circle 2)

Now includes two sub-spheres: *Edmund* and *(objective) Time* (9). *Edmund* contains three sub-spheres: *Clock display* (10), *Belief* (11) and *Perception* (used for justification, circle 12).

Accordingly, when *Edmund* processes (comprehends) the clock display (13) and is aided by perception (e.g., it's light outside), these trigger the creation of the proposition *It's 8:20* and its truth value (15). In reality, the time is also 8:20. *Reality*, here, means the total sphere that includes all spheres.

Because Edmond (i) believes that *it is 8:20*, (ii) the proposition *"It's 8:20" is true,* and (iii) there is reasonable justification for such a belief; then, does Edmond have *knowledge* about what time it is?

Analysis

Diagrammatic modeling techniques are hardly used in philosophical interpretation because such an act is inclined to be text based. Nevertheless, FM modelling form allows for analysis and hypothesis about the solution to the problem. Accordingly, in this sub-section, we present our interpretation of the cause of the problem involved: why Edmund does not have knowledge even though he seems to have a justified true belief. This is an attempt to demonstrate the capacity of FM to produce new understanding, not only to represent the problem in a graphical form. Of course, this analysis is open for refutation.

From the diagram, we can observe that there are two ways to originate that the proposition *"It's 8:20" is true*:

- Either, as shown in Fig. 4, where the proposition and its truth-value are triggered by the time on the clock (13) and perception (12)
- Or, as shown in Fig. 5, where the proposition and its truth-value is triggered by actual time in reality (e.g., seeing another clock, hearing the time on the radio, etc.).

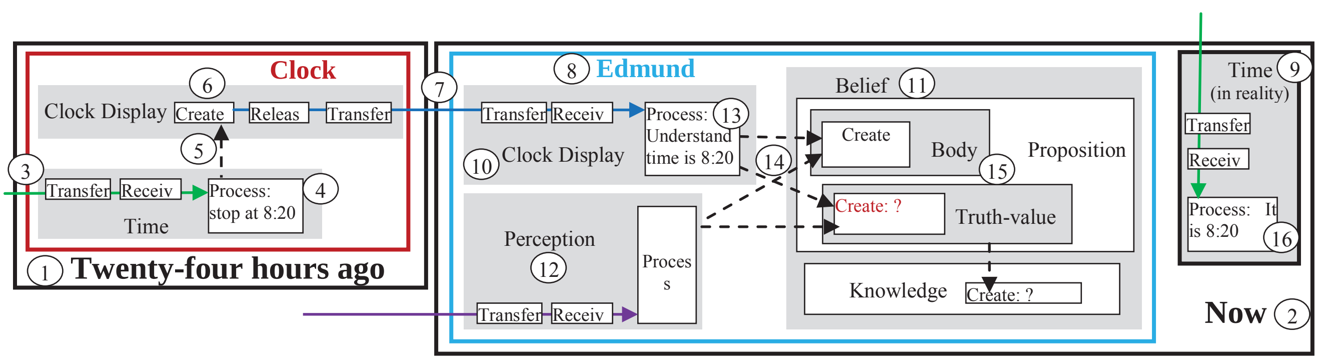


Fig. 4. In the *tripartite* account of knowledge (Triangles) the truthiness of the proposition is triggered by Clock and perception (justification)

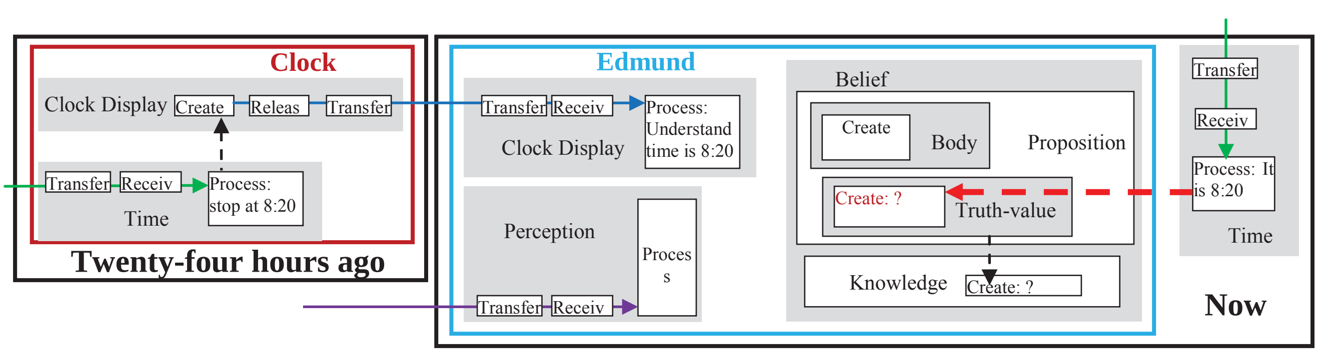


Fig. 5. In the *tripartite* account of knowledge (Triangles) the truthiness of the proposition is triggered by real time

If *"It's 8:20" is true* is based on the stopped clock (13) in Fig. 4) and perception (12 in Fig. 4), then clearly, *It's 8:20* is the product of wrong data. Thus, Edmund has no knowledge.

If *"It's 8:20"* is based on actual time in reality (9 in Fig. 4), then this is not possible because, according to the given account, he never looked at another source of timing.

We reach the conclusion that Edmond does not have *knowledge* about the time now.

Accordingly, to handle such an example, in the knowledge definition, we require that the truth of the proposition is based on the *objective* truth. Objective, here, refers to that the truth-value of the proposition in the *tripartite* account of knowledge (Fig. 6) should be triggered by "reality." In the example, even though the *truth value* that is assigned to *"It's 8:20"* in Edmund's mind coincides with the objective truth value, it has not been triggered by real time.

The moral of such analysis is that it is not enough that the truth value in *tripartite* account of knowledge is true, but also the method (triggering) of such an assignment of value should be originated in reality.

Accordingly, the FM schematization of this problem has provided the following:

- A non-verbose description of the involved situation that can be used in discussion and teaching.
- A possible tool for analysis

B. Example 2

Consider the following example given by Gettier [21]. Smith and Jones have applied for a certain job, and Smith has evidence that (d) *Jones is the man who will get the job, and Jones has ten coins in his pocket*. The president of the company assured him that Jones would be selected. Also, Smith counted the coins in Jones's pocket ten minutes ago. Proposition (d) entails (e) *The man who will get the job has ten coins in his pocket*. In this case, Smith is justified in believing that (e) is true. But suppose that unknown to Smith, he himself, not Jones, will get the job. And unknown to Smith, he himself has ten coins in his pocket. In this example, then, all of the following are true:

(i) (e) is true,
(ii) Smith believes that (e) is true, and
(iii) Smith is justified in believing that (e) is true. Gettier [21].

Fig. 7 shows the corresponding FM representation.

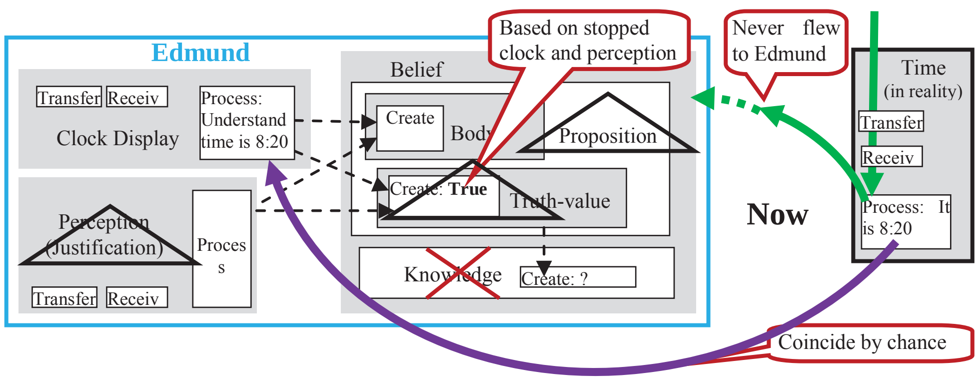


Fig. 6. In the *tripartite* account of knowledge (Triangles) the truthiness of the proposition is questionable

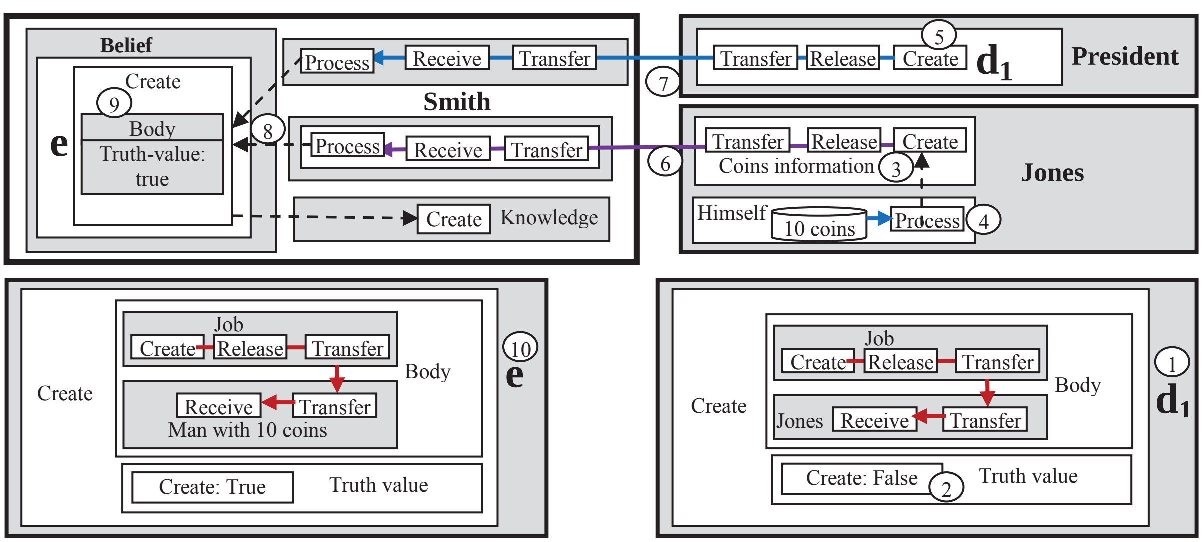


Fig. 7. The truth-value of proposition e is based on d1 from the president and coin information (circle 8).

In the figure, (d) *Jones is the man who will get the job, and Jones has ten coins in his pocket* is divided into two portions.

(1) d1: *Jones is the man who will get the job.*
It is shown (circle 1) as a sphere in the figure, which, in reality, is false (2). Note that the body in (1) is just a *description* of the proposition.

(2) d2*: Jones has ten coins in his pocket*.
This fact is produced as a piece of information (circle 3) by actual processing of Jones (physically counting the coins in his pocket — 4).

Accordingly, the coin information (3) d1, which is created by the president (5) flow to Smith (6 and 7, respectively, to be processed and triggers (8) the creation of proposition e (9) as a true belief. The description and the truth value in reality are shown in the e sphere (10). This sphere expresses that in reality there appears a proposition (sentence) *The man who will get the job has ten coins in his pocket,* and it is true.

We can see that the truth value of the proposition e in the tripartite account of knowledge can be based on the following:

- d1 from the president and coin information (circle 8 in Fig. 7), where d1 is false.
- The reality truth value of e (see Fig. 8)

Because Smith never accessed the truth value of e, he does not have knowledge.

C. Example 3

Consider the following example given, also, by Gettier [21].

Smith … has a justified belief that "Jones owns a Ford." Smith, therefore (justifiably), concludes (by the rule of disjunction introduction) that "Jones owns a Ford, or Brown is in Barcelona," even though Smith has no knowledge whatsoever about the location of Brown.

In fact, Jones does not own a Ford, but by sheer coincidence, Brown really is in Barcelona. (This description of Gettier [21]'s example is taken from Wikipedia [22].)

Fig. 9 shows the corresponding FM diagram. It includes the spheres of Smith, the two propositions (sentences/information) *Jones owns a Ford* and *Brown is in Barcelona,* and the OR logical operation. Smith includes a *Perception* sub-sphere (for justification, e.g., Jones has at all times in the past within Smith's memory owned a car — 5), and *beliefs* (6) regarding the two propositions and their OR.

In the *Jones owns a Ford* sphere (2), *create* means bringing into existence in the setting of situation under consideration. It is similar to a film script that lists the characters and their roles. This *Jones owns a Ford* is involved in the OR operation, thus, it flows to the sphere of OR to be processed. These spheres exist "outside" the sphere of Smith in "reality."

In Smith's belief sphere (6), *Jones owns a Fo*rd is assigned true (8) based on the perceived justification (5). The Or-ing of the two proposition is assigned *true* (9); however, this assignment is based on the triggering (11) and (12) in Smith's belief sphere, not on the truth value of OR in reality (12). Accordingly, it is not Knowledge.

IV. Conclusions

This paper proposes to use a flow-based representation as a base to facilitate understanding in epistemology. The paper demonstrates the viability of the proposed methodology by applying it in three examples that challenge the definition of propositional knowledge as justified true belief. The resultant representation of these three problems provides diagrammatic descriptions that can be used in teaching in artificial intelligence where students are not oriented toward philosophical explanations.

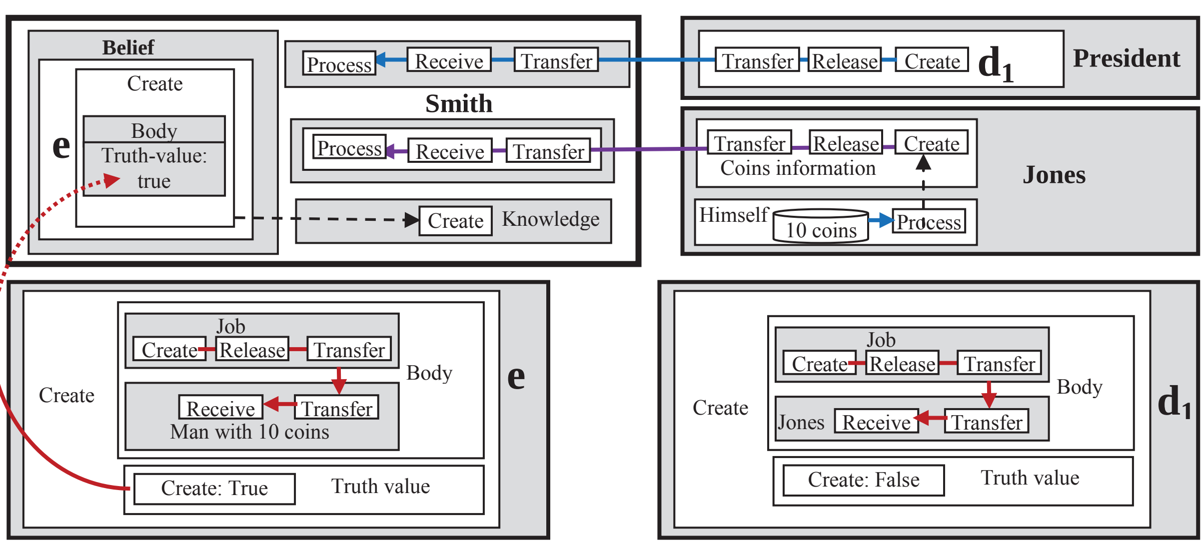


Fig. 8. The "reality" truth-value of proposition e never reaches Smith.

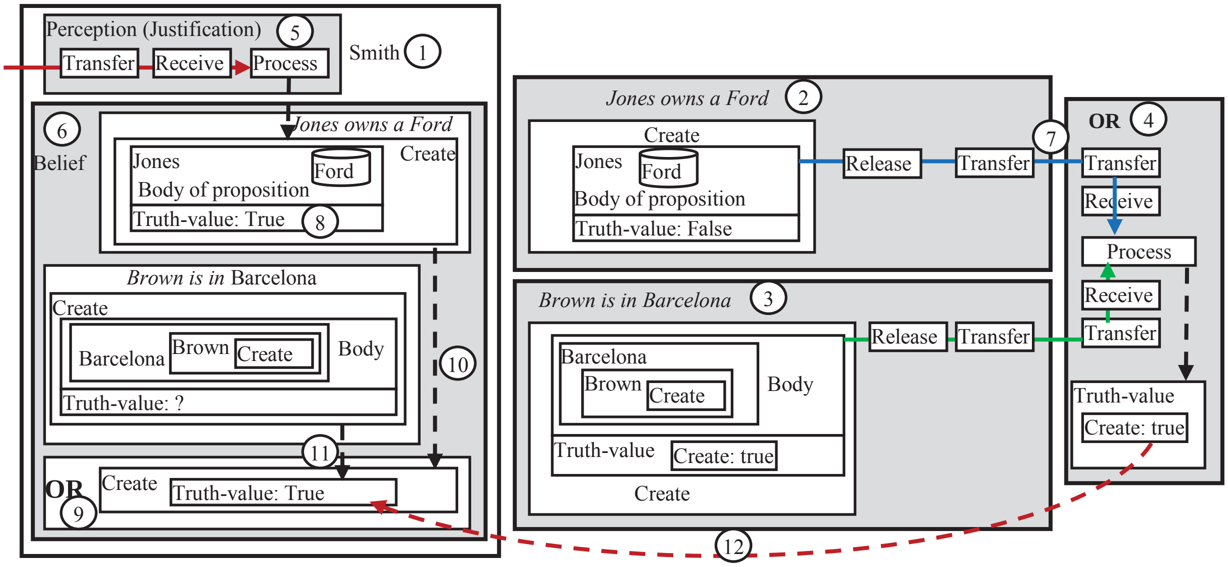


Fig. 9. FM representation of example 3

There is also, an attempt, in the paper to analyze the involved epistemic problems in terms of the structure of the diagram.

The study is exploratory in the sense that it aims at introducing the idea and demonstrating its feasibility. Future research can further develop the approach of FM representation in different epistemic problems.

V. References

[1] R. Arnheim, Visual Thinking, University of California Press, Berkeley, 1980.

[2] A. Barr, and E. A. Feigenbaum, The Handbook of Artificial Intelligence, vol. 1, William Kaufmann, Los Altos, CA, 1981, pp. 200–206.

[3] A. Sloman, "Interactions between philosophy and A.I.: the role of intuition and non-logical reasoning in intelligence," Artificial Intelligence, vol. 2, 1971, pp. 209–225.

[4] A. Sloman, "Afterthoughts on analogical representations," in Proceeding of the 1st Workshop on Theoretical Issues in Natural Language Processing (TINLAP-1), Cambridge, MA, 1975, pp. 164–168.

[5] S–J. Shin, O. Lemon, and J. Mumma, "Diagrams," The Stanford Encyclopedia of Philosophy (Winter 2014 Edition), Edward N. Zalta (ed.). http://plato.stanford.edu/archives/win2014/entries/diagrams/.

[6] G. Allwein, & J. Barwise (Eds.), Logical Reasoning with Diagrams. New York: Oxford University Press, 1996.

[7] A. Shimojima, "The graphic linguistic distinction," Artif. Intell. Rev., vol. 13, 2001, pp. 313–335.

[8] K. Stenning, "Distinctions with differences: comparing criteria for distinguishing diagrammatic from sentential systems," in Diagrams 2000, M. Anderson, P. Cheng, and V. Haarslev, Eds. LNCS (LNAI), vol. 1889, 2000, pp. 132–148.

[9] "Diagrams," in Stanford Encyclopedia of Philosophy, 2012. www.science.uva.nl/~seop/archives/spr2012/entries/diagrams

[10] S. Krämer, "Epistemology of the line. Reflections on the diagrammatical mind," in Studies in Diagrammatology and Diagram Praxis, Alexander Gerner and Olga Pombo (Eds.), , London: College Publications, 2010, pp. 13–38.

[11] L. Euler, Lettres à une Princesse d'Allemagne, St. Petersburg; l'Academie Imperiale des Sciences, 1768.

[12] J. Venn, Symbolic Logic, London: Macmillan, 1881.

[13] D. Roberts, The Existential Graphs of Charles S. Peirce. The Hague: Mouton, 1973.

[14] S. Al-Fedaghi, "Schematizing proofs based on flow of truth values in logic," IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2013), Manchester, UK.

[15] S. Al-Fedaghi, Visualizing Logical Representation and Reasoning, The 15th International Conference on Artificial Intelligence (ICAI '13), July 22–25, 2013, Las Vegas, USA.

[16] S. Al-Fedaghi, "Enriched diagrammatic representation for resolution," The Asia Pacific Symposium of Intelligent and Evolutionary Systems, November 7–9, 2013, Seoul, Korea. This article also appears in Procedia Computer Science, Elsevier.

[17] S. Al-Fedaghi, "How to diagram your drama story," HCI International 2013, July 21–26, 2013, Mirage Hotel, Las Vegas, Nevada, USA. Lecture Notes in Computer Science/Artificial Intelligence, Springer. Extended version is published in the Communications in Computer and Information Science (CCIS) series, vol. 374, 2013, pp. 531–535.

[18] S. Al-Fedaghi, High-Level Representation of Time in Diagrammatic Specification, Procedia Computer Science Journal, This paper is from the 2015 International Conference on Soft Computing and Software Engineering (SCSE '15), University of California at Berkeley (UC Berkeley), USA, March 5-6, 2015.

[19] K. H. Rosen. Discrete Mathematics and Its Applications, 7th ed., 2011.

[20] D. Pritchard, "Knowledge," Central Issues of Philosophy, J. Shand, Ed., Blackwell, 2000, 24-36

[21] E. Gettier, "Is justified true belief knowledge?," Analysis, vol. 23, 1966, 121–123. http://philosophyfaculty.ucsd.edu/faculty/rarneson/courses/gettierphilreading.pdf

[22] Wikipedia, Gettier problem, Accessed May, 10, 2015.

Automated Monitoring and Validation of Synthetic Intelligent Behavior

Randolph M. Jones[1], Ben Bachelor[1], Webb Stacy[2], John Colonna-Romano[2], and Robert E. Wray[1]
[1]Soar Technology, 3600 Green Court, Suite 600, Ann Arbor, MI 48105, USA
[2]Aptima, 12 Gill Street, Suite 1400, Woburn, MA 01801, USA
rjones@soartech.com, ben.bachelor@soartech.com, wstacy@aptima.com, jcromano@aptima.com, wray@soartech.com

Abstract - *Validation of software models that emulate complex human reasoning has historically been informal, subjective, and difficult or impossible to scale to large numbers of models. This paper describes an approach to validation of intelligent behavior models (and semi-automated force, SAF, models more generally) that employs a formal knowledge representation (called a Behavior Envelope) to validate SAF behavior in both off-line and on-line modes of operation. The Goal Constraint System (GCS) employs constraint-based representations that enable requirement specification at different levels of abstraction and a penalty assessment approach that allows a subject matter expert to specify the relative importance of constraint violations. We describe the GCS language, interpreter, and several applications of the technology.*

Keywords: Constraint-based knowledge representation, software verification and validation, intelligent behavior modeling, synthetic intelligent forces

1 Intelligent behavior validation

Synthetic forces are an application of artificial intelligence that raise the quality of simulation-based Semi-Automated Forces (SAFs) to the level where they realistically emulate human tactical reasoning [1]. Because these models see use in realistic training and experimentation applications, validating the quality of modeled intelligent behavior is a key concern. The essential question to be addressed is "Given the richness and complexity of human behavior, how can we tell if any specific behavior produced by a synthetic force or SAF model is acceptably realistic?" Validation is one form of behavior evaluation that is useful for multiple reasons and at a variety of different levels of detail. For example, evaluation techniques are relevant to software verification, assessment of scientific contributions, assessment of student and SAF performance, runtime monitoring and intervention, evaluation of behavior-driven scenarios, and experimental assessment of new technologies. For the purposes of this paper, we will focus on software validation. In later sections, we will discuss approaches to extending our validation techniques to other types of intelligent-systems evaluation.

Validating human-level model behaviors (as well as actual human behaviors) is particularly challenging ([2] includes a thorough discussion of the challenges). Intelligent behavior is complex, and it is difficult to build automated validation tools that capture all the nuances of "good" and "bad" behavior. As a result, validation is often performed subjectively. Human subject-matter experts (SMEs) or training experts observe behavior and create assessments or scores based on their observations. Subjective validation allows the assessment of complex and nuanced behavior (to some extent), but it also has limitations:

- Subjective validation is inconsistent across different evaluators.
- Subjective validation is often inconsistent even when performed by a single evaluator.
- Validation criteria (i.e., requirements on behavior) themselves are often subjective and often not specified in detailed, archival, or formal form.
- Subjective validation is often qualitative or categorical, lacking nuanced assessment of individual decisions that led to a result.
- Subjective validation requires the availability of one or more human evaluators, which increases opportunity costs of validation.
- Depending on the complexity of the observed behaviors, validation may stress the cognitive capacity of the evaluator, especially for run-time validation.
- Validation may require deliberation that prevents the evaluator from keeping pace with the execution of the behaviors.

The overarching issues is that it is difficult and expensive to define objective requirements that capture all the nuances of human-level behavior. These limitations argue for an automated validation solution that provides rapid, consistent, and accurate assessments, using objective validation criteria than can be archived, inspected, and adjusted by humans. The primary gap for automation is having an inexpensive way to create formal requirements specifications to validate from.

We describe an automated behavior validation system called the Goal Constraint System (GCS). GCS works within a broader requirements-specification concept called *Behavior Envelopes*, being jointly researched and developed by Aptima and Soar Technology. Behavior Envelopes allow users to specify behavior validation criteria in the form of *behavior constraints* that hold within a particular *behavior context*. GCS comprises a formal representation language for Behavior Envelopes and a language for automatically scoring violations of constraints. It incorporates an implemented reasoner that infers the current behavior context for an entity, and monitors and scores observed behaviors for that context, either at run time or off line. We outline the need for formal behavior specification and the particular solution provided by the Behavior-Envelope concept. We then describe details of the GCS implementation and present several detailed examples of application of the GCS system, as well as higher-level descriptions of other GCS applications.

2 Expected behavior specification

From one perspective, behavior validation is the problem of measuring *observed* behavior against a formal characterization of *expected* behavior. There is a long tradition of formal behavior specification in computer science, particularly with respect to defining the requirements for the behavior of engineered software systems [3]. One of the primary advantages of a formal specification is that it requires unambiguous and objective descriptions of the desired behavior. This removes the elements of subjectivity, inconsistency, and ambiguity. Formal specification often provides the additional benefit that validation can be automated by computer software that is able to interpret the formal specification and match that specification to observable system behavior.

While these specification languages are formal, like computer programming languages, an additional advantage (usually) is that they provide a higher level of abstraction than a programming language does. This allows requirements builders to specify requirements for complex systems without having to duplicate all the work that would be required to build the system in the first place. However, for standard specification languages, this higher level of abstraction can leave the specification non-executable. This, in turn, can make system validation difficult to automate.

Property-oriented specification languages [4] facilitate automated validation. A property-oriented specification asserts particular relationships between elements of a system's data or behavior. A big advantage is that property-oriented specification does not need to be complete to be useful or to be automated. Automated software can monitor various assertions about properties and report any violations. Property-oriented specifications therefore have some appeal for application to the validation of intelligent behavior systems. However, there are two issues to be resolved in adopting a similar approach to validating complex intelligent behavior.

First, the context of a specification must be determined. In standard software engineering systems, assertions about behavior occur at the point in the code at which those assertions are applicable. This is feasible because, even for complex software, there are individual threads of execution that define the "location" of the execution logic at any point in time. In contrast, intelligent behavior involves much more loosely bound goals to be achieved, methods for achieving them, and processes for making sense of the world. In an intelligent system, all of these processes must interleave flexibly in ways that make it difficult to recognize a "state" for the system. (This behavioral flexibility explains in part why formalizations like state machines break down as behavior complexity increases.) We desire the ability to specify the context in which some property holds, including contexts that may not be entirely observable.

Second, intelligent behavior is rarely usefully classified as simply "correct" or "incorrect". Intelligent behavior is varied and flexible, and competence for accomplishing goals occurs in varying degrees. Thus, we desire a specification language that supports validation scoring functions that are not simply binary. The automated specification system should be able to indicate *the degree to which* an observed behavior meets the specification, rather than simply reporting that it fails to meet the specification.

3 Behavior envelopes

Aptima and Soar Technology have developed an approach to property-oriented specification languages that we term *Behavior Envelopes.* The approach builds and unifies work on scenario envelopes [5] and behavior bounding and variation [6][7]. A primary advantage of Behavior Envelopes is that they allow the specification of constraints to an arbitrary level of detail. As with property-oriented specifications for traditional software systems, this allows users to create inexpensive but useful behavior specifications, or to invest in more detailed specifications for particular intelligent behaviors.

Behavior Envelopes are a general concept to support a broad range of applications that all rely on the idea of a formal, declarative representation of behavior contexts and behavior constraints. For this paper, however, we focus on a particular implementation of Behavior Envelopes in GCS, targeted toward a smaller class of monitoring and validation applications (described below). A GCS Behavior Envelope consists of two primary components. The first component is a formal, relational representation of a situational context for an intelligent entity. The context can be considered a formula in predicate logic, composed of constituent predicates and propositions. The context may include observable features of the entity's situation (such as geographic location), as well as unobservable features describing the entity's internal state (such as a particular goal that the entity is trying to achieve). The second component is a formally specified set of constraints that the entity behavior should meet in situations where the behavior context applies. Each constraint is also a formal, logical predicate, often relying on relational predicates that related multiple properties together. GCS provides the machinery for making each logical predicate operational for a particular simulation system.

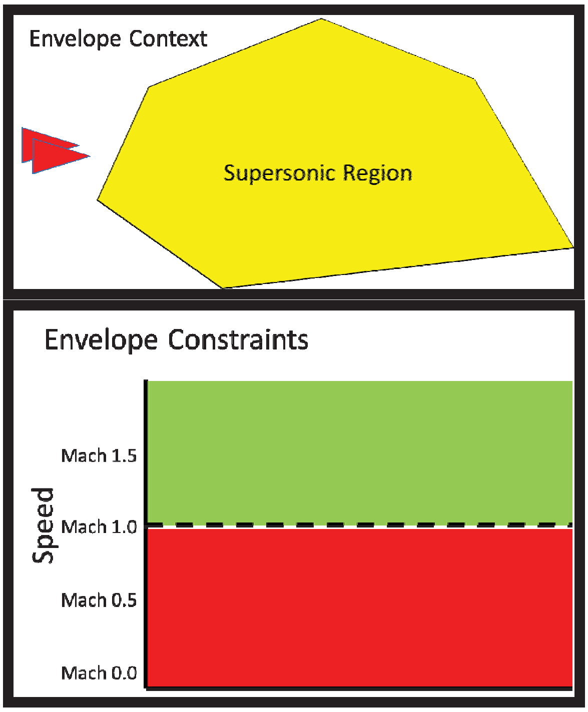


Figure 1. A Behavior Envelope consists of a Context and a set of Constraints. In this simple example, the Context specifies that the envelope only applies to entities who have a current position inside the Supersonic Region. The Constraints specify that any entities inside this region are considered to have acceptable speeds if the speed is above Mach 1.0, and unacceptable speeds if the speed is below Mach 1.0.

In order for the representations of context and constraints to qualify as "formal", a Behavior Envelope system must strictly define a set of domain-specific terms and predicates that compose to define each context and constraint. Thus, any particular instantiation of a Behavior Envelope system (such as GCS) must provide a well-defined language for specifying the conditions under which a particular envelope is relevant (Context conditions), as well as the conditions that must be met during the enforcement of a particular envelope (Constraints conditions). In addition, to operationalize these representations, they must be connected (via a programming interface) to the system generating the behavior. For example, a Behavior Envelope system that includes geographical regions for context must access a simulation interface that detects when entities are in particular geographical regions. GCS thus provides a level of formality necessary to enable objective, automated validation. We provide below examples of some of the formal representations built into GCS.

As a simple example, consider a Behavior Envelope that specifies a speed constraint over a geographic area. This type of constraint is common, for example, in training ranges. In such a case, the envelope context would consist of relations that specify the entity's position inside the controlled speed area. The envelope constraints would dictate the required speed limitations within this geographic area (See *Figure 1*).

A slightly more complex example envelope might describe the behavior expectations for an aircraft to fly a "racetrack" (oval) pattern. The envelope context in this case might be a complex set of conditions that specify when it is appropriate for the aircraft to be flying a particular racetrack pattern. These conditions would include mission specifications, the aircraft's geographic location, the position of the center of the racetrack point, the orientation expectations for the oval, and possibly some historical information about the entity (such as recent command-and-control messages).

Although these examples are simple, the GCS language supports arbitrary levels of complexity, for both envelope contexts and constraints. Users can create fine-grained envelopes when detailed validation is necessary, but complexity can be traded off in areas that do not require such detail. GCS thus supports a "spiral" approach to requirements, starting simple and increasing complexity as resources allow.

4 The Goal Constraint System

Thus, GCS provides a behavior validation capability (based on Behavior Envelopes) that enables user-specified validation of entity behavior at varying levels of detail. The GCS implementation consists of 1) the language that specifies Behavior Envelopes and 2) the executable system that interprets these envelops and uses them to monitor and validate observable intelligent behavior in a simulation. We describe each of these in this section.

4.1 The GCS language

The GCS language instantiates a specific version of the Behavior Envelope concept introduced above. The language supports the declaration of individual entities (i.e., the behavior generators), variables that bind at validation time to any of a predefined set of entities, and individual envelopes that define constraints on the behaviors of these individual entities or groups of entities. In GCS, the envelope context is primarily defined by a goal (or subgoal) that an entity (or group of entities) is attempting to achieve (note that this is the approach used in GCS, but Behavior Envelopes generally support a much wider range of context definitions). GCS provides methods to infer when an entity has spawned a subgoal from a previously active goal, or transitioned from

Table 1. A portion of an example GCS file for tactical air behaviors.

```
@fly_racetrack(=waypoint):
  **start -> @turn_to_inbound_racetrack_leg(=waypoint) within %time_delta minutes || inf
  **start -> @turn_to_outbound_racetrack_leg(=waypoint) within %time_delta minutes || inf


@turn_to_inbound_racetrack_leg(=waypoint):
  **start -> @fly_inbound_racetrack_leg(=waypoint) within %max_turning_time || inf

@fly_inbound_racetrack_leg(=waypoint):
  ^holds($has_racetrack_inbound_heading(=self)) || %medium_continuous_penalty per minute
  ^holds($at_cap_altitude(=self)) || %medium_continuous_penalty per minute
  ^holds($at_inbound_cap_speed(=self)) || %medium_continuous_penalty per minute
  $at_racetrack_distance(=self,  =waypoint)  ->  @turn_to_outbound_racetrack_leg(=waypoint)  ||
%large_one_time_penalty

@turn_to_outbound_racetrack_leg(=waypoint):
  **start -> @fly_outbound_racetrack_leg(=waypoint) within %max_turning_time || inf
```

achieving one goal to achieving a new goal. In contrast to alternative methods that assume agent goals are always explicit, GCS does not make that assumption. Thus, GCS can validate behavior generation methodologies that do not use explicit representations of goals, as well as those that do. This one of GCS' primary strengths, because it separates the representation of goals in GCS from the representation of goals in the behavior system. For example, GCS goals can be specified by SMEs who have no knowledge of a particular behavior system's implementation.

Each constraint set is a list of actions or goal transitions that the entity is expected to generate. A GCS Behavior Envelopes definition first declares the constants, variables, functional predicates, and goals relevant to the envelope. Then envelope constraint definitions specify relationships between series of functional predicates. GCS also elaborates the Behavior Envelope with a set of penalty functions. These penalty functions generate numeric scores when constraints are violated. Penalty values can be either "one shot", or they can accrue over time when envelope constraints are violated.

GCS defines functional predicates from an easily extensible set of native primitives. A GCS file that defines a set of constraints includes five sections: constant-value definitions, entity-variable definitions, predicate definitions, event definitions, and Behavior Envelope definitions. The Behavior Envelope definitions include the goal context for each envelope, the set of constraints for that context, and the penalty function for scoring constraint violations. Table 1 provides a snapshot of a portion of a GCS file describing envelopes for air combat behaviors. This snapshot does not include declarations or predicate definitions, but it gives two examples of Behavior Envelopes with constraints and penalty functions. One envelope's context is the "fly_racetrack" goal. This goal includes a subgoal constraint, indicating the conditions under which GCS infers an entity to be pursuing the subgoal "fly_inbound_racetrack_leg".

Constants provide symbolic references to constraint parameters that may take different values for different entities and scenarios. For example, a %radar_range parameter may specify the typical radar range of a particular aircraft, and the user could change this constant to validate a set of behaviors over a range of different aircraft. As another example, the %cap_orientation parameter can be changed when validating the behavior of different entities with different specific combat air patrol (CAP) missions.

Entities define the different types of objects over which GCS can specify constraints. Usually there is at least one entity variable representing the entity whose behavior is being validated. However, there can also be other "behaving" entities that interact with the primary entity of interest. Additionally, there can be "non-behaving" entities, such a geographical features (e.g., waypoints or boundaries) and physical systems (e.g., radar systems or weapons systems). Each type of entity has associated properties that can be included in envelope constraints. Another strength of GCS is that it can validate the aggregate behaviors of groups of entities or even entire scenarios, as well as individual-level behaviors.

Predicates define relations of interest that combine to build envelope constraints. Predicates are the primary form taken by individual constraints. A predicate is a relation that can either be met or unmet in a particular situation, and predicates can be constructed from other predicates, from events, and from primitive relationships (such as equalities and inequalities). An example simple predicate is "entity_is_airborne", which is met for an entity if the entity's altitude above ground level is greater than zero. A more complex example is "has_racetrack_inbound_heading", which is met if the reciprocal of the entity's heading is within some parameterized range around the mission-specified CAP orientation assigned to the entity (See *Figure 2*).

Events are similar to predicates, in that they also define relations to be included in envelope constraints. The difference is that the GCS Interpreter (described below) continuously checks whether predicates match. In contrast, for events, GCS only checks for transitions between being "unmet" and being "met". Once an event has been "met", the interpreter does not continue monitoring that event. An example event is "bandit_is_destroyed".

Goals define the contexts of the Behavior Envelopes, as well as the conditions under which GCS should transition to monitoring a new envelope (either via activation of a subgoal or transition from one goal to a subsequent goal). The current set of "active envelopes" (isomorphic to the current set of "active goals") defines the constraints that should be monitored for a particular entity. An envelope is active as long as GCS assumes that the entity is still pursuing the goal that defines the context for that envelope.

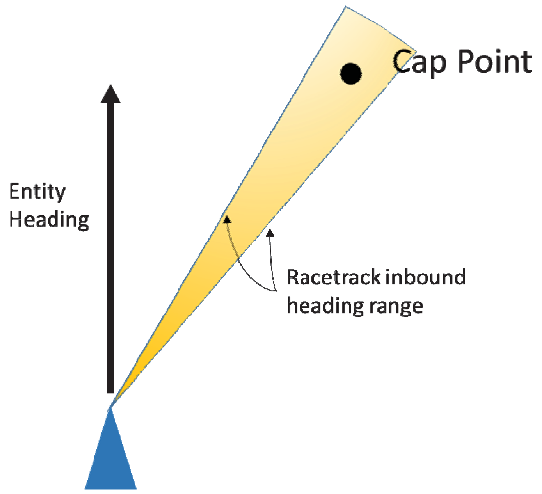


Figure 2. The "has_racetrack_inbound_heading" predicate is true if the entity's current heading lies within parameterized limits for approaching the Cap Point.

Each envelope constraint conjoins predicates and events, together with a penalty formula to use when the constraint is violated. Envelopes also include special "goal transition constraints". That is, an envelope's constraints can specify the conditions under which the entity "ought to" transition to a new goal. The GCS interpreter performs a search (using the penalty functions as heuristics) to infer whether the entity has started pursuing the behaviors associated with the new goal. The interpreter invokes the goal-transition constraint's penalty formula if the constraints indicate that a goal transition should have occurred, but the entity's behavior does not appear to be consistent with a transition to the new goal. Penalty formulas can have varied forms. Two primary forms are "one-time" penalties and "continuous" penalties. One-time penalties incur on the initial violation of a constraint. Continuous penalties accumulate over the duration of a constraint violation. For continuous penalties, it is also possible to specify a time period for the penalty accumulation (such as "per second", or "per minute"). It is also possible to specify a penalty value of "inf", meaning an infinite penalty, to represent constraints that are always expected to hold.

4.2 The GCS interpreter

The GCS Interpreter matches envelope contexts (goals) and constraints (specified in the constraint language) to streams of behavior data. The interpreter can run simultaneously with a dynamic behavior stream (e.g., during run time of a scenario) or offline on a collected data log file. The first job of the GCS interpreter is to infer, at any given time point, which goal(s) an entity is pursuing, indicating which Behavior Envelopes should currently be in force. The interpreter's second job is to monitor which constraints are satisfied or violated for the set of active Behavior Envelopes. Finally, the interpreter uses the penalty scoring functions to compute penalty values for each violated constraint. Using heuristic search, the penalty values also allow the interpreter to infer whether the entity has made any changes in the goals it is pursuing. Ultimately, the interpreter generates a series of penalty values for each set of envelope constraints, reflecting how well the entity is meeting the behavior requirements for each active goal at each point in time. These penalty values aggregate into general validation measures of the behavior fidelity the entity generates for each Behavior Envelope.

The first implementation of GCS modeled entity behavior as a series of unknown goals with associated constraints. This version assumed that an entity could potentially transition from a current goal to any other goal, in response to some event. This approach naturally produced quite a large search space of potential goal sequences, which the interpreter navigated using the A* search algorithm. The interpreter used the GCS constraints and penalties associated with each envelope to compute a "best fit" score of the entity's current behaviors to each particular set of envelope constraints. This allowed the interpreter to infer the goal sequence pursued by the entity during the course of a scenario. This approach was effective in analyzing off-line logs of behavior data. However, it was not efficient enough to produce timely validation of real-time behavior data streams.

For run-time validation, we refactored the GCS Interpreter, leaving the GCS language intact. Version 2 of the interpreter models entity goal sequences as a hybrid Hidden Markov Model / Finite State Machine. The interpreter tracks, in real time, the lowest possible penalty that could result in each possible goal being an active goal. Transition penalties between goals are recalculated each tick based on the current state of the simulation. Because a graph structure indicates the possible goal transitions, the GCS interpreter can efficiently track entity progress through potential goal sequences. This allows the solver to infer goals and compute penalties at low computational expense, allowing validation of both off-line data logs or run-time data streams. This approach does have the limitation that it cannot recognize goal transitions not specified in advance but, for synthetic force applications, such a limitation is typically acceptable.

5 Applications

The automated behavior monitoring, validation, and scoring that GCS implements can be used in a number of different applications. This section introduces potential applications and describes in more detail several examples of actual implementations.

5.1 Run-time SAF behavior corrections

If a constructive simulation scenario exhibits a run-time problem, such as an unexpected and incorrect SAF behavior, then an obvious remediation step would be to attempt to remove or reduce the violation while the scenario is still running. One way to use GCS would be to signal a human operator to respond to such a behavior violation. Alternatively, GCS could simply record violations in a log. We have been exploring a third option, in which violations inform adaptive technology that can attempt to “repair” the SAF behavior problem during execution (although this application can also do signaling and logging).

We are using violations detected by GCS to inform the Training Executive Agent (TXA), an adaptive operator-aiding technology [8]. The TXA is integrated with a distributed simulation, and it exploits the behavior representations of the simulator to modify the “native” behavior of its SAFs. The goal of the TXA is to reduce operator workload by automating the monitoring and management of a variety of interventions during the execution of a training scenario. GCS implements a major part of the monitoring and validation capability. As an example, one possible training objective is for trainees to intercept bandits moving at supersonic speeds. There is a geographic zone in which the bandit entities are expected to exhibit those supersonic speeds in order to support this training goal. Under “normal” conditions, a human operator might need to pay attention to aircraft entering this area and increase the speed if needed. The TXA reduces operator workload by automating these monitoring and management functions. GCS can be used to monitor entity behaviors and detect if any entity speeds are in violation of the expected speeds for the entity’s geographical region and goals. If a violation is found, GCS uses its penalty functions to score the severity of the violation. This information is then passed to the TXA, which adjusts the entity’s behavior in order to bring the behavior in line with the expected constraints.

In the longer term, the ability of GCS to assess penalties in real time, together with TXA’s ability to permute SAF behavior, could be used in a more open-ended way (“emergent repair”). However, thus far we have used GCS only to identify violations of specific types and then trigger pre-defined TXA repairs. For example, an observed speed violation triggers a TXA directive that matches the prescribed SAF speed to range requirements at the entity’s current location. The advantage of using GCS in this application is that the TXA can use the penalty score (and accumulating scores) to determine if/when to modify behavior, rather than simply recording and responding to a constraint violation. For example, for an aircraft briefly transiting thru the supersonic area, GCS can apply a different penalty function (based on context) than a bandit that is intended to engage the trainees.

5.2 Validating training scenario goals

SAFs may perform actions that are appropriate from a tactical perspective, but that may not match the specific required actions for a particular training context. For example, imagine a training goal intended to provide a trainee with the experience of two successive air-to-air intercepts. In this situation, the trailing bandits need to stay far enough away from the lead group to not interfere with the initial intercept but close enough that the trainee has to engage them immediately on successful prosecution of the first group. This is the instructor’s intent for this scenario.

On the TXA effort, we are using the GCS to provide objective assessment of the “presentation quality” (from the perspective of instructor’s intent) of a scenario as it evolves under a range of experimental conditions [9]. If all the various interactions in support of training goals execute as desired, then the scenario can be scored as having high quality. Various undesired interactions can impact the overall quality of the training scenario to different degrees. Subject-matter experts defined scoring criteria for each experimental scenario, and then we encoded the constraints and scoring functions into GCS. This allowed us to run a series of experimental variations to a training scenario, and to compute the presentation quality for each experimental condition in an automated and objective fashion.

5.3 Verification and validation of SAFS

If a SAF behavior results in some kind of (tactical or instructional) constraint violation, then there are additional potential responses, besides attempting to repair or adapt the current scenario. When GCS detects a poor behavior, this could be interpreted as a signal for repairing or refining the underlying behavior representation for future use (and thus eliminating the need for run-time repair in the future). We have investigated the potential for behavior refinement from several different perspectives. Most importantly, we have developed a test harness that enables systematic variation of parameters within a SAF behavior representation. The test harness is able to replay a given scenario with specific variations and combinations of behavior parameters. The test harness uses GCS to validate, score, and summarize the resulting behaviors. The results of this analysis indicate the range of “presentation qualities” that a SAF behavior definition can produce, as well as a sensitivity analysis of the behavior definition to various parameter settings.

Systematic variation of parameters helps modelers, operators, and instructors develop an understanding of the “topology” of a behavior model. Although such topology analysis has not seen significant application, the testbed investigations with GCS demonstrate that this type of analysis is feasible and worthwhile, especially for generalized behaviors. Using this approach, we foresee a much larger future role for GCS in facilitating behavior verification and validation. Systematic variation using GCS may be especially useful for learning systems that attempt to synthesize behavior representations from observation data [10].

5.4 Additional applications of GCS

There are additional potential applications of GCS that we have not yet implemented or explored significantly. For example, the search-based inference engine in the GCS interpreter can also be viewed as a form of plan recognizer. This type of plan recognition can be used to infer the intent of constructive force models without having access to the models' internal representations (as above), or also to generate explanations of observed behaviors when we do have some knowledge of a model's internal representations. The use of GCS to support the TXA experiments demonstrates the ability to go beyond validation of individual entity behaviors to validating aggregate behaviors at the scenario level. Additionally, GCS could assist in the creation of suitable scenarios or the run-time adjustment of scenarios (extending the functions of the TXA).

6 Summary and conclusions

We have introduced the problem of generating useful, objective, and automated validation of complex intelligent behaviors in modeling and simulation. The fluidity and complexity of human-level behavior requires validation solutions that extend techniques for standard software systems. We have also described and illustrated one implemented solution to this problem. The Goal Constraint System relies on the concept of Behavior Envelopes, which is itself an adaptation of property-oriented specification to the complexities of intelligent behavior systems. The GCS representation additionally extends the Behavior Envelope concept to accommodate unobservable features (entity goals) and quantitative scoring of constraint violations. The GCS interpreter exploits its penalty scoring functions to assist in inference about entity goals, as well as to produce quantitative scores of individual and aggregate entity behaviors. We have identified a number of application areas in which such automated and objective validation is useful.

7 Acknowledgments

This work is supported in part by the Naval Air Warfare Center Training Systems Division (Aptima Prime Contract: N68335-12-C-0146) and the Office of Naval Research project N00014-1-C-0170 Tactical Semi-Automated Forces for Live, Virtual, and Constructive Training (TACSAF). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of Defense or Office of Naval Research. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

We would like to thank sponsors at NAWC TSD and ONR who have provided insights and operational perspectives in the development of GCS: Beth Atkinson, Heather Priest, Melissa Walwanis, Amy Bolton, and LCDR Brent Olde. GCS also benefitted significantly from the contributions of J.T. Folsom-Kovarik, Leon Pryor, Marcus Huber, and Sam Wintermute at Soar Technology.

8 References

[1] Jones, R. M., Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, P., & Koss, F. V. (1999). Automated intelligent pilots for combat flight simulation. *AI Magazine, 20*(1), 27–41.

[2] Wallace, S. (2003). *Validating Complex Agent Behavior*, Ph.D. Thesis. The University of Michigan, Ann Arbor.

[3] Lamsweerde, A. V. (2000). Formal specification: A roadmap. In *Proceedings of the Conference on the Future of Software Engineering, ICSE '00*, 147-159.

[4] Dasso, A., & Funes, A. (2009). *Formalization process in software development*. IRMA International.

[5] Stacy, W., Picciano, P., Sullivan, K., & Sidman, J. (2010). From flight logs to scenario: Flying simulated mishaps. In *Proceedings of Interservice/Industry Training, Simulation, and Education Conference* (I/ITSEC 2010). Orlando, FL.

[6] Wallace, S., & Laird, J. E. (2003). Behavior Bounding: Toward Effective Comparisons of Agents & Human Behavior, *International Joint Conference on Artificial Intelligence*.

[7] Wray, R. E., & Laird, J. E. (2003). Variability in Human Behavior Modeling for Military Simulations. *Proceedings of the 2003 Conference on Behavior Representation in Modeling and Simulation*. Scottsdale, AZ. May.

[8] Wray, R. E., & Woods, A. (2013). A Cognitive Systems Approach to Tailoring Learner Practice. In J. Laird & M. Klenk (Eds.), *Proceedings of the Second Advances in Cognitive Systems Conference*. Baltimore, MD.

[9] Wray, R. E., Bachelor, B., Jones, R. M., & Newton, C. (to appear). Bracketing human performance to support automation for workload reduction: A case study. Accepted for publication in *Proceedings of the Human Computer Interaction International (HCII) Conference 2015.*

[10] Levchuk, G., Shabarekh, C., & Furjanic, C. (2011). Wide-threat detection: Recognition of adversarial missions and activity patterns in Empire Challenge 2009. In *Proceedings of the SPIE Defense, Security, and Sensing Conference*.

Classification of Debate Threading Models for Representing Decentralized Debates

Abdulrahman. Alqahtani [1,2], and Marius. Silaghi[2]
[1]Department of Computer Science, Najran University, Najran, Saudi Arabia
[2]Department of Computer Science, Florida Institute of Technology, Melbourne, FL, USA

***Abstract*—We analyze Debate Threading Models for representing Peer-to-Peer Decentralized Debates. The challenge, we address, is how to structure arguments in debate on a clear topic such that they are informative to the users trying to make up their mind, and such as to foster reciprocal understanding of each of the views in the community. Moreover, to enable the participation of average skilled users, the technique for inputting the arguments may have to be *simple* (i.e., close to the common practice, which is a subjective criteria). Knowledge representation for serving human users has been an important area of study in Computer Science and Artificial Intelligence. The area has grown significantly with conferences on topics such as ontologies, collaborative filtering, semantic web, and argument maps. We study and compare the various existing structures for representing complex knowledge in understandable forms, focusing on their applicability to decentralized debates with arguments from general users. We believe that this is the first kind of work that deals with the issue above.**

Keywords: Related Work, Knowledge Representation, Threading Models for Arguments in Electronic Debates, Threading Model Classification, Debate Threading Model, Comparison between TED, YourView, and DDP2P

I. Introduction

One can observe that electronic arguments are significantly different from face to face debates. An effort was made over the centuries to find optimal ways of organizing face-to-face debates [1]. Research is needed to decide whether current electronic debates are carried out with the optimal debate threading models and debate representation, as a means of knowledge representation. P2P Petition Drives requires decentralized synchronization of equal peers' data [2]. The information being exchanged is about opinions and arguments in debates ocurring around the petitions. DirectDemocracyP2P is a possible implementation of this approach. The information that it exchanges is structured as a set of autonomous pieces of knowledge (objects). Each such object belongs to one of the supported types: organization, motion, justification, vote, news, translation, constituent, neighborhood, witnessing stance, peer or tester. The types of objects that are relevant for the debates are: motions, justifications, and votes. The other objects are relevant only in the measure in which they can validate the votes (e.g., organizations, neighborhoods, constituents and witness stances), help with communication (e.g., peer and tester) or help with usability (e.g., translations and news). Those objects and their mechanism of validation are part of different studies and are outside the scope of this research.

In petition drives, the only relevant voting alternative is *Support*, since legally for moving on a motion, one only counts the supports. Opposing votes are normally counted only by the final official referendum/ballot. They are present in DirectDemocracyP2P is a way of applying collaborative filtering to the motions viewed for gathering the needed support [2], [3]. In DirectDemocracyP2P, users can support somebody else's justification, as an alternative to providing his/her own justification. Justifications with large support can be favored by graphical user interfaces, as they may better represent the opinion of the group. DirectDemocracyP2P (DDP2P) system is Representing Decentralized Debates. Threads In a DDP2P application start with a motion that is relevant to a given organization. User can vote on motion with only single justification. DDP2P is counting opinions and justifications.

II. Related Work

A few researchers have already addressed the relationship between social network behavior and these networks' intended purposes [4]. This section highlights previous studies on the type and quality of collaboration based on social networks. In this report, the related work is presented and classified according to the means and purpose of the studied collaboration. We identify the following types of generic tools for collaboration, on an idea, between unsophisticated users:

- Arguments
- Thumb up/down
- Voting
- Threading

a) Arguments:: According to the Merriam Webster dictionary, argumentation is "the act or process of giving reasons for or against something: the act or process of making and presenting arguments" where an argument is "a statement or series of statements for or against something" [5]. The possibility to express arguments in favor of or in opposition to ideas is considered an important instrument in collaboration [6].

b) Thumb up/down:: An easy way to garner feedback on ideas and arguments is to let people show their opinion using simple tags such as a thumbs-up or a thumbs-down. Often this is done in systems where participants are not authenticated. Moreover, users can often use a thumbs-up tag for opposing arguments, thereby making it difficult to use these tags for extracting reliable statistics.

c) Voting:: Voting is an approach to measure the opinion of a certain group of people. Unlike the thumbs-up/thumbs-down practice mentioned above, commonly the identity of people involved in voting is verified and each person can vote only once. Therefore, a user cannot vote simultaneously on opposing arguments. Statistics about opinions can be reliably estimated.

d) Threading:: Many writers are skeptical about the usefulness of the comments or threads that are placed on common web forums [7], despite the large number of useful comments. However, the idea of abolishing comments is not popular. Users feel more motivated to visit forums where they can comment since they may believe in the impact of these comments.

Besides collaborating on ideas, people also collaborate on projects by contributing work or resources [8], [9].

III. Knowledge Representation

The best example of the mechanism for Knowledge Representation is structured knowledge in tree or graph form of either classification, ontologies, or taxonomies. Many representation languages and mechanisms were developed to take advantage of these concepts: [10]

- Semantic Networks: A Semantic Network consists of a set of nodes that are linked by arcs [11].
- Conceptual Graphs: A conceptual graph connected between concepts nodes and conceptual relations nodes [12].
- Frame-based System: Frame based system is ordering knowledge into chunks (frames). Each frame contains information of how to use the frame and expectations if something is wrong [13], [14].

A. Major Knowledge Representation Types

Classification:

- Ontologies. The research in Ontologies deals with the establishment of languages that have no ambiguities. This language should be powerful enough to represent knowledge. For example, a message based on a standard ontology starts by a reference to the ontology version, and then has to follow its strict syntax. Ontologies define communities [15].
- Logic
  One of the techniques, that is used to store sentences within the computer, is mathematical logic. It is used in Knowledge databases such as those used by PROLOG and GOLOG. They are difficult to use with general users [16].
- Semantic Networks
  A Semantic Network consists of a set of nodes that are linked by arcs. In general, nodes represent concepts while arcs represent the relations between concepts [17].
- Mind Maps
  Mind Map is the name given to a tool or a means of expressing human ideas. It is drawn on a forked tree and relies on the pictures and words in the human memory. Since the human memory is a graphic memory, which depends on information linked with pictures, mind maps help to save and retrieve information easily. The use of mind maps is an innovative way to assist students to understand key information. Mind maps have the ability to provide students with an information retention strategy, integration of critical thinking, and problem solving skills [18].
- WordNet
  WordNet is a huge lexical database of English. It is a set of cognitive synonyms that have been collected from nouns, verbs, adjectives, and adverbs with a clear expression of concept [19].
- Argument Maps
  An Argument Map is a logic structure for arguments. It breaks up the argument into statement, reason, or fact. Also, it shows the relation between parts of the argument [20]. A lot of people have difficulty understanding complex arguments presented in textual form. An argument map is a graphical way to represent the complex relations between relevant statements. Therefore, it can improve human comprehension. It looks similar to a flowchart and a specialization/generalization hierarchy. There exist tools that can be used to build argument maps [21]. These argument maps have been built to help people better understand the issues related to a given problem. Arguments are frequently attached to emotions. In argument maps, there is no requirement for arguments to be objective. Simple and complex arguments are captured with the help of the diagram.

B. Uses and Purpose of Knowledge Representation Models

There are many benefits of formal Knowledge Representation Models.

Some mechanisms are intended to simplify automatic data mining. Data Mining is basically a process for inspecting and searching for specific information in large amounts of data. Other knowledge representation mechanisms are used to enhance human comprehension, such as mind maps, and argument maps. To evaluate arguments, different techniques are discussed in [22], [23]. Graphical representation of evidence-based dialogue, questions, ideas, pros, cons, and data helps in scientific reasoning [24]. In teaching, argumentation mapping helps the student to build critical thinking skills [25].

IV. Threading Models for Arguments in Electronic Debates

A. Introduction to Threading Models

The relevant point to mention about threading models is that users can benefit in their decision making by exploring comments and justifications provided by others (e.g., on fora or various social networks). This exploration can be enhanced not only by search engines, but also by using hints provided by links generated by the comment authors themselves.

The whole World Wide Web was designed for improving information exploration based on links (here hyperlinks) [26]. The Internet fora and social networks took this a step further by hosting such links in a more compact way but typically at the expense of generality. Namely restrictions are added as to how many links are possible from each article item. The most common and simple restriction is one where each article (comment) can link to a single other article, to which it is a “reply". This structure, together with restrictions on the maximum chain of “replies", gives rise to the so-called tree structures seen on common Internet fora. A compact representation can be enabled by such rules.

Conversation Threading has been standardized for the IMAP email systems [27]. A common technical classification is:

- **Client based**: Microsoft Outlook, Thunderbird
- **Web based**: Gmail, Reddit, Slashdot

B. Examples of Threaded Fora Specializing in Debates

There are many fora on the Internet. They are frequently attached to articles in online newspapers or next to items in online shops or product rating sites. Here we focus on fora that are specially designed to support debates on ideas. Some of the other fora are occasionally used for debates around linked issues but they are not generally designed to help navigate or extract a conclusion for those debates.

The three fora that we have identified as specially designed for debates are:

- TED [28]
- YourView [21], [29]
- DebateDecide (Client-based version: DirectDemocracyP2P) [3], [30]

Of these fora, only TED is currently used spontaneously by real users. The other two fora (YourView and DebateDecide) have so far only been used by controlled groups as part of usage experiments.

e) Threading of TED : The TED website is claimed to be designed for helping in sharing ideas, supporting debates, and addressing questions about scientific topics for educational purposes. Comments in TED are threaded but the amount of support of the idea cannot be easily extracted.

f) Threading of the Forum “YourView": [31]

YourView is being developed as an experiment in visualizing political debates around elections. A fund was used to encourage a set of users to register, argue, and vote on issues relevant to the parties involved in a recent election in Australia. Comments are sorted by support or rejection of the motion.

YourView is an online platform for presenting debate comments and helping users to explore these comments by scoring them. Registered users have the right to vote and the ability to submit multiple comments with their vote. Moreover, the participant can give a thumbs up or thumbs down for any vote or comment (multiple simultaneously!). The score change produced by a thumbs up/down differs from user to user depending on a measure called the ”credibility score.” The ”credibility score” is computed with a secret formula based on the activities of the user and on the responses of others to these activities. In YourView, users do not really have the possibility to customize the formula for the credibility [31].

g) Threading of DebateDecide: As in YourView, comments are sorted by type of support of the motion that they answer [2].

C. Types of Threading Models for Arguments in Electronic Debates

Only a few types of Threading Models are common for Arguments in Electronic Debates:

- Single level of comments [29]: This has an article and the list of comments to it.
- Two levels [29]: This has the article, comments to it, and comments that answer comments in the first level.
- Unlimited hierarchy [32]: Comments can answer any previous comments.

From the perspective of the ordering of comments we have:

- Comments sorted by support and rejection of the article [29].
- Comments sorted by support and rejection of the comments that they answer [30].

V. Threading Model Classification

Threading Models can be classified based on the number of links into "Single-link"and "Multiple-links". Also, each class of link numbers (Single link and/or Multiple links) can be characterized based on the length of the allowed chains (Single-homogeneous, Single-heterogeneous, unlimited-homogeneous, unlimited-heterogeneous, limited-homogeneous, and limited-heterogeneous).

Threading Models can be classified into "Single-derivation"and "Multiple-derivations".

- Single-derivation: Single derivation of Threading Models provides one way down the chain [28]. The motion

has only one method to link with comments or justification.
For example:
Suppose we have the motion *"Children should drink milk daily"*
Suppose one user supports this motion with justification *"Milk is a health drink".*
A second user rejects this motion with justification *"Children should drink juice daily".*
The diagram in Figure 1 is an example of single-derivation.

Figure 1: Example of Single-Derivation.

- Multiple-derivations: Multiple-derivations of Threading Models introduce multiple ways down the chain [30]. The motion has multiple methods to link with comments/justification, suggestion and/or request.
  For example:
  Suppose we have again the motion *"Children should drink milk daily".*
  We have these derivations "Vote" and "Suggestion" which are indirect linked with each other by a motion.
  Suppose users submit reactions on the vote derivation as per the previous example.
  Others users can post proposals on the suggestion derivation *"Children should drink water daily." and "Ask your child's doctor"*
  In this example, there could be multiple suggestions. The diagram in Figure 2 is an example of motion interaction with multiple-derivations.

Also, each class of derivation numbers (single derivation and/or multiple derivations) can be characterized based on the number of choices.

- Single-choice: All items under the derivation have the same semantic [28]. The diagram in Figure 3 is an example of the single-choice.
- Multiple-choices: The items under the derivation are classified by semantics under different choice names [29]. The diagram in Figure 4 is an example of Multiple-choices derivations.

The general augment structure of the Debate Threading Model of the DirectDemocracyP2P system is provided by

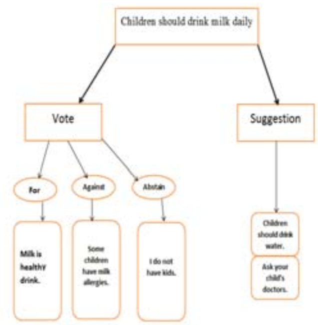


Figure 2: Example of Multiple-derivations

Figure 3: Example of the single-choice.

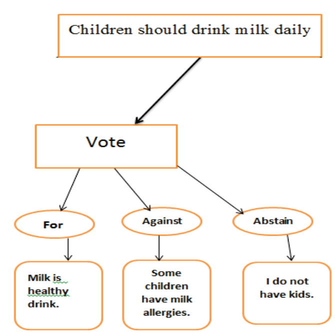


Figure 4: Example of multiple-choices.

the following set of rules:

CREATE (Organization)
PROPOSE (Science/Education/Religious/ Business)

CREATE (Motion)
PROPOSE (Topic/Question/Statement)

CREATE (Justification)
PROPOSE (explanation/ reasonable grounds)

VOTE (Motion/ Justification)
REQUEST (Explanation/ Reasonable grounds)
SUPPORT (Motion/ Justification) |

OPPOSE (Motion/ Justification) |
ABSTAIN (Motion/ Justification) |

CREATE (Motion).

In a DDP2P application, all debates and arguments start with a motion that is relevant to a given organization. The **organization** is a definition of a scope (set of relevant topics) and a constituency (set and weight of users) whose opinion is relevant to the decisions. A **motion** could be a topic, a question, a statement, or any idea that forms the subject of the debate between the parties, who are defined as being members of a given organization to which the motion is relevant. Every motion gets at most a **vote** by each constituent, whether they are for it or against it, with an optional **comment** to justify their stance. Anyone can add additional comments (justifications) to their own thread or give feedback to any previously posted comments (threads) by others. However, when submitting a second comment, a user automatically retracts his/her support for any previous comments. This allows the user to highlight a motion of interest and to see the corresponding participants' votes or replies.

VI. Debate Threading Models

TED's threading models can thereby be described as single-link and unlimited-homogenous while DDP2P is characterized as single-link and unlimited-homogenous chains. Next we give a formal definition of the threading model.

Definition 1 (Debate Threading Model): The set of restrictions placed on the number and type of links in an article item, as well as on the global rules on types and size of link chains, form a "*debate threading model*."

For a given threading model, multiple ways of graphically displaying the result to exit. For example, some items (i.e., at given depths in the tree) could be hidden until required by users. Otherwise, links can be shown graphically or the number or existences of connected items can just be mentioned. We want to separate the knowledge relations by users representation.

Definition 2 (Debate Representation Model): The set of rules describing how debate instances are graphically presented to viewers is called "*debate representation model*."

Some fora specialize on debates for specific topics: music, fashion, software bugs, and politics. Since some topics are more disputable than others, different threading models may be appropriate for different topics and, thus, fora.

VII. Comparison between TED, YourView, and DDP2P

As we mentioned previously, this study focuses on three fora for electronic debate (TED, YourView and DDP2P). We can compare these fora across some criteria to better understand differences. Table I compares between TED, YourView and DDP2P

	TED	YourView	DDP2P
Voting	None	Count the voting, topics and participants	Count the voting
Comments(justifications)	Comments not classified	Comments classified by vote	Justifications classified by vote
Threads	Comments threaded as Reply-To	Comments threaded as Reply-To	Justification threaded as Answer-To
Emotion thumb up/down	Part of comments	Part of comments	None
Metrics of ordering	Ordering by date	Ordering by date, credibility	Number of votes and/or date
Threading models	Only Single link, unlimited, homogeneous	Only Single link, homogeneous	Only Single link, unlimited, heterogeneous
Comments by participants	Unlimited comments	Unlimited comments	Only Single justification
Life span	limited	limited	Unlimited
Relations of comments	Replies and replies to replies	Unlimited replies	Unlimited replies with vote
Display of relation	Embed tree	Embed tree	Linked derivation

Table I: Comparison between TED, YourView, and DDP2P, from the perspective of their support for debates

Criteria:
In this paper we provide the comparison between TED, YourView, and DDP2P from the perspective of their support for debates.

- Voting:
  It is a score that counts the number of supports. This criterion has as domain:
  - None
  - Count the votes
  - Other score (Count comments, topics, participants, credibility, etc)

  TED has a debate representation model that is not structured by the type of voting.
- Comments/Justifications:
  These are arguments presented by voters. The domain of this criteria is:
  - Not Classified
  - Classified by Vote

  TED does not provide a structured comment section. All comments are in the same column and the user, once again, has to go through each of them to assess the decision behind each comment. While YourView uses the term argument, and DDP2P uses the term"Justification" to identify any argument entered by the users, the concept is similar to the other two fora. Both YourView and DDP2P provide a structured comment/justification section that is organized by voting choice.
- Threads:
  Threads are graphs/trees of arguments and relations between those arguments. TED and YourView calls the relation Reply-To. DDP2P calls the relation Answer-To.
- Emotion Thumb (Up/Down):
  This is a feature that allows the users to express their emotional support of an argument by selecting "Up" when they agree and "Down" for disagreement. Some platforms do not offer this feature while others have it available. The domain for this criteria is:
  - Part of voting
  - Separate from voting
  - None

  TED does not have a feature that allows the users to express their emotion (Thumbs Up/Down). YourView allows the users to express their emotion by selecting Thumbs Up/Down for any comment. DDP2P allows each user to exercise a Thumbs Up/Down selection one time only, as part of its vote.

- Metrics of Ordering:
  This is a sorting option that allows the user to order the comments and arguments. The domain for this criteria is:
  - Ordering by date
  - Ordering by the number of votes
  - Ordering by credibility
- Threading Models:
  They are the general rules on types and size of link chains (Homogeneous or Heterogeneous). Homogeneous: all links have the same semantic: e.g. refutation. Heterogeneous: links can have different semantics: e.g. refutation, support. There are a several types available:
  - Single-homogeneous
  - Single-heterogeneous
  - unlimited-homogeneous
  - unlimited-heterogeneous
  - limited-homogeneous
  - limited-heterogeneous
- Comments by participants:
  This is the possible number of comments by participants. The domain for this criteria is:
  - Unlimited comments [29].
  - Only Single justification [30].
- Life span:
  It is the maximum amount of times that users are allowed to participate by votes or comments. The domain for this criteria is:
  - Limited [28], [29].
  - Unlimited [30].
- Relation of comments:
  There can exist relations between comments. The domain for this criteria is:
  - Replies and replies to replies [28].
  - Unlimited replies [29].
  - Unlimited replies with vote [30].
- Display of relationships of comments/threads:
  Here we classify techniques used to display the relation of comments. The domain for this criteria is:
  - Embed tree [28].
  - Linked derivation [30].

Threading models of TED have only single link and are unlimited-homogeneous.

VIII. Conclusion

Threading models for representing decentralized debates constitute a relatively unexplored field of research. This area significantly relates to many different activities in our lives such as communication, chatting, argumentation, and education. Here, we introduce a classification of debate threading models. We identified some generic tools for collaboration on an idea (Arguments, Thumbs up/down, Voting, Threading, etc. On the Internet, we can find many fora attached to some blogs and other sites that provide the users with a comment/justification opportunity. However, most of the time, they are not designed to provide or extract a conclusion on any ongoing debate. There are three platforms we focused on in this report (TED, YourView, and DDP2P) that are designed to structure comments for electronic debates.

References

[1] H. M. Robert, *Pocket Manual of Rules of Order for Deliberative Assemblies*. SC Griggs, 1896.

[2] K. Kattamuri, M. Silaghi, C. Kaner, R. Stansifer, and M. Zanker, "Supporting debates over citizen initiatives," in *Proceedings of the 2005 national conference on Digital government research*. Digital Government Society of North America, 2005, pp. 279–280.

[3] M. C. Silaghi, K. Alhamed, O. Dhannoon, S. Qin, R. Vishen, R. Knowles, I. Hussien, Y. Yang, T. Matsui, M. Yokoo *et al.*, "Directdemocracyp2pâĂŤdecentralized deliberative petition drivesâĂŤ," in *Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on*. IEEE, 2013, pp. 1–2.

[4] R. Cross, S. P. Borgatti, and A. Parker, "Making invisible work visible: Using social network analysis to support strategic collaboration," *California management review*, vol. 44, no. 2, pp. 25–46, 2002.

[5] M.-W. Inc., *Merriam-Webster's dictionary*. Merriam-Webster.

[6] C. Rinner, "Argumentation maps: Gis-based discussion support for online planning," *Environment and Planning B: Planning and Design*, vol. 28, no. 6, pp. 847–863, 2001.

[7] J. D. Barkan, "Comment: Further reassessment of âĂIJconventional wisdom:âĂİ political knowledge and voting behavior in rural kenya." *American Political Science Review*, vol. 70, no. 02, pp. 452–455, 1976.

[8] R. Bhathal, "Australian optical seti project," in *Bioastronomy 99*, vol. 213, 2000, p. 553.

[9] R. Ferzli, R. Bazzi, and L. J. Karam, "A captcha based on the human visual systems masking characteristics," in *Multimedia and Expo, 2006 IEEE International Conference on*. IEEE, 2006, pp. 517–520.

[10] F. Deloule and C. Roche, "Ontologies and knowledge representation," in *Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on*, vol. 5. IEEE, 1995, pp. 3857–3862.

[11] J. F. Sowa, "Principles of semantic networks." Morgan Kaufmann, 1991.

[12] ——, "Conceptual graphs for a data base interface," *IBM Journal of Research and Development*, vol. 20, no. 4, pp. 336–357, 1976.

[13] M. Minsky, "A framework for representing knowledge," 1974.

[14] B. Nebel, "Frame-based systems," *MIT Encyclopedia of the Cognitive Sciences, MIT Press, Cambridge, MA*, pp. 324–325, 1999.

[15] C. Brewster, K. O'Hara, S. Fuller, Y. Wilks, E. Franconi, M. A. Musen, J. Ellman, and S. Buckingham Shum, "Knowledge representation with ontologies: the present and future," *IEEE Intelligent Systems*, pp. 72–81, 2004.

[16] H. J. Levesque, R. Reiter, Y. Lesperance, F. Lin, and R. B. Scherl, "Golog: A logic programming language for dynamic domains," *The Journal of Logic Programming*, vol. 31, no. 1, pp. 59–83, 1997.

[17] T. J. Ylonen, "Semantic network clustering influenced by index omissions," Mar. 4 2014, uS Patent 8,666,923.

[18] M. Noonan, "Mind maps: Enhancing midwifery education," *Nurse education today*, vol. 33, no. 8, pp. 847–852, 2013.

[19] C. Fellbaum, *WordNet*. Wiley Online Library, 1998.

[20] T. Van Gelder, "Argument mapping with reason! able," *The American Philosophical Association Newsletter on Philosophy and Computers*, vol. 2, no. 1, pp. 85–90, 2002.

[21] T. van Gelder, "Mapping the arguments in an academic text," 2011.

[22] C. Reed and G. Rowe, "Araucaria: Software for argument analysis, diagramming and representation," *International Journal on Artificial Intelligence Tools*, vol. 13, no. 04, pp. 961–979, 2004.
[23] K.-L. Cho and D. H. Jonassen, "The effects of argumentation scaffolds on argumentation and problem solving," *Educational Technology Research and Development*, vol. 50, no. 3, pp. 5–22, 2002.
[24] G. Fischer, A. C. Lemke, R. McCall, and A. I. Morch, "Making argumentation serve design," *Human–Computer Interaction*, vol. 6, no. 3-4, pp. 393–419, 1991.
[25] O. Scheuer, F. Loll, N. Pinkwart, and B. M. McLaren, "Computer-supported argumentation: A review of the state of the art," *International Journal of Computer-Supported Collaborative Learning*, vol. 5, no. 1, pp. 43–102, 2010.
[26] T. J. Berners-Lee, "The world-wide web," *Computer Networks and ISDN Systems*, vol. 25, no. 4, pp. 454–459, 1992.
[27] M. Crispin and K. Murchison, "Internet message access protocol-sort and thread extensions," RFC 5256, June, Tech. Rep., 2008.
[28] "http://www.ted.com/conversations."
[29] "https://yourview.org.au/."
[30] "http://debatedecide.org/."
[31] T. van Gelder, "Cultivating deliberation for democracy," *Journal of Public Deliberation*, vol. 8, no. 1, p. 12, 2012.
[32] "http://www.hotnews.ro/."

Automatic Concept-base creation method using document groups

Misako Imono[1], Eriko Yoshimura[2], Seiji Tsuchiya[2], and Hirokazu Watabe[2]
[1] Organization for Advanced Research and Education, Faculty of Science and Engineering, Doshisha University, Kyo-Tanabe, Kyoto, Japan
[2] Dept. of Intelligent Information Engineering and Sciences, Faculty of Science and Engineering, Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract - *This paper describe a method for creating concept-bases (CBs), which are knowledge bases comprising concepts that have been mechanically extracted from multiple sources and attributes that express their semantic features. In a CB, concepts are assigned attributes and weightings that express their importance. This means that that it is not necessary to define systematized relationships between concept and attributes, as is the case of a thesaurus, a semantic network, and/or an ontology. Concepts and attributes are defined based solely on the relationships that can be associated with each other. Using such definitions, a CB aims at including various meanings that human beings understand automatically based on words used, not simply definitions as described in dictionaries. The proposed method is capable of automatically building a CB from document groups such as newspaper articles, scientific papers, and Web articles that have not been analyzed in depth. Since this method is not restricted by document type, CBs can be built easily and automatically to suit the intended usage purpose.*

Keywords: Knowledge base, Concept-base, Degree of association, Association mechanism

1 Introduction

Currently, numerous challenges restrict the use of natural languages in information processing technologies in areas such as effective Web searches, recommendation systems, document classification, and robot communication. This is primarily because the natural language information contained in many documents can vary with the addition or subtraction of a single sentence or word. Accordingly, a different approach to defining word or phrase meanings is necessary when dealing with the information contained in natural language.

Human beings understand word, phrase, and sentence meanings flexibly expressed in natural language because their in-depth knowledge allows them to make “meaning” associations outside the definitions appearing in dictionaries. This includes matching contexts based on the other words, phrases, or sentences used. For example, humans readily and naturally find an association between the words “art” and “impression”, even though “impression” is not normally included in dictionary definitions of “art”.

General natural language processing defines clear relationships as the basic approach to understand word meanings. Thesauruses define the meaning of words by constructions that express super-sub relations and synonyms, while ontologies create models around a certain reality by defining parameters that indicate characteristic and clear relationships with a topic.

However, while significant amounts of knowledge have been systematized and utilized via such techniques, they remain insufficient to make rapid associations based on the example described above. It other words, it would be difficult to express the relationship between the word “art” and the word “impression” utilizing the knowledge that has been systematized in a thesaurus or an ontology. In a thesaurus, the higher node for “art” is “creation”, and the higher node for “impression” is “feeling”. Common nodes for these words exist only in the “abstract”.

While the vague relationships that permit humans to understand complex concept associations cannot be expressed systematically, such association can be identified if the appropriate sources and attributes are included in a concept base (CB)[1], which is a knowledge base consisting of natural word and phrase combinations built by focusing attention solely on their associate relationships.

A CB defines the meanings of various phrases called concepts that are expressed in natural language based on their relationships to other phrases called attributes. This means that neither labels nor categories indicating clear relationships need exist between concepts and attributes. If a source indicates that human beings detect relationships between “art” and “impression”, the CB will include “impression” in the attribute of the “art” concept without attempting to define the relationship between them. This CB structure allows meanings to be defined flexibly, much like humans do.

Therefore, in this paper, we describe a method that allows CBs to be built easily and automatically from document groups such as newspaper articles, academic papers, and Web articles that have not been analyzed in depth. This method is not restricted by document type.

2 Concept-Base

A CB is a knowledge base that defines words as concepts. A concept is defined in the following equation:

$$A = \{(a_1, w_1), (a_2, w_2), \ldots, (a_i, w_i)\}$$

where A is the concept label, a_i is the attribute, and w_i is the weight of the attribute. Table 1 shows specific examples of concepts.

Table 1: Specific example of concepts

Concepts	(Attribute, Weight)
Art	(Masterpiece, 0.34) (Impression, 0.23) (Ceramic Art, 0.12) (Sense of beauty, 0.08) …
Impression	(Sensitivity, 0.18) (Heart, 0.18) (Sense of beauty, 0.04) (Deep, 0.02) …
…	…

An attribute of a concept is called a first order attribute. In the CB, words defined by concepts also form attributes, which can then be used to derive other attributes. Attributes derived from attributes are called second order attributes of the original concept.

Concepts are defined by the synonymous and unforced "associative" relationships. Synonymous unforced relationships exist between the concepts and attributes in CBs. Synonymous relationship are not necessarily clear. However, relationships can be defined based on their "associative" level. This flexible semantic definition is a difference between WordNet.

3 Degree of Association

The Degree of Association (*DoA*)[2] quantifies the relationship between concepts by using attributes that characterize the chain-reaction structure of the CB. Table 2 shows specific *DoA* examples.

Table 2: Specific *DoA* example

Concept A	Concept B	*DoA*
Art	Artwork	0.15
	Impression	0.018
	Routine	0.0015

In this process, the relationship between multiple concepts is expressed quantitatively. The following shows the method used to calculate the *DoA* between Concept A and Concept B. This is defined as *DoA(A, B)*. For concepts A and B with primary attributes a_i and bi, weights u_i and v_j, and numbers of attributes L and M, are respectively ($L<M$), the concepts can thus be expressed as follows:

$$A = \{(a_1, u_1), (a_2, u_2), \ldots, (a_L, u_L)\}$$

$$B = \{(b_1, v_1), (b_2, v_2), \ldots, (b_M, v_M)\}$$

The degree of match (*DoM*) between concepts A and B *DoM (A,B)*, where the sum of the weights of the various concepts is normalized to 1, is defined as follows:

$$DoM(A, B) = \sum_{a_i = b_j} \min(u_L, v_j)$$

The *DoA* is found by calculating the *DoM* for all of the targeted primary attribute combinations, and then determining the relationships between them. Specifically, priority is given to the correspondence between matching primary attributes. For primary attributes that do not match, the correspondence is determined by maximizing the total *DoM*. This makes it possible to give consideration to the *DoA*, even for primary attributes that do not match perfectly. When the correspondences are thus determined, the *DoA(A,B)* between concepts A and B is as follows:

$$DoA(A, B) = \sum_{i=1}^{L} DoM(a_i, b_{xi}) \times \frac{(u_i + v_{xi})}{2} \times \frac{\min(u_i, v_{xi})}{\max(u_i, v_{xi})}$$

In other words, the *DoA* is proportional to the degree of identity of the corresponding primary attributes, the average of the weights of those attributes, and the weight ratios.

4 Automatic CB creation method

If an information source is defined by a direction-word and sentence pairs that express the meaning of the direction-word, it is possible to easily build a CB by defining the direction-word as the concept and the other words in the sentence as attributes. However, human beings constantly make various kinds of word association that are not found in dictionary definitions, and significant amounts of such information exist in miscellaneous sentences of various documents. In addition, the word knowledge that human beings use when making associations also exists in the miscellaneous sentences they utter naturally.

This paper proposes a method of extracting concepts and attributes automatically from document group information sources that have not been analyzed. Document group examples include newspaper articles, academic papers, and Web articles. Indeed, any document type can be used as an information source as long as it contains sentences that are suitable to the CB usage purpose.

4.1 Information source

In this study, a CB was automatically created using a year's worth of Japanese newspaper[3] issues as an information source. The field was not limited by article type and all newspaper articles, a total of 111,497, were examined.

4.2 Acquisition of concepts and attributes from co-occurrence range

A sentence (which is the range divided by periods within an article) is used to define the co-occurrence range. Concept and attributes pairs are acquired from this range. Figure 1 shows specific examples of co-occurrence range in article.

> …Managing Director of xxxx Corporation emphasized the importance of a precise judgment while slump of the economy was whispered. "I think that it is important to investigate a state of the economy through the conversation with the on-site person in charge."
> From slump of the demand in the iron manufacture industry, he look in retreat of the economy….

Figure 1: Specific example of co-occurrence range in article.

In figure 1, three sentences extracted from an article are shown. The underlined sentence is an example of the co-occurrence range from which words and phrases such as "economy", "slump", "whisper", "importance", "judgment", among others, are extracted. These words and phrases are defined as concepts and attributes for each. In the above example, the attributes "slump", "whisper", "importance", and "judgment" define the concept of "economy". This process performs such definition of concepts and attributes for all information sources. Figure 2 shows specific concept and attribute definition examples.

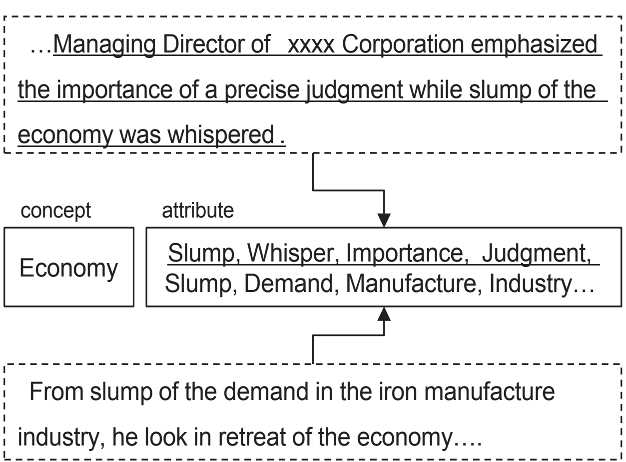


Figure 2: Specific example of definition of concepts and attributes.

The "economy" concept and its attributes are acquired from the first co-occurrence range (underlined portion). Afterwards, attributes such as "slump", "demand", "manufacture", and "industry" are added to the "economy" concept because they appear within the same co-occurrence range. After examining all articles for concept and attribute acquisition, a total of 316,319 concepts were extracted.

4.3 Weighting of attributes

TF-IDF[4], which is a commonly used technique for weighting words in documents for document searches, etc., is used to assign a weight that expresses the importance of each attribute. In this process, *TF* is the term frequency (the appearance frequency of the words) and *IDF* is the appearance inverse document frequency. These products calculate the weight of words. When *N* pieces of documents exist, the weight of word t which appears in document *d* is expressed as *TF(t,d)*, which is the frequency of *t* in document *d*, and *IDF(t)* is expressed as shown below:

$$IDF(t) = \log_2\left(\frac{N}{df(t)}\right) + 1$$

where *df(t)* is the total number of documents in which word *t* appears.

In attribute weighting, one concept is regarded as one document, and attributes of this concept are regarded as the words in the document. Therefore, the number of concepts (316,319) is regarded as the number of all documents *N*. The weight of attribute a_i of Concept A is calculated from $TF(a_i,A)$, which is the number of times that attribute a_i is collocated with Concept A, and $IDF(a_i)$ is calculated with the number of concepts that have a_i in its attribute. Figure 3 shows specific weighting examples.

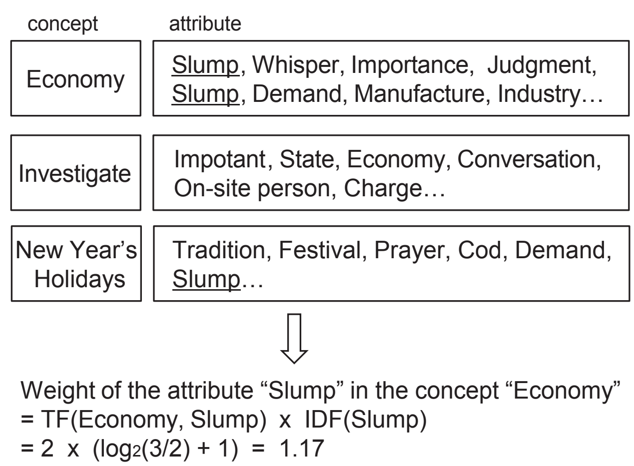


Figure 3: Specific example of weighting

This example presupposes that the total number of concepts is three. *TF(economy, slump)* becomes two, because the attribute "slump" appears twice in the concept of "economy". *IDF(slump)* becomes $log_2(3/2) +1 = 0.585$, because the number of concepts that have "slump" for an attribute is two and the total number of concepts is three.

4.4 Concept deletion via IDF threshold setting

Concepts can be deleted from the *IDF* threshold calculated in Section 4.3. by adjusting the IDF threshold setting. If the *IDF* for a concept is set too small, this concept will appear as an attribute in numerous other concepts. If this occurs, the concept appears in a vast number of co-occurrence ranges, and loses importance when defining other concepts. Examples of such concepts include the English words "the", "is", and "this". (Note that these examples are different from the Japanese words that fulfill similar roles that were eliminated when this proposed method was tested on a Japanese newspaper.) If the IDF for a concept is excessively large, it is thought that it is too high specific concept. It is thought that this process performs the CB refinement.

5 Evaluation and Validation

In this section, CB evaluation and validation methods are discussed. When evaluating a CB that has been refined by the process explained in section 4.4, the *IDF* threshold is set to several phases and the CB is evaluated for each phase.

5.1 Evaluation Method

Evaluations are carried out using an *X-BC* evaluation set. This evaluation set is composed of three concepts, X, B, and C. In Table 3, specific *X-BC* evaluation set examples can be seen. Concept B entries have some relations with Concept X entries, but Concept C entries do not.

Table 3: Specific X-BC evaluation set examples

Concept X	Concept B	Concept C
Art	Impression	Both
Situation	Consultation	Error
Write	Paper	Space
Bad Crop	Field	Keep Up
Festival	Lively	Confusion
Tea	Long-established Shop	Pail
…	…	…

This evaluation method calculates *DoA(X,B)* and *DoA(X,C)*, after which the *DoA* values are compared. If a CB is built correctly, *DoA(X,B)* should have a value that is bigger than *DoA(X,C)* because human beings can detect a relationship between concept X and concept B. Therefore, answers are considered correct when *DoA(X,B)* is bigger than *DoA(X,C)*. In this study, the total number of evaluation sets made by plural people is 500.

5.2 Evaluation and Validation at each threshold

First, an evaluation is carried out on the result of the IDF upper limit threshold. This evaluation deletes any concepts with IDF values larger than the threshold setting. Table 4 shows the result of this evaluation. In addition, it should be noted that the correct answer rate is calculated by using only the sets that contains all three (X, B and C) concepts while remaining within the threshold. The number of existing sets expresses the number of sets that contain all the (X, B and C) concepts. A set ration is a ration of number of the left sets.

Table 4: Evaluation result with IDF upper limit threshold

Threshold	Correct answer rate (%)	Number of existing sets	Set ration
14	**59.6**	319	63.8
13	59.4	318	63.6
12	59.4	310	62.0
11	59.2	282	56.4
10	56.2	217	43.4
9	42.4	118	23.6
8	29.1	55	11.0
7	22.2	18	3.6

In the case of an upper limit threshold of 14, there were no deleted concepts. Thus, the correct answer rate for the CB itself (for one newspaper year) was 59.6%. In addition, since the correct answer rate for all other threshold was less than 59.6%, no refinement effect was seen by application of upper limit threshold concept deletion. Next, an evaluation was carried out for the *IDF* lower limit threshold result. This evaluation deletes *IDF* concept values that are lower than the minimum threshold. Table 5 shows these evaluation results.

Table 5: IDF lower limit threshold evaluation results

Threshold	Correct answer rate (%)	Number of existing sets	Set ration
1	59.6	319	63.8
2	61.8	319	63.8
3	65.4	315	63.0
4	70.5	305	61.0
5	78.4	241	48.2
6	79.7	128	25.6
7	64.3	56	11.2
8	31.6	19	3.8

The correct answer rate reached the highest level when the lower limit threshold was set to 6. However, these rates cannot be compared because the number of existing sets is different depending on threshold. Accordingly, evaluations were carried out using the 128 evaluation sets that remained in the case of a lower limit threshold of 6. Table 6 shows the evaluation results.

Table 6: Evaluation result with 128 evaluation sets

Threshold	Correct answer rate (%)
TH	58.6
1	69.5
2	68.8
3	72.7
4	78.1
5	82.8
6	79.7

As a target for comparison with the correct answer rate, a TH that resembles the degree calculation technique based on the distance on the thesaurus is utilized[5]. In the case of a lower limit threshold of 5, the evaluation result is 82.8%, which is the highest correct answer rate. It should be noted that all correct answer rates are higher than TH.

Next, the IDF value distribution was validated. To accomplish this, the distribution of all concepts (316,319) is first investigated. Table 7 shows the IDF distribution of all concepts.

Table 7: IDF distribution (All concepts)

IDF level section	Number of concepts	Accumulation rate (%)
0-1	0	0.00
1-2	2	0.00
2-3	33	0.01
3-4	85	0.04
4-5	542	0.21
5-6	1832	0.79
6-7	4634	2.25
7-8	11248	5.81
8-9	24081	13.42
9-10	51335	29.65
10-11	126690	69.70
11-12	83553	96.12
12-13	12010	99.91
13-14	274	100.0

In Table 7, the IDF level column contains “greater than left value, less than right value”. For example, row “1-2” means that the value is greater than 1 but less than 2 of the IDF. In addition, the distribution of concepts in the evaluation sets (Concepts X, B and C) is investigated. Table 8 shows the IDF distribution of these evaluation sets.

Most concepts belonging to the IDF column are greater than 11 and smaller than 10. On the other hand, the peaks of the three concepts (X, B and C) exist from IDF level 6 to 9. From this, it can be seen that there is difference in word association trends between newspaper usage and human beings. These results indicate that the low IDF value is a word (concept) that appears in numerous articles. From Table 7 and Table 8, concepts that a human being can associate (concepts in evaluation sets) appear with low IDF values in the CB. This result may be due to the fact that human beings use words in conversational contexts that would be unclear and difficult to understand in newspaper articles. All CB evaluation results exceed the distance provided by a thesaurus, so it can be said that our proposed CB creation functions well. The tendency for a gap to be included in the provided concepts may dissolve when using a source of information that more closely conforms to the CB usage purpose.

Table 8: IDF distribution (Evaluation sets)

IDF level section	Number of Concept X	Number of Concept B	Number of Concept C
0-1	0	0	0
1-2	0	0	0
2-3	0	3	2
3-4	0	12	3
4-5	4	67	17
5-6	34	129	56
6-7	49	120	62
7-8	86	74	58
8-9	76	44	78
9-10	82	21	68
10-11	44	11	48
11-12	14	7	18
12-13	5	0	3
13-14	0	0	1

6 Conclusion

This paper describes a method for creating CBs easily and automatically from document groups such as newspaper articles, academic papers, and Web articles that have not been analyzed. In a CB, the meanings of various natural language phrases (called concepts) are associated with other phrase sets (called attributes) in order to detect indirect relationships. In this paper, one sentence (a range divided by periods within a newspaper article) was used to define a co-occurrence range, and concept and attributes pairs were acquired from within this range. In our CB evaluations using the proposed method, a correct answer rate is 82.8% was obtained, which is higher than resemblance degree calculation technique provided by a thesaurus.

It is desirable to make CBs using information sources that are applicable to the CB usage purpose because differences occur between the IDF distribution values of concepts extracted from newspaper articles and evaluation sets made from human conversation. However, it is clear that, by choosing an appropriate information source, the method proposed in this paper would permit the creation of CBs suitable to their usage purposes.

Acknowledgment

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

References

[1] K. Kojima, H. Watabe, and T. Kawaoka. "A Method of a Concept-base Construction for an Association System: Deciding Attribute Weights Based on the Degree of Attribute Reliability"; Journal of Natural Language Processing, Vol.9 No.5, pp.93—110, 2002.

[2] H. Watabe and T. Kawaoka. "The Degree of Association between Concepts using the Chain of Concepts"; Proc. of SMC2001, pp.877—881, 2001.

[3] The Mainichi Newspapers. "CD- Mainichi '95 data collection", 1995.

[4] T. Tokunaga. "*Jyouhou Kensaku To Gengo Syori*"; University of Tokyo Press, 1999.

[5] S. Ikehara, M. Miyazaki, S. Shirai, A. Yokoo, H. Nakaiwa, K. Ogura, Y. Ooyama. "*Nihongo Goi Taikei*"; *Iwanami Shoten*, 1997.

[6] M. Nagao. "*Iwanami Kouza SoftWare Kagaku 15 Sizen Gengo Syori*"; *Iwanami Syoten*, 1996.

Network-based relevance relationship generating for empirical engineering knowledge

Xinyu Li[1], Zuhua Jiang[1], Bo Song[1], and Lijun Liu[1]
[1]Department of Industrial Engineering and Management, Shanghai Jiao Tong University, Shanghai 200240, PR China

Abstract - *Cognizing and utilizing the relevance relationship of knowledge are evitable issues for enterprises and organizations to maintain preponderance. However, the grueling analysis of relevance relationships, especially for empirical knowledge in engineering field, has been manually processed by domain experts. In this paper, an automatic network-based relevance relationship generating method is proposed for representing the relations among empirical engineering knowledge (EEK) and assisting in comprehending structure of the engineering domain. Two phases, EEK elicitation and formalization as well as EEK networks foundation, are included in the generating method and implemented with natural language process, sematic similarity calculation and fuzzy neutral network prediction techniques. Relevance network of empirical knowledge in computer-aided design (CAD) is constructed and verified by domain experts and long-term practitioners. Experimental results show that the proposed method outperforms the former approaches in feasibility and effectiveness, and thereby offer a way for further understanding the evolution course of EEK.*

Keywords: Empirical engineering knowledge; network; relevance relationship; knowledge representation; data visualization

1. Introduction

In the era of knowledge-driven economy, emerging knowledge based on the long-existing concepts, techniques, methodologies, experiences and activities, as well as the best management and utilization of it, is the key to maintain the competitiveness preponderance of the organizations and enterprises in creativity and adaptability [1]. And with the development of the Internet and information technology, knowledge is presented with rapid transmission and multiple interdisciplinary, which further promotes the complexity of the relevance relationships among knowledge [2]. How one piece of knowledge links to others and what potential information is concealed behind that relationships are two urgently answered and increasingly evitable problems in the field of knowledge management. The answers of the two problems could help the intellectual workers to find a breakthrough for the facing questions, obtain a comprehensive cognition for the engineering field, and grasp the direction of the future research. However, current analysis of the relevance relationship of knowledge mainly depends on the domain experts, which may be difficult to be comprehensive and objective because of the intrinsic time-delay and capacity-limitation of the experts. Such defects are more critical in feasibility and effectiveness of the expert-relied analysis when analyzing a proliferating field with huge amount of information and knowledge. Therefore, the adaptation of an automatic and scientific method for filtering colossal information and mining persuasive relationships should become a task of top priority.

New ideas and concepts are often the consequences of the original ones, which is the fundamental relevance mechanism in knowledge development [3]. In the field of academia, the researchers used metrology method to collect and census the keywords, abstracts and cited references, and, hence, speculated the relevance relations [4-7]. However, the method of metrology may not achieve a satisfied result when it is deployed into the field of engineering, especially faced with the empirical engineering knowledge (EEK). Two main reasons lead to this unavailability: (1) Although there is some clear-coded and correct-recognized knowledge in engineering field, such as the formulas, standards and specifications, most of the engineering knowledge is derived from the solutions of the actual engineering missions and presented as non-canonical knowledge with scenario dependency, concepts ambiguity and correctness uncertainty [8, 9]; (2) The precise expression of the subject knowledge via several explicit concepts and propositional logics is the fundamental of the literature statistic and quantitative analysis, while the concepts and logic relations in EEK is concealed in the records formed by natural language and incapable to elicit directly [10].

Some scholars dealt with relationships with the network methods. Pyka A. et al. [11] simulated the innovation in modern knowledge-based industries by an agent-based network model. Liu J. et al. [12] presented two knowledge-generation models via the hyper-network and analyze the distribution of knowledge stock, which could be helpful for deeply understanding the scientific research cooperation. Lee K.M. et al. [13] expressed the knowledge in Bayesian networks and proposed an agent framework. Although such network-based methods put forward some new ideas for detecting knowledge relationship, only simulation models were established in their works, rather than the detailed analysis of the specific field with large amount of knowledge, which led to a questionable feasibility.

Considering the advantages and shortages of all above research works, this paper proposes an automatic two-phase network-based relevance relationship generating method combined with natural language process, sematic similarity calculation, and fuzzy neutral network prediction based on the

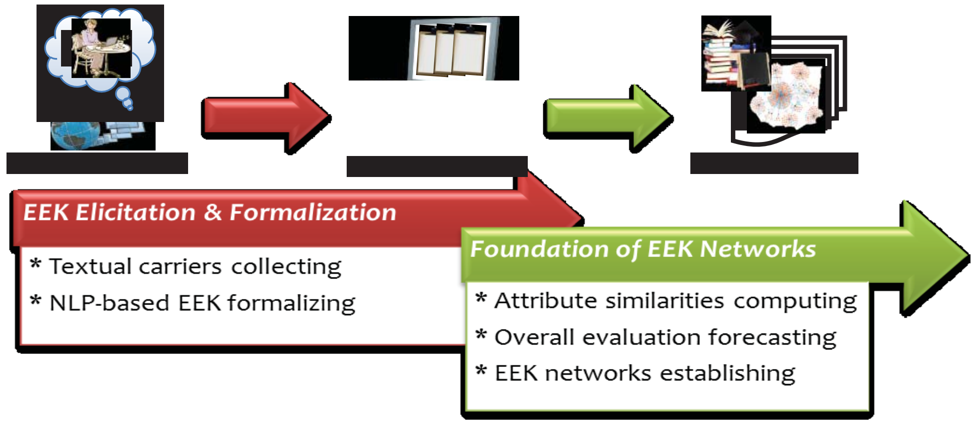


Fig. 1 Framework of network-based relevance relationship generating method

elicitation and formalization of EEKs. With few human interventions in annotating the samples, the proposed method performs well in discovering the numerical and semantic relevance relationships behind huge amount of existing actual EEKs. The feasibility and advancement of the proposed method are testified with the typical EEKs of computer-aided design (CAD) from 2011 to 2015, and verified by some domain experts and long-term practitioners.

The remainder of this paper is organized as follows. Section 2 designs the general framework of the proposed generating method. The elicitation and formalization of EEKs is presented in section 3. Section 4 details the foundation of EEK networks with calculation of attribute similarities and fuzzy neutral network prediction. The example of using the proposed method to generate the relevance network of EEKs originated from accomplishing of engineering design mission using AutoCAD software is presented in section 5. Last section includes the comparison with former works and concludes the paper with some possible improvements.

2. Framework of proposed method

Oriented to generate the relevance relationships of empirical engineering knowledge (EEK) automatically, this paper proposed a two-phase generating method based on natural language process (NLP), sematic similarity calculation, and fuzzy neutral network (FNN) prediction. Figure 1 presents the framework of proposed method.

(1) Eliciting and formalizing EEKs: Collected from threads in professional virtual communities, meeting notes, email exchanges, success or failure cases, revision history of a Wiki page and other electronic documents, available textual carriers of EEK are gathered and analyzed. Seven attributes of EEK, namely Engineering Problem, Problem Context, Problem Solution, Feature Association, Effectiveness, Contributor and Time, are extracted with part-of-speech tagging, sentences parsing, word weights computing and other natural languages process (NLP) techniques, and hence the form of EEK is constructed with $EEK = < EP, PC, PS, FA, E, C, T >$.

(2) Founding EEK networks: Similarity of each pair of attributes in two EEKs is calculated with their numerical relationships and semantic relationships. Based on seven attributes similarities, an overall evaluation of EEK relationship is forecasted by T-S Fuzzy Neutral Networks (T-S FNN). The network of EEK is founded with the EEK pairs whose strength of relationships over a threshold, and EEK networks are saved with undirected weighted graphs, and visualized by data visualization software.

3. Elicitation and formalization of EEKs

3.1 Definition and carrier of EEKs

In the enterprises, engineering experiences are of significant value for innovative design and decision-making process, possessing an indispensable part of the corporate knowledge base. Many scholars devoted themselves to the definition of empirical engineering knowledge, as well as the subsequent empirical knowledge acquisition and reuse [8, 9, 14-17]. Concluding from the related research works, the empirical engineering knowledge (EEK) processed in this paper is defined as *a consequence of probable association and extension of engineering concepts and engineering objects under specific constraints of engineering scenarios, obtained through repeated observation and practice of engineering technicians in long-term engineering activities*. Composed in personalized and ambiguous natural language, the specific engineering problem, the problem context, the solution of the problem, and the feature association with other EEKs induced from different scenarios, are described in an EEK.

In widely adopted collaborative network working environment, increasing number of EEKs are recorded and spread with the form of electronic documents existing inside and outside the enterprises, such as virtual community Q&As, meeting notes, success or failure cases, revision history of a Wiki page or other electronic documents. In these documents, the major ingredients are the textual carriers of a specific EEK happened in an actual engineering mission. Several documents also use images, videos, audios, program files and mathematical models for implementation, which is not considered in this paper. Figure 2 shows a textual EEK carrier downloaded from a professional CAD virtual community.

These textual carriers vary in the word choice and sentence building, but they all have a similar generative process. Beginning with one or several key VO structures describing the propose or the topic of the engineering problems, such as "fix layer" in the title of the carrier presented in figure 2, the question askers descript the engineering scenarios around these key VO structures, and will receive growingly admissive solutions in the interactions with the respondents. When the last interaction is completed, the EEK is generated with all its attributes recorded.

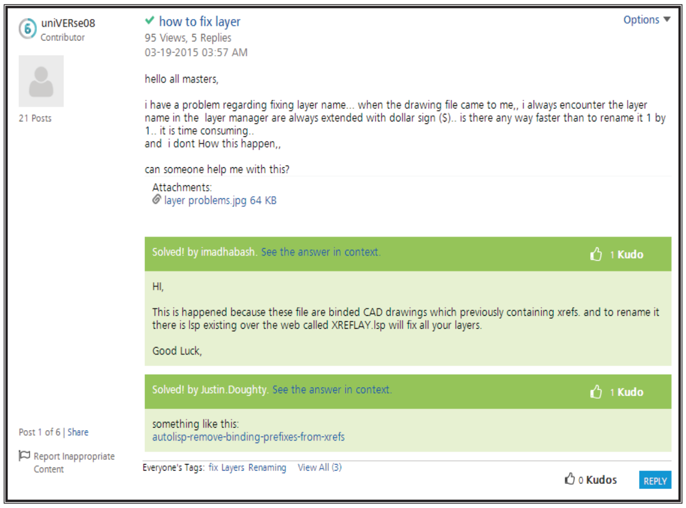


Fig.2 An example of textual EEK carrier

3.2 NLP-based EEK formalization

Certainly, the generation of textual EEK carriers is not always as perfect as the above process described. Since the askers and respondents are often unable to grasp the key to the engineering problems, there may be plenty of “noise” formed from the unrelated concepts existing in the discussion, and the same concepts may be stated in different forms by diverse participants. What’s more, for a textual record that contains no succinct topic, for example, the non-topic meeting discussion or Wiki pages revision, the key VO structures are not directly given and should be summarized from the comprehensive understanding of the whole context, which all bring difficulties for computer automated processing.. Therefore, the textual carriers of EEK should be processed with natural language processing (NLP) techniques and generated in appropriate structures before they are used for relevance relationship generation.

According to the definition in section 3.1, this paper uses seven corresponding attributes to structure an EEK, namely *EEK* = < *EP, PC, PS, FA, E, C, T* >. The content of each element is listed as follows:

(1) *EP* (Engineering Problem) proposes a specific engineering problem.

(2) *PC* (Problem Context) descripts the background informations and constrains of this *EP*.

(3) *PS* (Problem Solution) shows the empirical solution.

(4) *FA* (Feature Association) lists the relationships and corresponding strengths between other *EEK*s.

(5) *E* (Effectiveness) evaluates the fitness of *PS* in this *EP* under such *PC*.

(6) *C* (Contributor) collects all the participants in the generative process of this *EEK*.

(7) *T* (Time) records the time when this *EEK* finally formed.

In addition, for each piece of *EEK*, a unique *EEKID* is generated for indexing.

In order to filter out the unrelated informations and elicit the main attributes, Song et al. [18] adopted a method that used the Conditional Random Field (CRF) text classification techniques to label the role of each sentence and elicited problem objectives and context constrains of the empirical knowledge. Shah C. et al. [19] extracted various features from questions, answers, and the posters to select best answers through a prediction model constructed by Logistic Regression. This paper combines two methods and formalizes the EEKs.

Using CRF to label the role of each sentence (*QUESTION / CONTEXT / ANSWER / PLAIN*) in the textual carriers is the first step. Then we elicit the VO structures from the sentences labelled *QUESTTION* and use them as *EP* in an *EEK*. The singularized noun-phrases in the sentences labelled *QUESTTION* or *CONTEXT* will form *PC* in this *EEK*, and all the noun-phrases are converted into lowercase and repeated ones are eliminated. For the *ANSWER* sentences, we organize them with their original posters and evaluate them using the feature-based Logistic Regression Model. The highest regression value is chosen to be *E* of *EEK* and the set of singularized noun-phrases in corresponding *ANSWER* sentences will be *PS*. *C* and *T* of *EEK* can be obtained directly from the textual carriers, while *FA* is left blank at present and will be filled in when calculating the EEKs attribute similarities. An example of structured EEK elicited from figure 2 is shown in table 1.

Table 1 A formalized EEK elicited from figure 2

EEKID = 2494
EP = { fix layer; fix layer name } (**Noun-Phrase**: { layer, layer name});
PC = { layer name, drawing file, layer manager, dollar sign};
PS = { file, cad drawing, xref, lsp, web, xreflay.lsp, layer};
FA = { };
E = 0.645;
C = { universe08, imadhabash, beekeecz, justindoughty };
T = 2015.302

4. Foundation of the EEK networks

4.1 Similarity calculation for EEK attributes

The evaluation of the relevance relationships between a pair of EEKs is the basis of establishing EEK networks. Due to the formalization of EEKs, EEK could be expressed with seven attributes. Therefore, the evaluation will be finished with similarities of each pair of attributes. For two EEKs, $EEK_1 = < EP_1, PC_1, PS_1, FA_1, E_1, C_1, T_1 >$ and $EEK_2 = < EP_2, PC_2, PS_2, FA_2, E_2, C_2, T_2 >$, seven similarities, *EPSim, PCSim, PSSim, FASim, ESim, CSim* and *TSim*, will be calculated.

On the basis of article [18], seven similarities are calculated respectively as follows:

$$EPSim=\begin{cases} 1 & EP.I \\ 0 & EP.II \\ \frac{1}{2}\left(\frac{\sum_{NP_i\in EP_1}\max_{\forall NP_j\in EP_2} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_i\right)}+\frac{\sum_{NP_j\in EP_2}\max_{\forall NP_i\in EP_1} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_j\right)}\right) & EP.III \end{cases} \quad \ldots\ldots\ldots. (1)$$

EP.I When EP_1 and EP_2 have at least one same VO;
EP.II When EP_1 or EP_2 is empty;
EP.III When EP_1 and EP_2 don't have any same VO, the similarity between EP_1 and EP_2 is calculated by all the noun-phrases NP_i of EP_1 and NP_j of EP_2;

$$PCSim=\begin{cases} 0 & PC.I \\ \frac{1}{2}\left(\frac{\sum_{NP_i\in PC_1}\max_{\forall NP_j\in PC_2} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_i\right)}+\frac{\sum_{NP_j\in PC_2}\max_{\forall NP_i\in PC_1} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_j\right)}\right) & PC.II \end{cases} \quad \ldots\ldots\ldots. (2)$$

PC.I When PC_1 or PC_2 is empty;
PC.II The similarity between PC_1 and PC_2 is calculated by all the noun-phrases NP_i of PC_1 and NP_j of PC_2;

$$PSSim=\begin{cases} 0 & PS.I \\ \frac{1}{2}\left(\frac{\sum_{NP_i\in PS_1}\max_{\forall NP_j\in PS_2} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_i\right)}+\frac{\sum_{NP_j\in PS_2}\max_{\forall NP_i\in PC_1} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_j\right)}\right) & PS.II \end{cases} \quad \ldots\ldots\ldots. (3)$$

PS.I When PS_1 or PS_2 is empty;
PS.II The similarity between PS_1 and PS_2 is calculated by all the noun-phrases NP_i of PS_1 and NP_j of PS_2;

For FA similarities of a pair of EEKs, two assumed relationships are considered:

- Trigger Relationship: if EEK_1 will trigger EEK_2, PC_2 of EEK_2 contains EP_1 of EEK_1
- Solved-by Relationship: if EEK_1 is solved by EEK_2, PS_2 of EEK_2 contains EP_1 of EEK_1

The similarity of EP and PC or PS will measure the two relationships, thus determining the value of *FASim*, and the value of attribute FA is filled in.

$$FASim=\max\left\{\begin{matrix} Trigger\left(EP_1,PC_2\right),Trigger\left(EP_2,PC_1\right), \\ Solve\left(EP_1,PS_2\right),Solve\left(EP_2,PS_1\right) \end{matrix}\right\} \quad \ldots\ldots\ldots. (4)$$

$$Trigger\left(EP_1,PC_2\right)=\begin{cases} 0 & Trigger.I \\ \frac{1}{2}\left(\frac{\sum_{NP_i\in EP_1}\max_{\forall NP_j\in PC_2} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_i\right)}+\frac{\sum_{NP_j\in PC_2}\max_{\forall NP_i\in EP_1} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_j\right)}\right) & Trigger.II \end{cases} \quad (4.1)$$

Trigger.I When EP_1 or PC_2 is empty;
Trigger.II The Trigger-relevance is calculated by all the noun-phrases NP_i of EP_1 and NP_j of PC_2;

$$Solve\left(EP_1,PS_2\right)=\begin{cases} 0 & Solve.I \\ \frac{1}{2}\left(\frac{\sum_{NP_i\in EP_1}\max_{\forall NP_j\in PS_2} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_i\right)}+\frac{\sum_{NP_j\in PS_2}\max_{\forall NP_i\in EP_1} NPSim\left(NP_i,NP_j\right)}{Count\left(NP_j\right)}\right) & Solve.II \end{cases} \quad .. (4.2)$$

Solve.I When EP_1 or PS_2 is empty;
Solve.II The Solved-by-relevance is calculated by all the noun-phrases NP_i of EP_1 and NP_j of PS_2;

Function *Count*(*NP*) is the number of non-repetitive NPs in EP, PC or PS separately. And noun-phrase similarity is computed as:

$$NPSim\left(NP_1,NP_2\right)=\begin{cases} 2^L & NP.I \\ 1 & NP.II \\ \max_{Word_i\in NP_1,Word_j\in NP_2} WSim\left(Word_i,Word_j\right) & NP.III \end{cases} \quad (5)$$

NP.I When words in the corresponding positions of the two phrases are the same; L is the phrase length;
NP.II When some words of the two phrases are the same;
NP.III When all the words of the two phrases are different, the similarity between NP_1 and NP_2 is calculated by all the words $Word_i$ of NP_1 and $Word_j$ of NP_2;

Word similarity is generated with the normalized point wise mutual information:

$$WSim\left(Word_1,Word_2\right)=\log\frac{p\left(Word_1,Word_2\right)}{p\left(Word_1\right)p\left(Word_2\right)}/\log\left|D\right| \quad \ldots (6)$$

$p(Word_1)$ is the proportion of EEKs that contain $Word_1$, and $p(Word_1, Word_2)$ is the proportion of EEKs that contain $Word_1$ and $Word_2$ simultaneously. Since the similarity of two different words should not exceed the similarity of two same words, we use the largest possible value of point mutual information log $|D|$ to normalize it.

$$ESim=\begin{cases} 0 & \text{if } E_1=E_2=0 \\ \frac{\min\left(E_1,E_2\right)}{\max\left(E_1,E_2\right)} & \text{else} \end{cases} \quad \ldots\ldots\ldots\ldots. (7)$$

$$CSim=\begin{cases} 0 & \text{if } C_1\cup C_2=\varnothing \\ \frac{2Count\left(C_1\cap C_2\right)}{Count\left(C_1\right)+Count\left(C_2\right)} & \text{else} \end{cases} \quad \ldots\ldots\ldots. (8)$$

Function *Count*(*C*) is the number of non-repetitive contributors in contributor set *C*.

$$TSim=1-\frac{\left|T_1-T_2\right|}{T_0} \quad \ldots\ldots\ldots\ldots\ldots (9)$$

T_0 in Eq. (9) is the maximum lag of time among all possible EEK pairs.

4.2 Fuzzy evaluation of overall relationship

Seven similarities are calculated with the Eq. (1-9), and thereby available for evaluation of overall relationship of EEK pairs. A commonly used linear weighted sum method is easy to compute the overall relationship, but the result may be not cogent for two reasons. One is that the emphasis of attributes is uncertain and weight of each of seven attributes is hard to decide; the other one is that the exact numerical values will not influence the structure of the networks and the precision of the evaluation is of slight significance.

Actually, this is an overall fuzzy evaluation with multi-input and single-output, commonly appeared in expert decision-making in geological structure, management level and finance risk assessments. Some scholars used T-S Fuzzy Neutral Networks (T-S FNN) method and achieved some good results [20-23]. T-S FNN is a method that combined supervised machining-learning and fuzzy logics and its architecture is shown in figure 3.

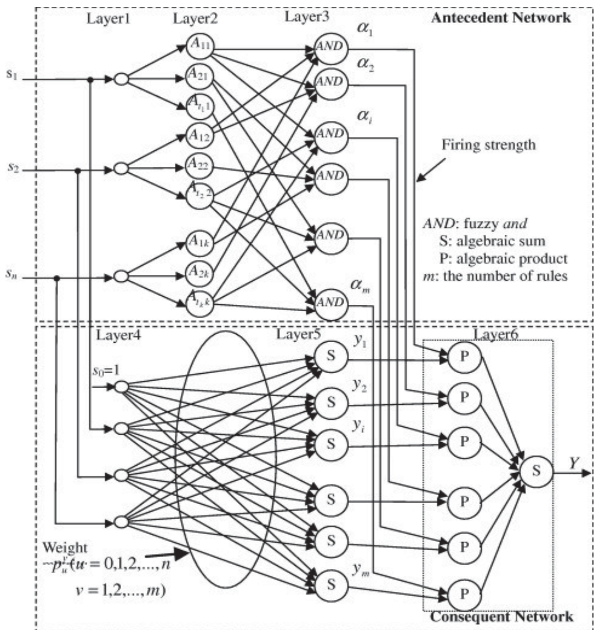


Fig.3 Architecture of the T-S FNN [23]

Layer 1 and 4 are the input layers, both receiving the values of the attributes except that a constant $s_0 = 1$ is input additionally in layer 4. Layer 2 uses Gaussian-shaped membership function $A(s) = \exp\left(-(s-c)^2 / 2\sigma^2\right)$ to fuzzificate the input attribute values, where s is the input value and c, σ are the shape parameters decided by the number of input attributes and output grades. The strength of each fuzzy rule is calculated in layer 3 and becomes the output of each neuron. Layer 5 summarizes the consequent of each fuzzy law with the function $y_j = p_0^j + \sum_{i=1}^{n} p_i^j s_i \quad (j = 1,2,\ldots$, where p_i^j is the consequence parameter and varies in the iterations. Defuzzification and normalization of all the consequents is completed in layer 6, where the overall evaluation is generated.

4.3 Establishing the EEK networks

The well-trained T-S FNN will forecast the strength of relevance relationship in any pairs of EEKs, and an undirected weighted graph $UWG = <V, E>$ representing the EEK networks is constructed in consequence. The collection of EEKs forms the vertex set V. If the relationship strength of two EEKs equals or exceeds a certain grade $Grade_{threshold}$, two corresponding vertexes are connected in the graph and the weight of the edge is their strength. The relevance network could be visualized with data visualization software for a better cognition of the structure of field.

With the relevance network, it is intuitionistic for domain experts or practitioners to find the relevance relationship among EEKs. The network will also be of significant assistance for detecting the dispersed key EEKs or concentrating on groups of several intensively related EEKs.

5. Case study

In this section, we evaluated the feasibility of proposed relevance relationships generating method. We ran the Java code on a Core i5 2.5 GHz PC with 8 GB memory, and visualized the result with Gephi software.

From three professional virtual communities, *forums.autodesk.com*, *www.cadtutor.net* and *www.cadforum.cz*, 2501 EEKs of accomplishing computer-aided engineering design missions using AutoCAD software were elicited and formalized, ranging from January 2011 to March 2015. Choosing CAD as the evaluation of proposed generating method has three reasons: (1) Computer-aided design is a typical knowledge intensive mission in the engineering field, which receives frequent attention from the engineering technicians as well as the knowledge management practitioners; (2) AutoCAD software, published by AUTODESK Corporation, is the most popular CAD tool in worldwide, and its application has been discussed deeply, widely and perennially by huge amount of CAD workers; (3) CAD experts and long-term CAD practitioners are available for analyzing the proposed process and assessing the experiment result.

We selected 320 pairs of EEKs randomly and invited 3 domain experts to scoring the pairs with a scale of 1 to 5 according to the relevance relationship in the pairs. 308 valid evaluated samples were returned and 12 samples were deleted because of the significant difference of the opinions. Actual score was determined by evaluation of the majority of experts. Table 2 lists part of the sample pairs and theirs attribute similarities.

Table 2 The attribute similarities and experts' evaluation of valid samples of EEK pairs (an illustrative part)

EEK pair		**Attribute similarities**							**Experts' evaluation**			
EEK1	EEK2	EP	PC	PS	FA	E	C	T	Rank1	Rank2	Rank3	Actual
7	586	1.00	0.30	0.29	0.54	0.94	0.00	0.97	5	5	4	5
29	1655	0.00	0.15	0.22	0.20	0.96	0.00	0.68	1	1	1	1
34	2087	0.37	0.42	0.16	0.50	0.84	0.33	0.90	2	2	3	2
58	294	0.51	0.35	0.12	0.37	0.98	0.00	0.89	4	3	3	3
66	217	0.31	0.48	0.38	0.48	0.99	0.40	0.89	3	3	3	3
71	1280	0.00	0.12	0.00	0.10	0.00	0.25	0.47	1	1	1	1
88	346	0.75	0.18	0.29	0.45	0.89	0.00	0.67	4	4	4	4

250 samples train-data and 58 samples test-data were divided and used for training of T-S FNN. In the FNN, there

were 7 input nodes, 5 output nodes and 14 fuzzy rule nodes considering the number of EEK attributes and scoring scales. Learning constant and Momentum constant both were 0.5, and the iteration number was 50000. The result of train-data and test-data is shown in figure 4.

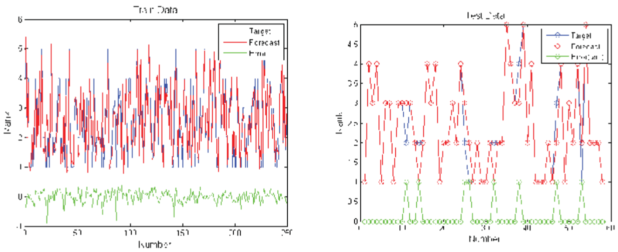


Fig. 4 The network output of training data and test data

Without the consideration of grade error, the precision of the test-data result is 86.2%, and it escalates to 100% when the admissible error is set to ±1 grade. Therefore, the relationship forecast through T-S FNN is considered reliable. The well-trained T-S FNN is utilized for evaluating any EEK pairs and forming the EEK networks. Figure 5 presents part of the undirected weighted graph of EEK networks when $Grade_{threshold}$ was set to 3. Nodes intensively located in the center illustrate the key EEKs that constructed relevance relationship with a lot of other EEKs, while ones in the margin represent the isolated EEKs with few relationships.

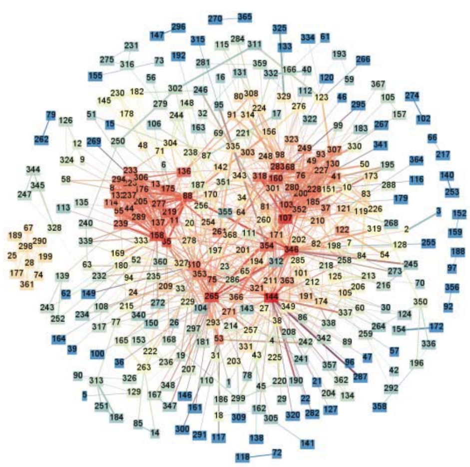


Fig.5 The undirected weighted graph of the CAD EEK relevance network (an illustrative part)

Delphi method is a useful tool for acquiring a consensus-based opinion from a panel of experts. 20 randomly selected relevance relationships in the network as well as the original EEKs textual carriers of referred EEKs were sent to 15 domain experts and long-term CAD practitioners to evaluate the validity. The performance was assessed by the questionnaire referred to Chen Y. et al. [15]. In the investigation with experts and practitioners, approximate 90% of them were satisfied with the generated relevance network in managing the empirical engineering knowledge, comprehending the domain structure of computer-aided design and mining the CAD knowledge relevance relationships with less time. Table 3 and Table 4 present the questionnaire and the result of the respondents.

Table 3 Questionnaire for assessing the performance

1. The degree to which the *CAD EEK Relevance Network* helps CAD experts and practitioners in organizing empirical CAD knowledge.
A. Very useful B. Useful C. No comment D. Useless E. Very useless
2. The degree to which the *CAD EEK Relevance Network* helps CAD experts and practitioners in understanding the structure of CAD field.
A. Very useful B. Useful C. No comment D. Useless E. Very useless
3. The degree to which the *CAD EEK Relevance Network* helps CAD experts and practitioners in mining accurate CAD knowledge relevance relationships.
A. Very useful B. Useful C. No comment D. Useless E. Very useless
4. The degree to which the *CAD EEK Relevance Network* helps CAD experts and practitioners in saving time for generating empirical CAD knowledge relationships.
A. Very useful B. Useful C. No comment D. Useless E. Very useless

6. Discussion and concluding remarks

This work developed a network-based generating method for mining relevance relationships of empirical engineering knowledge. With natural language process, sematic similarity calculation, and fuzzy neutral network prediction, the relevance network were built on the basis of the formalized elicited EEKs with seven attributes of EP, PC, PS, FA, E, C and T. The establishing of the relevance network for the field of computer-aided engineering design, as well as the assessing of the network by CAD domain experts and practitioners, has shown the feasibility and effectiveness of the proposed generating method.

Since the generating method depends on encoded empirical knowledge instead of a shared dataset, it is impossible for us to compare the proposed method with former related research work quantitatively. However, in qualitative aspects, network-based relevance relationship generating method outperforms the methods represented in articles [4-7] because of the sufficient consideration of the ambiguity and individuality of empirical engineering knowledge, and the complete combination of numerical relationships and semantic relationships among EEKs. The established relevance network using our generating method is also more persuasive than the former works. Successfully

Table 4 Respondent assessment result for the proposed modelling method

Question	Very useful	Useful	No comment	Useless	Very useless	Total	Satisfaction
Q(1)	9	5	1	0	0	15	93.3%
Q(2)	7	6	1	1	0	15	86.7%
Q(3)	6	6	2	1	0	15	80.0%
Q(4)	10	5	0	0	0	15	100.0%
Average	8.0	5.5	1.0	0.5	0	15	90.0%

operating with plenty of pragmatic empirical knowledge in actual engineering field and verified by senior experts and practitioners, the relevance network will be more cogent and obvious than the simulation models in articles [11-13].

There are several possible improvements for our methods. First, some terms used in the EEK textual carriers often mean more than they are shown literally. For example, "XREF" shown in Table 1, which is the abbreviation of "external reference", is used as a command that allow the users to load the lines, annotations and other attachments from another drawing file in a collaborating project, which could only be matched semantically with an auto-learnt domain dictionary or a professional ontology for translating codes into meanings. Second, the proposed generating method results a static panorama of the field structure. In the future, we will try to illustrate the network dynamically along with the time and analyze the variation of the networks for the purpose of researching the evolution of empirical engineering knowledge in a long history.

Acknowledgements

The research was supported by National Natural Science Foundation of China (No. 70971085, 71271133), Shanghai Science and Technology commission (No. 13111104500), and Shanghai Municipal Education Commission (13ZZ012).

References

[1]. Dosi, G., M. Faillo and L. Marengo, Organizational capabilities, patterns of knowledge accumulation and governance structures in business firms: An introduction. ORGANIZATION STUDIES, 2008. 29(8-9): p. 1165-1185.

[2]. Žemaitis, E., Knowledge Management in Open Innovation Paradigm Context: High Tech Sector Perspective. Procedia - Social and Behavioral Sciences, 2014. 110: p. 164-173.

[3]. Palvia, P.C., S. Palvia and J.E. Whitworth, Global information technology: a meta analysis of key issues. INFORMATION & MANAGEMENT, 2002. 39(5): p. 403-414.

[4]. Yu, C., C. Davis and G.P.J. Dijkema, Understanding the Evolution of Industrial Symbiosis Research A Bibliometric and Network Analysis (1997-2012). JOURNAL OF INDUSTRIAL ECOLOGY, 2014. 18(2): p. 280-293.

[5]. Avila-Robinson, A. and K. Miyazaki, Evolutionary paths of change of emerging nanotechnological innovation systems: the case of ZnO nanostructures. SCIENTOMETRICS, 2013. 95(3): p. 829-849.

[6]. Gerdsri, N., A. Kongthon and R.S. Vatananan, Mapping the knowledge evolution and professional network in the field of technology roadmapping: a bibliometric analysis. TECHNOLOGY ANALYSIS & STRATEGIC MANAGEMENT, 2013. 25(4): p. 403-422.

[7]. Robert, C., et al., The evolution of the sleep science literature over 30 years: A bibliometric analysis. SCIENTOMETRICS, 2007. 73(2): p. 231-256.

[8]. Liu, L., Z. Jiang and B. Song, A novel two-stage method for acquiring engineering-oriented empirical tacit knowledge. International Journal of Production Research, 2014: p. 1-22.

[9]. Chen, Y., Development of a method for ontology-based empirical knowledge representation and reasoning. DECISION SUPPORT SYSTEMS, 2010. 50(1): p. 1-20.

[10]. Kump, B., et al., Tracing knowledge co-evolution in a realistic course setting: A wiki-based field experiment. COMPUTERS & EDUCATION, 2013. 69: p. 60-70.

[11]. Pyka, A., N. Gilbert and P. Ahrweiler, Simulating knowledge-generation and distribution processes in innovation collaborations and networks. CYBERNETICS AND SYSTEMS, 2007. 38(7): p. 667-693.

[12]. Liu, J., G. Yang and Z. Hu, A Knowledge Generation Model via the Hypernetwork. PLoS ONE, 2014. 9(3): p. e89746.

[13]. Lee, K.M. and K.M. Lee, Agent-based Knowledge Evolution Management and Fuzzy Rule-based Evolution Detection in Bayesian Networks. 2013 INTERNATIONAL CONFERENCE ON FUZZY THEORY AND ITS APPLICATIONS (IFUZZY 2013), 2013: p. 146-149.

[14]. Ruiz, P.P., B.K. Foguem and B. Grabot, Generating knowledge in maintenance from Experience Feedback. KNOWLEDGE-BASED SYSTEMS, 2014. 68(SI): p. 4-20.

[15]. Chen, Y. and Y. Chen, Knowledge evolution course discovery in a professional virtual community. Knowledge-Based Systems, 2012. 33: p. 1-28.

[16]. Argote, L. and E. Miron-Spektor, Organizational Learning: From Experience to Knowledge. ORGANIZATION SCIENCE, 2011. 22(5): p. 1123-1137.

[17]. Chan, F., Application of a hybrid case-based reasoning approach in electroplating industry. EXPERT SYSTEMS WITH APPLICATIONS, 2005. 29(1): p. 121-130.

[18]. Song, B., Z. Jiang and X. Li, Modeling knowledge need awareness using the problematic situations elicited from questions and answers. Knowledge-Based Systems, 2015. 75: p. 173-183.

[19]. Shah, C. and J. Pomerantz. Evaluating and predicting answer quality in community QA. in Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. 2010: ACM.

[20]. Shi, Y., H. Pan and T. Li, Evaluation model of university management informatization level based on fuzzy neural network T-S. JOURNAL OF INVESTIGATIVE MEDICINE, 2014. 62S(8): p. S108-S108.

[21]. Mosleh, M., T. Allahviranloo and M. Otadi, Evaluation of fully fuzzy regression models by fuzzy neural network. NEURAL COMPUTING & APPLICATIONS, 2012. 211: p. S105-S112.

[22]. Fukuda, S., Assessing the applicability of fuzzy neural networks for habitat preference evaluation of Japanese medaka (Oryzias latipes). 2011. p. 286-295.

[23]. Wong, W.K., X.H. Zeng and W.M.R. Au, A decision support tool for apparel coordination through integrating the knowledge-based attribute evaluation expert system and the T-S fuzzy neural network. EXPERT SYSTEMS WITH APPLICATIONS, 2009. 36(2): p. 2377-2390.

SESSION

WORKSHOP - INTELLIGENT LINGUISTIC TECHNOLOGIES, ILINTEC'15

Chair(s)

Dr. Elena B. Kozerenko

Parametrizing Verb Second Languages and Clitic Second Languages

Anton Zimmerling [1]
[1]Institute for Modern Linguistic Research, Sholokhov Moscow State University for the Humanities / Institute of Linguistics, Russian Academy of Sciences, Moscow, Russia

Abstract - *In this paper, I discuss verb second (V2) and clitic second languages (CL2) as an object of the parametric typology. V2 systems are a small group of languages sharing a number of parameters constraining the clausal architecture and the finite verb placement. CL2 systems are a large group of languages sharing a number of parameters constraining the clausal architecture and clitic ordering. The constituting property of both V2 and CL2 syntax is the bottleneck condition licensing only one constituent in XP. The ordering of finite verbs in V2 languages and the ordering of clitics/clitic clusters in CL2 languages is explained as head movement of the diagnostic category (verb or clitic) to 2P accompanied by the XP-movement of phrasal categories. An extension of the class of rigid V2 languages is the class of V1/V2 languages which license IS-motivated V1 orders in declarative clauses, while CL1/CL2 languages are rare*.

Keywords: parametric typology, movement, clausal architecture, verb second languages, clitic second languages

1 Introduction

Word order systems of natural languages are based on predictable constraints. Predictability of word order is often explained by some underlying principles of Universal Grammar (UG) [1]. However, such well-known constraints as the verb second constraint (V2) attested e.g. in German and Swedish, or clitic second constraint (CL2) attested e.g. in Serbo-Croatian, Pashto and Warlpiri, correspond to specific parameter settings characteristic of their language type rather than to any elementary omnipresent UG features, cf. [5 ; 12; 22; 20] for discussion. V2 languages share many parameters in word order which can be implemented in rule-based NLP parsing: the description of a V2 language should contain language-specific parameter settings plus shared type-specific features [14]. A similar procedure can be applied to a larger class of CL2 languages [21]. Preliminary research findings suggest that cross-linguistic variation within V2 languages and CL2 languages is set out by similar or identical parameters licensing or ruling out topicalization, multiple XP-movement, V1/CL1 orders etc. Therefore, we put forward the hypothesis that V2 languages and CL2 form together a natural class of second position (2P) languages defined in terms of shared constraints on clause structure, cf. [9].

2 Definitional properties of 2P languages

A descriptive schema of 2P needs four symbols – a symbol of clausal (left) border (#), a symbol of the preverbal or preclitic constituent (XP) and symbols for the pivotal category – the finite verb (V_{fin}) or clustering clitic (CL), as represented by (G)eneralization 1.

$$\text{(G1)}\quad \#XP — V_{fin}, /CL \; {}^{*}\#XP — Y — V_{fin}/CL$$

The bottleneck condition is crucial for 2P diagnostics. It predicts two features of 2P syntax: a) that a combination of two (or more) phrasal categories X, Y preceding the finite verb in a V2 language / the clustering clitics in a CL2 language should be ungrammatical; b) that the XP-position in the diagnostic type of V2 declaratives / any clause type with CL2 is not reserved for any particular syntactic category (e.g. noun phrase) and does not express any particular grammatical relation (e.g. subject).[2] In other words, the XP-position in a 2P language can be filled by any element in an OR-expression $\{Cat_1 \vee Cat_2 \vee \ldots Cat_n\}$, but simultaneous spell-out of two or more hierarchically independent categories in XP is blocked: $^{*}\#\{_{XP}\; Cat_1 \;\&\; Cat_2 \;\&\ldots\; Cat_n\} — V_{fin}/CL$. Parsing of well-formed 2P structures is licensed by a combination of an OR-expression filter, which lists sentence categories that fill XP in language L, and an &-expression filter which determines which types of expressions count as single constituents when filling XP in language L, and which do not.

[1] UG is defined framework-neutrally as shared part of all partial grammars.

[2] Roberts [12] refers to the (b) property of V2 languages as x-EPP], which is to some extent misleading, because the [EPP] feature in minimalist syntax is responsible for attraction of a constituent to the specifier of a head. The insight behind Roberts' notation is that in a language like English, where T is characterized as having an [EPP] feature, it attracts the subject exclusively, which is not the case with the first clause position in V2 languages. Thus, the label [-EPP] is rather a shortcut of an empirical generalization that the preverbal position is not dedicated for subjects.

2.1 Differences between V2 and CL2 systems

There are four main differences between V2 and CL2 languages.

(a) Sententional clitics form clusters in 2P, finite verbs do not. Clitic clusters are formed by Template algorithms predicting the rigid ordering in each pair of clustering clitics (a, b). Template rules for clitic clusters are equivalent to A-grammars [19].

(b) Finite verbs in V2 languages are obligatory elements of tensed clauses, clustering clitics in CL2 are optional elements.[3]

(c) Finite verbs in V2 languages take 2P only within a diagnostic type of declarative clauses (either root clauses or all declaratives), while clustering clitics normally take 2P in all types of root and complement clauses.

(d) V2 languages and CL2 languages differ with respect to topicalization. In those V2 languages which license initial topicalized elements in front of XP, such topicalized phrases lie clause-external, cf. example (1) from Kashmiri showing an topicalized DP coindexed with the clause-internal resumptive pronoun, and example (2a) from Swedish showing a dislocated VP fragment coindexed with a resumptive clause-internal auxiliary.

(1) [$_{DP}$ Su laRk]$_i$, Rameshan vuch *temis*$_i$ tsuur karaan.
that boy Ramesh saw he theft do
'As for that boy, it is Ramesh who saw him stealing'.

(2) a. [$_{VP'}$ Läser$_j$ boken]$_i$ *det*$_i$ *gör*$_j$ han nu.
read book it does he now
'He is reading the book, that is what he is doing now'.
b. *[$_{VP'}$ Läser$_j$ boken]$_i$ *gör*$_j$ han nu.[4]

In CL2 languages, the initial topicalized phrases causing late clitic placement lie clause-internally. The proof comes from those CL2 languages where initial topical constituents at once trigger CL > 2 orders and attract verbs to clausal 2P. This mechanism is known as Communicative Barrier, cf. [21]. Optional Barrier effect triggered by the topicalized PP [$_{PP}$Poslije svega toga] 'after all that' is illustrated by the pair of Serbo-Croatian examples (3a-b) both taken from one and the same Croatian idiom[5].

(3) a. [$_{PP}$ Poslije toga]=*su* dobili pozive u reprezentaciju.
after that CL get calls to team
'After that, they have been summoned to the national team.'

b. [^{BARRIER} [$_{PP}$Poslije svega toga]] bilo =*mi* =*je*
After all that was CL CL
potrebno samo ležati na pijesku.
necessary only lie on sand
'After all that, everything I needed was to lie on sand.'

The distribution of word orders #[$_{XP}$ X] — CL…V in (3a) vs # [^{BARRIER} [$_{XP}$ X]] — V — CL in (3b) proves that the verb and the clitics compete here for one and the same slot, i.e. 2P. The initial topical Barrier prevents the clitics from taking 2P, while the vacant 2P attracts the verb.

2.2 Areal distribution of 2P languages

Most V2 systems are known from Germanic languages. Moreover, all attested Germanic languages, except for Gothic, Elder Futhark, English and Old English are V2 systems. Rhaeto-Romance is considered a V2 system developed due to contact influence of Germanic languages [6]. The same has been claimed about Old French [9], but in the corpus of 1170 Old French Bible V>3 orders in declaratives are still frequent [6]. Thus, Old French is hardly a V2 system, since it violates the condition (G1). The only one genuine example of a V2 system developed without any contact with Germanic V2 languages is the Dardic language of Kashmiri [2].

CL2 systems are attested in virtually all areas, cf. Lummi (Salishan), Kashibo-Kakaitabo (Panoan), Caviňena (Tacan), Mayo (Uto-Aztecan), Kabyle (Afro-Asiatic), Pashto and Ossetic (Indo-Iranian), Maori (Polynesian), Warlpiri and Djaru (Pama-Nyungan), Czech, Slovene, Serbo-Croatian (Slavic) as well as in a number of Old Indo-European languages including Old Greek, Sanskrit, Hittite, Old Persian, Old Novgorod Russian etc. [20: 62; 21].

2.3 Feature-driven movement and its triggers

In the Minimalist Program and its mathematical formalization, Stablerian Minimalist Grammars [13], movement is feature-driven.

2.3.1 Head movement to 2P

Early (up to mid-1990s) versions of Chomsky's framework tag the left periphery as a C(omplementizer)P. Functional heads like C or, in later notation, Fin(iteness), are defined as having an uninterpretable Tense feature (*u*T-feature) or a set of uninterpretable inflectional agreement features (*u*Phi-features) attracting the pivotal category – tensed verbs in V2 languages and second position clitics in CL2 languages, cf. [12] and [21]. If such categories (verbs, clitics etc.) get a fixed position with respect to the clausal left border, this means that verb movement/clitic movement is obligatory, at least within the diagnostic group of clauses — with a stipulation that verb/clitic movement does not take

---

[3] 2P clitics are free syntactic elements that are inserted (merged) in clause structure only if the predicate requires their presence. In this sense, all 2P clitics are optional. E.g. an auxiliary perfective clitic is merged only when there is a perfective verb requiring it.

[4] The ill-formedness of (2b) proves that topicalized VP fragments like [$_{VP'}$ Läser$_j$ boken] do not take XP is Swedish

[5] For the details about this idiom see [20: 457-462].

place if the target position is already filled by some other category, as in the example (3b) above.

2.3.2 XP-movement

XP-movement is triggered by an active Edge Feature [+EF], which attracts phrasal categories to the clause-initial position (SpecCP or SpecFinP in the later notation). The moved phrasal category and the moved verb/clitic head (V^0, cl^0) form a required Spec-Head configuration for feature agreement, which is a well-formedness condition in the early minimalist syntax. This analysis captures correctly three basic facts about 2P languages: a) verb movement to 2P is obligatory in the diagnostic group of clauses, b) all categories that can fill XP lie clause-internally, c) head movement to 2P and phrasal movement to SpecTP in 2P languages have grammar-internal motivation and do not depend on IS/prosody. On the contrary, topicalization constructions with the order [TOP] — XP — V_{fin}/CL, cf. (1), (2a), (3b), are marked and always have Information Structure (IS) triggers.

2.3.3 [+ EPP] languages with non-clustering 2P elements

Non-clustering clause-level 2P elements called 'predicative markers' or 'finite operators' are attested in a number of Mande languages like Bamana, Dan-Gweeta [15] and Kpelle [7] with the basic word order (X) –S – AUX – O – V. The main verb takes a different position which overtly reminds of V2 systems. However, the clause-initial /pre-finite position in Mande languages is reserved for grammatical subject. Hence, they are [+EPP] languages without XP-movement. It is unclear whether finite operators in such word order systems take clausal 2P by movement or are directly merged there: in any case, their placement does not pattern with verb movement and they lack both verbal morphology and, in many Mande languages, clitic features.

3 Parametric variation

In this section, I am discussing parameters of cross-linguistic variation pertaining both to V2 and to CL2 languages.

3.1 Standard XP-movement

The clause-initial position in 2P languages (XP) can be filled both by maximal projections i.e. spelled-out constituents as well as by non-maximal expressions e.g. syntactic heads, syntactic complements, phonetic words. This holds for both V2 languages and for CL2 languages, though CL2 languages have more options.

3.1.1 Types of clitic hosts in CL2 systems

Basing on conditions imposed on the structure of clitic hosts, all CL2 systems are classified into four types [20: 69].

(a) W_1 systems. These are 2P languages which either do not have NPs/DPs or do not license them in XP. XP is filled by two categories in an OR-expression $\{Cat_1 \vee Cat_2\}$. W_1 systems are rare. They are attested in Lummi and in Berber languages where 2P clitics are hosted either by verbs or by some operator words/TAM markers.

(b) W_{2A} systems are 2P languages which license NPs/DPs and other types of constituents in XP and invariably place 2P clitics after the first phonetic word with the basic word order. This type is exemplified by Hittite, Sanskrit, Old Novgorod Russian etc.

(c) W_{2B} systems are 2P languages which license NPs/DPs and other types of constituents in XP and invariably place 2P clitics after the first spelled-out constituent with the basic word order. This type is exemplified by Pashto, Ossetic, Czech, Slovak, Cavineña etc.

(d) W_{2C} systems are 2P languages which license NPs/DPs and other types of constituents in XP and license 2P clitics both after the first phonetic word and after the first spelled-out constituent with the basic word order. This type is exemplified by Walrpiri, Warumungu, Luiseňo, Serbo-Croatian, Old Czech [20: 481-484] etc.

3.1.2 Non-maximal expressions in XP in V2 languages

From 4 theoretic possibilities corresponding to W_1 and W_2 -systems, only two, notably $V2_{2B}$ and $V2_{2C}$ systems are attested. In V2 systems, the structure of the initial constituent does not have prosodic triggers. $V2_1$-systems without NPs/DPs are theoretically possible but not attested. $V2_{2A}$ systems ruling out maximal projections in NP are not attested either: if such languages exist, they could be interpreted as a transitional stage from CL2 to V2. The majority of V2 languages pattern with the $V2_{2B}$ systems, while several V2 languages including Old Icelandic are true $V2_{2C}$ systems. Old Icelandic example (4a) shows extraction of a head element from an NP/DP, while Old Icelandic example (4b) shows left branch extraction (LBE)[6].

(4) a. [$_{XP}$ *Útfall*] var [$_{DP}$ ~~útfall~~ sjávarinnar].
flood was sea
'The was a flood of tide.'

b. [$_{XP}$ *Þeirrar*] skal=tu [$_{DP}$ ~~þeirrar~~ konu] biðja.
This shall.you woman woo
'You shall woo that woman.'

[6] In the copy theory of movement, cf. [6], both (4a) and (4b) are explained by postulating movement of maximal projections and discontinous spellout of the moved constituent.

3.2 Multiple XP-movement

Some 2P languages occasionally or regularly license multiple XP-movement which gives rise to the so called initial ensembles. An ensemble is a string of elements that make up a single constituent in a dedicated position but in other positions are hierarchically independent [21: 196; 22: 666]. In some cases it is impossible to determine whether a sequence Cat_1 & Cat_2 forms a single constituent or not without checking its capacity to fill XP. So Norwegian, which is considered a strict V2 language, occasionally licenses sequences of several adverbials in XP, cf. (5).

(5) $[_{\text{XP}}\ [_{\text{AdvP}}$ *I byen*$]\ [_{\text{AdvP}}$ *i dag*$]]$ trefte jeg Marit.
$[_{\text{XP}}\ [_{\text{AdvP}}$ In the town$]\ [_{\text{AdvP}}$ today$]]$ met I Marit.
'Today, I met Marit in the town'.

Both adverbials in (5) have the same IS status (correspond to a Theme), and there are no grounds to believe that any of them is extraclausal. Therefore, one must assume that at least those native speakers who accept (5) generate/parse a single adverbial phrase there. Multiple XP-movement in declaratives is characteristic for a minority of 2P languages including Modern Icelandic, Faroese and Old Swedish, cf. [17: 291, 501], Czech and Bulgarian [20: 375-79]. XP-ensembles in declarative clauses always have IS triggers and often correspond to an initial contrastive Theme, cf. the Czech example (6)[7].

(6) $\{_{\text{Contr Topic}}[_{\text{XP}}\ [_{\text{DP}}$ *Petra*$]\ [_{\text{AdvP}}$ *do Francie*$]]\}$ =bych
Peter to France CL.would-I
ještě poslal, ale Martina do Maďarska ani náhodou'
still send but Martin to Hungary never
'I still could sent P. to France, but never M. to Hungary.'

XP-ensembles in 2P languages seem to obey the following generalization on the mapping of formal syntax and IS. Ensembles containing formal categories α, β are only licensed if they form a single communicative constituent A in XP, see schematically below in (G2):

(G2) $\#\{_{A}\ [_{XP}\ [_{XP}\ \alpha\ ]\ [_{XP}\ \beta]]\}$ — V_{fin}/CL, $*[_{XP}\{_{A}\ [_{XP}\ \alpha]\}\ \{_{B}[_{XP}\ \beta]\}]$ — V_{fin}/CL.

A similar parameter licenses optional multiple *wh*-fronting in Kashmiri interrogatives, cf. (7).

(7) a. $[_{\text{WhP}}\ [_{\text{WhP}}$ Kus$]\ [_{\text{WhP}}$ kemyis$]\ [_{\text{WhP}}$ kyaa$]]$ dii?
who whom what give-FUT
'Who will give what to whom?'

b. $[_{\text{WhP}}\ [_{\text{WhP}}$ Kus$]\ [_{\text{WhP}}$ kyaa$]]$ dii $[_{\text{WhP}}$ kemyis$]$?

c. $[_{\text{WhP}}\ [_{\text{WhP}}$ Kus$]]$ dii $[_{\text{WhP}}$ kemyis$]\ [_{\text{WhP}}$ kyaa$]$?

XP-ensembles can be explained by Sidewards Movement in the model of Baylin [1]. Note however that XP-ensembles do not always show Superiority Effects, i.e. fixed ordering. Other 2P languages ban multiple XP-movement and multiple wh-fronting. Bulgarian is the only one language which has both XP-ensembles in declaratives (without Superiority) and obligatory wh-fronting (with Superiority).

3.3 Root versus subordinate clause asymmetry

Stable CL2 languages have a tendency towards uniform clitic placement in all types of clauses, root and subordinate. V2 languages tend towards asymmetrical finite verb placement. V2 is primarily a root phenomenon which gave rise to theories that V_{fin} and Comp compete for one and the same slot, cf. [3; 4]. However, the prediction that an overt complementizer automatically blocks V2 in subordinate clauses has not been borne out: some V2 systems like Kashmiri or Afrikaans have symmetrical V2. There exist CL2 systems which skip complementizers as clitic hosts. Hence, both groups of 2P languages may apply both symmetrical and asymmetrical ordering of the pivotal 2P elements.

3.3.1 Asymmetrical V2

Modern German, Dutch, Danish, Swedish and Norwegian show root-subordinate clause asymmetry: in the presence of an overt complementizer 'that' finite verbs do not take V2 and are either placed clause-finally — the West-Germanic option, cf. German examples (8a-b), or one step further to the right, after negation/negative phrases/sententional adverbs — the Mainland Scandinavian option, cf. Danish examples (9a-b).

(8) a. Der Hans <u>hat</u> dem Peter keine Instruktionen <u>gegeben</u>.
'Hans has not given any instructions to Peter.'

b. Ich glaube, $[_{\text{CP}}$ *daß* {der Hans dem Peter keine Instruktionen <u>gegeben</u>} <u>hat</u>].
'I believe that Hans has not given any instructions to Peter.'

(9) a. Jens <u>har</u> ikke <u>givet</u> instruktioner til Peter.
'Jens has not given instructions to Peter.'

b. Jeg tror, $[_{\text{CP}}$ *at* Jens $\{_{\text{Neg}}$ ikke$\}$ <u>har</u> <u>givet</u> instruktioner til Peter].
'I believe that Jens has not given instructions to Peter.'

3.3.2 Symmetrical V2

Kashmiri and Afrikaans regularly have V2 orders in subordinate clauses. For these languages, the diagnostic group of clauses with V2 phenomena must be extended to all declaratives.

---

7 Curly brackets in example (6) and in (G2) mark communicative constituents, square brackets mark formal constituents.

3.3.3 Symmetrical CL2

In CL2 languages, clause-internal clitics and clause-initial complementizers never form a complementary distribution. The exact definition of root vs subordinate asymmetry with respect to clitic placement is a controversial issue. If 2P clitics skip complementizers as their hosts and attach the presence of an overt Comp node does not affect clitic placement, as shown on G(eneralization) (G3).

(G3) #XP/Xº — CL ~ <Comp> — YP — CL

(G3) correctly captures the situation in such CL2 languages as Pashto, Tagalog, Cebuano and other Central-Philippine languages [18; 20: 94].

3.3.4 Asymmetrical CL2

In Ossetic, Warumungu, South-Eastern Tepehuan, Ossetic and other CL2 languages clitics can or must attach to the complementizer, as shown on on G(eneralization) (G4).

(G4) #XP/Xº — CL ~ Comp> CL

Asymmetrical CL2 languages conforming to (G4) normally preserve the strategy (G3) as a reserve option, while symmetrical CL2 languages that have (G3) as the main strategy do not apply (G4) at all. Along the same lines, asymmetrical V2 languages often preserve the possibility of embedded V2 which makes some authors speak of 'subordinate clauses with the main clause order <i.e. V2>', cf. [14].

3.4 V1 and CL1 orders

3.4.1 V1/V2 languages

A natural extension of rigid V2 languages are V1/V2 languages where finite verbs can take clausal-initial position in the same diagnostic group of declarative clauses. V1/V2 are represented e.g. by Modern Icelandic, Faroese, Yiddish, Kashmiri, Old Icelandic, Old Saxon, Old High German. In these languages, V1 orders in declaratives are frequent and have IS-triggers: they can be interpreted as IS-marked variants of V2 declaratives as suggested in [17: 362-266; 22: 669]. CL1/CL2 are poorly represented, since clustering clitics are normally specified as [+ strict enclitics] i.e. as clause-internal elements. The shift from clustering 2P enclitics to clustering 1P proclitics is not attested, though two CL1/CL2 languages, Macedonian and Rittharngu, lie close to this type [18]. [8]

[8] Macedonian pronominal clitics behave as strict enclitics in nominal clauses and as proclitics in verbal clauses. CL1 orders (#CL—V) are the default option in declaratives, while CL2 orders (#V—CL) are considered marked. CL > 2 orders and non-adjacent placement of clustering clitics and verbs are impossible [20: 308].

3.4.2 V1 orders in rigid V2 systems

In some Germanic languages that normally described as rigid V2 systems, V1 orders are occasionally licensed in special pragmatic contexts e.g. in thetic sentences with inferential semantics, cf. the German example (10) and the Swedish example (11)

(10) #*Kenn* ich nicht.
'I <really> don't know <him, her, them>'

(11) <Var är tidningen?> #*Så* jag nyss på bordet.
'<Where is the newspaper?> I have <just> seen it on the table'.

Examples (10), (11) are probably indirect speech acts rather than true declaratives: verb fronting amounts here to adding an overt discourse particle. This strategy has a direct counterpart in some apparently rigid CL2 languages.

3.4.3 CL1 orders in rigid CL2 systems

Some presumably rigid CL2 languages apparently apply to a similar strategy – clitic fronting in indirect speech acts. This is attested in Slovene where CL2 is the default word order, cf. (12a), while CL1, cf. (12b) is a marked option reserved for thetic sentences similar to (10) and (11).

(12) a. #Videl=*sem*=*ga*.
'I have seen him/it'.

b. #*Sem*=*ga*=videl.
'I have <indeed> seen him/it'

3.5 Clausal architecture and 2P

3.5.1 Complementary distribution of verbs and clitics

One of the basic insights behind the 2P analysis is that 2P is a dedicated position attracting not only the pivotal category (finite verbs in V2 languages, clitics in CL2 languages) but also some other categories which may reach 2P if the pivotal category is absent [5]. However, the analysis that Comp and V2 must form a complementary distribution so V2 should only be possible if Comp is not filled lexically [4] falls short of explaining symmetrical V2 effects, cf. section 3.3.2 above. Besides, Comp freely co-occurs with CL2 and may host clitics, cf. section 3.3.4 above. All this indicates that the initial claim that Comp and V2 must be distributed complementary was wrong. The real validation of the hypothesis that 2P can be filled by different categories in one and the same language comes from 2P systems where such categories are found within *one and the same type of clauses*. The most salient case are Barrier configurations, verb movement and late placement of clitics discussed above in section 2.1, cf. example (3b) where the verb takes clausal 2P which is made vacant due to the impact of the clausal initial topical Barrier preventing 2P clitics from taking it. In asymmetrical CL2 languages, Barrier configurations with verb movement are especially characteristic of root clauses, cf. [20: 445-464].

3.5.2 Complementary distribution of finite verbs and adverbials in 2P

This rare option is attested in Swedish where the modal adverbial *kanske* 'maybe' takes the same slot as the tensed verb, so the auxiliary *vill* in (13a-c) must stand outside 2P as shown in [11].

(13) a. Nu *kanske* Johan inte vill komma.
now MAYBE John not FUT come.INF
'John probably won't come now'.

b. Johan *kanske* inte vill komma nu.
John MAYBE not FUT come.INF now
'the same.'

c. Vem *kanske* inte vill komma?
Who MAYBE not FUT come.INF
'Who won't probably come?'

Examples (13a-c) from a Germanic V2 system serve as a close parallel to the example (3b) illustrating the verb-and-clitic distribution in a Slavic CL2 system with verb movement. In both cases, the pivotal category is prevented from taking 2P by movement if this node is already filled.

3.6 Verb-clitic adjacency and 2P

A majority of CL2 languages (the so called standard W-systems in terms of [18; 21]) lack a constraint on clitic-verb adjacency with the basic word order #XP —CL, though [V—CL] orders may arise in derived structures with Barriers. However, there is class of CL2 languages which have developed the clitic-verb or verb-clitic adjacency with the basic word order. In [18] this class is dubbed W^+-systems i.e. modified CL2 systems. There are two principal varieties of W^+-systems. In W^+_1 systems. attested in Modern Bulgarian, Tagalog, Cebuano, Bikol and other Central Philippine languages clustering clitic invariably take 2P with the basic word order, while the verbs take adjacent positions either from the left or from the right of clitics. If clitics are absent, the verb does not have a fixed position. This parametric setting is captured by G(eneralization) (G5)[9].

(G5) #XP — [CL —V] ~# [V—CL]; #....V….#.

In the second variety of W^+-systems, the pivotal element is the verb which cannot move further to the right from 2P, while clustering clitics (or clustering weak elements) take adjacent positions. This parametric setting is captured by G(eneralization) (G6) and attested in such languages as Old Icelandic, Middle Norwegian [17: 465] and, according to some descriptions, also in a number of Middle Romance languages including late Old French [9] and Old Portuguese [16].

(G6) #XP — [V —CL] ~# [V—CL]; *#XP —Y —V.

The taxons 'V2 system' and 'W^+_2 system' intersect: a V2 or V1/V2 language may also have clustering clitics with a fixed position. The taxons 'V2 system' and 'W system', 'V2 system' and 'W^+_1 system' do not intersect per definition, since no language can have both CL2 and V2 in the same position with the basic word order.

4 Conclusions

The hypothesis that V2 languages and CL2 languages form together a class of 2P languages which is definable in terms of a shared set of parameters has borne out. The placement of the pivotal categories V and CL to 2P as well the filling of XP must be analysed as movement. The main difference between V2 systems and CL2 systems is that CL > 2 orders regularly arise in CL2 systems in derived structures with an initial topical constituent, while V_{fin} > 2 orders with two or more clause-internal constituents placed in front of the finite verb in the diagnostic type of declarative clauses are exceptionally rare. V2 declaratives may have IS triggered V1 variants, but CL2 clauses usually lack IS triggered CL1 variants.. Therefore, the class of V1/V2 languages is a natural extension of the class of rigid V2 languages, while true C1/C2 languages are infrequent. At the same time, occasional verb fronting in rigid V2 languages and occasional clitic fronting in rigid CL2 languages have identical IS triggers.

5 Acknowledgments

This paper has been written with financial support from the Russian Federal Project 2685 'Parametric Description of Grammar Systems'. I grateful to Ekaterina Lyutikova, Maria Konošenko and Valentin Vydrin for the consultation.

6 References

[1] Bailyn, John Frederick. "The Syntax of Russian". Cambridge, Cambridge University Press, 2012.

[2] Bhatt, Rajesh M. "Verb movement and the syntax of Kashmiri". Dordrecht, 1999.

[3] Den Besten, Hans. "On the interaction of root transformations and lexical deletive rules"; *On the formal nature of the Westgermania* /Abraham W. (ed.). John Benjamins, Amsterdam, 1983, 47-131.

---

[9] The Macedonian parametric setting can be set out as a modified variant of (G5), with an extra option #[CL —V]:
(G5)' # XP — [CL —V] ~#[V—CL] ~ #[CL —V]; #...V…#.

[4] Holmberg, Anders & Platzack, Christer. "The Role of Inflection in Scandinavian Syntax". Oxford, 1995.

[5] Holmberg, Anders. "Verb second". In: *Syntax – Theory and Analysis. An International Handbook.* T. Kiss & A. Alexiadou (eds.). Vol.1. Berlin: Mouton de Gruyter, 2015, 242–283.

[6] Kaiser, Georg & Zimmermann, Michael. "Exploring verb order differences between (Medieval) Romance and (Modern) Germanic". In: *Traces of History Conference*. The University of Oslo, March 9–10, 2015.

[7] Konošenko, Maria. "Person-and-number agreement in Kpelle: on the typology of person-and-number agreement" [Lično-čislovoe soglasovanie v yazyke kpelle: k tipologii soglasovanija po licu i čislu] (In Russian). In: *Voprosy jazykoznanija*, 2013, No. 1, 95-114.

[8] Legendre, Geraldine. "Masked second-position effects and the linearization of functional features". In: *Optimality-theoretic syntax*. Cambridge, MIT Press, 2001, 241–277.

[9] Mathieu, Eric. "Stylistic Fronting in Old French". In: *Probus*, 18, 2008, 219–266.

[10] Nunes, Jairo. "The copy theory of movement and linearization of chains in the Minimalist Program." Doctoral dissertation, University of Maryland, College Park, 1995.

[11] Platzack, Christer. "The Edge Feature on C". Lund, University of Lund, 2008.

[12] Roberts, Ian. "Phases, Head-movement and Second-Position Effects". In: *Phases developing the framework* / Angel J. Gallego (ed.). Berlin, Boston: Mouton de Gruyter, 2012, 385–440.

[13] Stabler, Edward P. "Derivational minimalism". In: *Logical Aspects of Computational Linguistics*, / Christian Retore (ed). Springer, 1997, 68–95.

[14] Vikner, Sten. "Verb Movement and Expletive Subjects in the Germanic Languages". Oxford Studies in Comparative Syntax, New York, Oxford: Oxford University Press, 1995.

[15] Vydrin, Valentin. "Agreement of the pronominal predicative marker with the subject in Dan-Gweeta". [Soglasovanie mestoimennogo predikativnogo pokazatel'a s podležaščim v Dan-Gueta.] (In Russian). In: *African collection'* / A.Ya.Želtov (ed.), Sankt-Peterburg, Museum of anthropology and ethnography, 2013, 264-285. [Afrikanskij sbornik. Sankt-Peterburg: Muzej antropologii i ethnografii RAN, 2013, 264-285].

[16] Wanner, Dieter. "Second Position Clitics in Medieval Romance". In: *Approaching Second: Second Posiition Clitics and Related Phenomena*. / A.L.Halpern & A.M.Zwicky (eds.). Stanford, CSLI Press, 1996, 537–578.

[17] Zimmerling, Anton. *Typological Scandinavian syntax*. [Typologicheskiy sintaksis skandinavskikh yazykov] (In Russian). Moscow, Yazyki slavyanskoj kultury, 2002.

[18] Zimmerling, Anton. "Word order systems with clitics in a typological aspect" [Sistemy poryadka slov v tipologičeskom aspekte] (In Russian). In: *Voprosy jazykoznanija*, 2012, No. 4, 3-38.

[19] Zimmerling, Anton."A unified analysis of clitic clusters in world's languages". In: *Dialogue. Computational linguistics and intellectual technologies*. Issue 11 (18), Moscow, 2012, 726-738.

[20] Zimmerling, Anton. "Slavic word order systems from the viewpoint of formal typology." [Sistemy poryadka slov slavyanskikh yazykov v tipologicheskom aspekte] (In Russian). Moscow, Yazyki slavyanskoj kultury, 2013.

[21] Zimmerling, Anton & Kosta, Peter. "Slavic clitics: a typology". In: *STUF – Language Typology and Universals*. Vol. 66. No 2, 2013, 178–214.

[22] Zimmerling, Anton & Lyutikova, Ekaterina. "Approaching V2: verb second and verb movement". In: *Computational Linguistics and Intellectual Technologie*s. Issue 14 (21) in two volumes, Volume 1, Moscow, RSUH, 2015, 663-675. [Kompjuternaja lingvistika i intellektual'nye texnologii: Po materialam ežegodnoj meždunarodnoj konferencii «Dialog». Moskva, 27–30.05.2015)]. Vyp. 14 (21): v 2 t. T. 1: Osnovnaja programma konferencii. Moskva, RGGU, 2015, 663–675.

Method for Generating Subject Area Associative Portraits: Different Examples

I. Galina[1], M. Charnine[1], N. Somin[1], V. Nikolaev[1], Yu. Morozova[1], O. Zolotarev[2]
[1]Institute of Informatics Problems, Federal Research Center,
Russian Academy of Sciences, Moscow, Russia
[2]Russian New University, Moscow, Russia

Abstract - For the last several years a group of scientists has been working on creation of a knowledge extraction system based on the automated generation of subject area associative portraits (SAAP) and on the construction of semantic context spaces (SCS) [1-11]. The ideology of the SAAP is based on the distributional hypothesis, stating that semantically similar (or related) lexemes have similar context and, conversely, similar context has similar lexemes. In the applied model we use an extended hypothesis, that includes not only the study of similarities and differences in the contexts of individual lexemes, but also arbitrary multiple lexemes fragments (significant phrases - SP) too.

Keywords: Subject Area, Associative Portraits, keywords extracting, significant phrases, thesaurus, Big Data

1 Introduction

As a theoretical model of the SAAP in this draft adopted semantic context space (SCS). Formally, a SCS is defined as a graph G=(V,E) with nodes v of V of important terms/phrases from subject area (SA) and the arcs of the graph (vi,v-j,Link,w-ij) E describing the relationship/connection between the phrases, where w-ij is the weight that expresses the power of association, and the Link-type of connection determined by the type of lexical and syntactic pattern/design, linking phrases. Informative, SCS is a dictionary of significant phrases, the elements of which are connected by associative relations. Any SAAP is a concrete implementation of SCS [1].

The project is implemented on very large data volumes (big data), presented in the Internet environment, which significantly increases the quality of the formed SAAP. As a material research we use freely distributed Russian-speaking NL-texts from the Internet, as well as for some of the subject areas - English language texts. In the future we plan to expand the project to other languages.

To date, we have processed the following subject areas: in Russian - "Remote sounding of agricultural land", "Psychology of business communication"; "Business processes of enterprises"; "Socio-political portrait of Russian regions" (as the training data for this subject are taken a part of SA one of Federal region "Socio-political portrait of Tatarstan". Information sources for this topic are: official websites of state institutions, political parties; national and regional media; social network Vkontakte, Twitter microblogging, etc.); "Monitoring of public opinion in the socio-political sphere", tapering to a training sample "Protest activity"; and AUV (Autonomous underwater vehicles") in Russian and English languages; "Computer Science" (in English), etc.

On these topics we found and processed from ten thousand to one and a half million documents in Russian and English languages, a total of about 160 GB. For example, on the SA of "Socio-political portrait of Tatarstan" formed the texts digest of about 20 GB, and on the subject of "Protest activity" there was found over 28 GB of test information.

2 Stages of project

This project is created in several stages. There was developed a version of algorithm formation of SAAP on the first stage, based on the material from different subject areas. The key component of a comprehensive methodology was created and would be developed. This is the module for statistical analysis for the algorithm of SAAP formation. There were performed: thematic searches of NL-texts downloaded from the Internet, the formation of large corpora of these texts, divided by subject areas, with highlighting of significant phrases (SP) and the creation of SP dictionaries in an automated mode. All of these items are composed to a united iterative algorithm which represents a method of obtaining SAAP for any subject area. Some stages will be corrected and improved in further work.

In the second phase of the project, a methodology for automatic generation of the SAAP has been further developed in the form of corrected mechanisms for semantic search in the space of NL-texts from the Internet. The algorithms methods were developed due to the training module of the statistical analysis on test samples with subsequent testing on real text flows from different subject areas. This project is based on the KEYWEN encyclopedia concepts [4].

The author’s experiments were conducted on above mentioned test samples; also the plan for further experiments for the entire term of the project was developed. The work will be continued.

At present time, the technique allows to automatically extract keywords (single words-tokens, terms and SP) from NL-texts, to identify associative links between them, to set an hierarchy of terms and SP with the clustering method and to build dictionaries of SP in various subject areas. The output of the algorithm is informative "cloud" of associative relationships for automatically allocated tokens (terms) and SP, from which the associative portraits of the subject areas (SAAP) are formed. Actually we received a "bunch" of associative links defined the value of the SP term and his place in the terminological system of the studied subject area. The associative links allow us to classify the considered term and/or SP to a particular theme

3 The SAAP forming method

The essence of the proposed method of forming SAAP is to iterate extension of the initial dictionary of SP to full SAAP. The method can be described as the following:

1. Choice of key terms that specify the subject area is carried out by the user: a) manually (at the first stage of the training sample, or to explore new topics); b) can be extracted automatically by the system based on the frequency characteristics (weights of the terms); or c) can be automatically taken out of the previous SAAP as initial AP (keywords).

2. Semantic search on a given keywords and an accumulation of relevant base of Internet texts which make up the corps of NL-texts from different SA (big data). During the debugging and improving the results the machine learning techniques are used.

3. Automatic extraction of terms from texts and drafting of frequency dictionaries of SA (statistical methods).

4. Formation of lists of the most important SA concepts on key terms and SP.

5. Splitting texts into segments or suggestions.

6. Construction of the context of each term as a set of containing segments/sentences.

7. Building of context vectors, i.e. a calculation is build for each meaningful term (or SP), and vector is formed from the obtained statistics. These context vectors determine the measure of proximity and associative relationships between keywords.

8. Calculation of the cosine proximity measure of context vectors (the vector components of SP - key words of this SA - is the joint frequency of occurrence of this SP with other SP's in the same context) and the choice of the most strong associative links. To calculate the cosine measures the following formula is used (Fig. 1):

$$\frac{x \cdot y}{|x||y|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

Fig. 1.

where x and y are vectors in a sign space, which themselves are SP from a given semantic context space (SCS), and i - is an index running over the number of signs.

9. Selection of candidates for the key terms (to expand the original frequency dictionary of SP) from the receipt of the strongest associations in key terms conducted as follows:

• searching for a cluster of closely related keywords;

• building an hierarchical structure on the set of the SPs with the hierarchical clustering method;

• calculating the center of the cluster specified keywords (SP) related to the subject area in the vector space;

• determining the most powerful terms that are candidates for inclusion in the SP from the number of new terms that are closer to the center of the cluster;

• setting the most distant from the center of the cluster keywords (SP) as candidates for exclusion from the composition of the SP;

• accepting the final decision on changing the composition of SP by adding or removing terms-candidates.

10. Replenishment of the initial composition of key terms with the strongest candidates that were automatically chosen by the system and go to step 1.

This is a layout algorithm. The statistical module was developed on the basis of this algorithm. It solves the problem of creating SAAP [1, 11]. The individual steps of the methodology, including mechanisms of semantic Internet surfing will be developed and improved in the next phase of the project.

4 The essence of machine learning

The processing of large amounts of texts from the Internet allows to collect the necessary statistical data to form a fairly complete picture of the SA, available in the form of SCS. The opportunity to perform machine learning on a large number of examples gives the system some flexibility and improves outcome.

Machine learning in identifying the key words of SA and associative relationships between them are based on the following scheme:

1) creation of training and testing samples for each text (document) with specified class;

2) preparation the texts to use them for learning: breaking the texts into elements - words, SP, punctuation, etc.; identification of "noise" in the text and its clipping; for some training samples lemmatization is done (bringing words to a normal form);

3) converting each document into a feature vector;

4) creating dictionary of SP (classifier) and further training of the classifier on the obtained training set;

5) verifying the accuracy of the classifier on the test array of texts.

Here are some examples of terms in the SP dictionaries denoting the various elements of business processes [5, 9].

A part of the SP dictionary on "Business processes in the enterprise" [12-16]

• names of business processes (repair, inspection, replacement of parts, parts fabrication, delivery);

• input (ordering, fault list, blank invoice, commercial invoice...);

• management (schedule, drawing, regulations, invoice...);
• participant/mechanism (master, mechanic, turner, accountant, driver...);
• output (repaired auto, fault list, item, invoice, goods...).

5 Socio-political portrait of Russian regions

The improvement of the formation algorithm of the SAAP is carried out, including machine learning techniques, with the conducting of experiments in building SAAP. As a training topics to address this problem was chosen subject area "Socio-political portrait of Russian regions". Republic of Tatarstanis was selected as a training sample. When building the ontology on a given SA we represent the most significant named entities of the following classes:
• geographical names of major settlements in the region;
• persons, their positions in the administration of the region;
• the most important organizations in the region;
• the most important events in socio-political life of the region;
• levels of power in the region;
• the scope of activities in the region.
The development of mineral resources (oil, coal, gas) for Tatarstan is very important, so as certain types of economic activity in the region. So we got several additional classes in ontology: "petrochemicals", "engineering", "agriculture", "education".
Basic classes of entities, their attributes and relationships are represented on the figure below (Fig. 2):

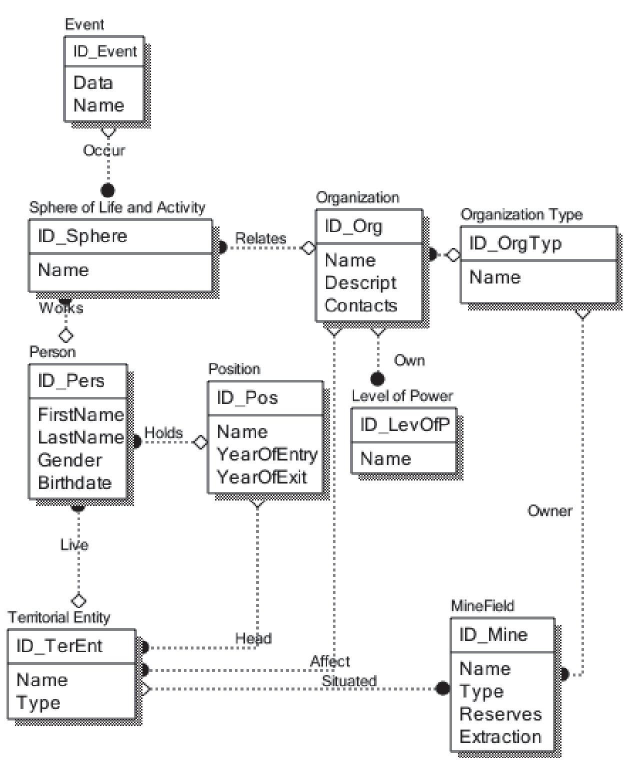


Fig. 2.

The use of ontology in the information system allows us to solve several different problems: 1) building of etalon entities to serve as models for comparison and identification of entities extracted from publications on specific SA originating from different information sources; 2) creating entities that can be presented to the user as a predefined query object to an integrated data warehouse; 3) constructing relations of reference entities that are represented in the ontology. It allows us to complement and summarize the information contained in the texts. For example, if in the text is found the reference person, it is possible to carry his name to the organization where he works (from ontology), and then generalize to the level of authority and scope of activities.
The low labor cost of automated way of creating ontologies is an advantage. The disadvantages are the lack of completeness of the ontology and sometimes random nature of allocated objects. A fully automated method of producing such ontologies is not possible yet, it still needs some work. While building the initial keyword set, analyzing the material, creating the ontologies for different training samples we had to do some manual editing (several iterations on the results of machine learning).
To improve the completeness and accuracy of the results this work was carried out using the automatic Internet surfing as a result of round the clock operation of the two servers and using crowdsourcing forces of our group, and with the involvement of undergraduate students of MTUCI [6, 10]. Crowdsourcing allowed us to significantly expand thematic corpora NL-texts formed at the initial stage. This work was carried out according to the following scheme:
• the system monitors the relevant information on the Internet of some SA;
• the user specifies a search query in the form of a key terms set (words and phrases);
• the analytical system finds some phrases on the Internet-site with specified key terms and fixes links (URL) to the relevant documents;
• out of found sentences automatically are extracted new SP that allow to expand the original search query and provide greater completeness of the information retrieved.

6 The constructing of search query

Improvement of surfing mechanisms is a necessary condition for the successful extraction of knowledge from natural language texts of very large scale (big data); improvement of these mechanisms is a separate subtask.
The used search algorithm is partially inherited from the electronic system - the encyclopedia of keywords Keywen; this algorithm produces as an intermediate result texts digests of specific subject area and looks like this. The search query is formed on the base of plurality significant phrases (SP). The search of all combinations of key terms is performed (singular terms, the terminology pairs, triples, etc.) so that the length of the query does not exceed a specific limit "n". The average number (several hundreds) of relevant documents must be found as a result. The n-value is determined

empirically. Then search queries are processed with the use of well-known search engines (Google, Yandex and other) and as a result we get a set M1 of text documents. This set is viewed concerning granting of URLs. The extended set of documents M2 is formed on these links. Further, the documents from the set M2 are divided into sentences (or fragments, similar in length to the usual offerings). As a result a database is compiled with entries of the following form:

<text fragment>-<URL pointing to this fragment>

Such a database has a significant amount of volume (up to several TB). For its storage and retrieval, it is not enough traditional means for working with databases. You must use special software and hardware technologies of Big Data processing for the purpose. Statistical estimates are obtained on the basis of the data. We calculate of how many different URLs refer to the same sentence. Thus, each fragment receives a rating equal to the number of its occurrences on independent sites. From the sorted top by rating fragments (by a certain procedure editing) is made a kind of texts digest containing rating information on this SA (without the URL).

7 Socio-political portrait of Tatarstan as an example of search query

Here are some examples of SAAP methods in various subject areas, including the resulting texts digests. Subject areas of "Socio-political portrait of Tatarstan" and "Protest activity" were taken as training samples.

Subject area: "Socio-political portrait of Tatarstan"

To create texts digest - training sample by Socio-political portrait of Tatarstan was used a search query consisting of the two components: 1) territorial formation of Tatarstan and 2) words that are characteristics of texts on social and political life. The work result of surfing semantic system was the digest of the sentences that satisfy the search query with over 40,000 sentences. Lists of officials and organizations of Tatarstan were received. These lists have been manually edited to exclude objects that are not related to the study SA.

Search query

(Tatarstan OR the Republic of Tatarstan OR Tataria OR Agryz OR Aznakaevo OR Almetyevsk OR Arsk OR Bavly OR Bugulma OR Buinsk OR Elabuga OR Zainsk OR Zelenodolsk OR Kazan OR Laishevo OR Leninogorsk OR Mamadysh OR fee OR fee OR Naberezhnye Chelny OR Nizhnekamsk OR Nurlat OR Tetyushi OR Chistopol OR Agryz district OR Aznakaevo district OR Vysokogorsky district OR state district OR Alexeyevsky district OR alkeevsky district OR Almetyevsk district OR Apastovo district OR Arsky district OR atna district OR Bavlinsky district OR Baltasi district OR Bugulma district OR Buinsk district OR Cherepovets district OR Vysokogorsky district OR erykla OR Elabuga region OR Zainsk district OR Zelenodolsky district OR kaibitzky region OR Kamsko-Ustinsky district OR Kukmor district OR Laishevsky district OR the Leninogorsk district OR Mamadysh district OR Mendeleev area OR Menzelinsk district OR its administrative district OR Nizhnekamsk district OR Novosheshminsky district OR Nurlat district OR Pestrechinsky district OR fish-Sloboda district OR Sabinsky district OR Sarmanovo district OR Spassky district OR Tetyushi district OR Tukai district OR tyulyachinsky district OR Aksubaevo OR Aktyubinsk OR Alekseevskoe OR Apastovo OR Baltaci Rich OR subs OR Vasilevo OR Jalil OR Kamskie Polyany OR Kamskoe Ustie OR Karabash OR Kuybyshevskiy Zaton OR Kukmor OR Lower Elm OR lower Maktama OR Rybnaya Sloboda OR Tenisheva OR Urussu) AND ("rating of heads of regions" OR "regional head" OR "public opinion" OR "service of state statistics" OR "national monitoring service " OR " governors rating " OR "the head of the region" OR "social wellbeing" OR "standards of living" OR "potential of the region" OR "the image of the Governor" OR "employment" OR "effective Governor" OR "media efficiency" OR "the economy")

A fragment of the texts digest

> Sozinov held meetings with students and their parents in the cities of Republic of Tatarstan, Naberezhnye Chelny, Nizhnekamsk, Yelabuga.
> Due to numerous requests of consumers in a list of cars sold under the program for the disposal, was added long-haul tractor KAMAZ M (5490).
> There is a railway line Akbash-Agryz in the Elabuga district (railway station Tikhonovo 14 km South-West of the city), the highway Kazan-Ufa, Elabuga-Mozhga.

As a result a set of texts has been found. High rank persons in the administration of Tatarstan Republic were highlighted.

Partial list of persons in Tatarstan

Person	Position
Rustam Minnikhanov	President of the Republic of Tatarstan
Ilsur Metshin Raisovich	the head of Kazan, the mayor, the Chairman of the City Duma, The Counsellor
Mintimer Sharipovich	former President of the Republic of Tatarstan
Andreeva Lyudmila Nikolaevna	Deputy Chairman of the City Duma
Gabdullina Rosalia Mirzaeva	Chairman of the Board of LLC "Kama commercial Bank"
mirgalimov Hafiz	The head of the faction of the Communist party of Tatarstan, member of the state Council of Tatarstan

A partial list of organizations of Tatarstan

State Council (state Council), Administration of Agryz, Administration Aznakayevo, Administration Almetyevsk Administration Arsk... Administration Chistopol, Agryz district Administration, the Administration Aznakaevo area, Administration Vysokogorsky district, the administration of the state district Administration... Chistopol district, Yutazinsky district Administration, Ministry of internal Affairs of Russia Kazan, OMVD Russia, Agryz... MOI of Russia Naberezhnye Chelny... OMVD Agryz district... OMVD Yutazinsky district, the Press service of the MOI of Russia Kazan, Russia, United Russia (UR) Kazan, Russia, Fair Russia, Kazan, LDPR Kazan, the Communist party of the Russian Federation Kazan, ...the Ministry For Civil Defence, Emergencies and Elimination of consequences of Natural Disasters (emergency Department) Russia Kazan,... Russia's Investigative Committee Naberezhnye Chelny,...

The subject area of " Protest activity "

Here is an example of a search query based on the selected keywords used to generate a digest of the subject area "Protest activity" using automatic systems semantic surfing.

Search query

(Activists OR Protests OR the Ruling Elite OR intra-elite Split OR a Coup OR pressure On the Government OR Demonstrations OR Euromaidan OR Seizure of Power OR Conflict of Elites OR the Crisis of Legitimate Authority OR the Maidan OR Mass Protests OR Rallies OR Dissatisfaction OR Public Opinion OR Opposition OR Picketing OR Pickets OR Political Protest OR a Protest Movement OR Protest Behavior OR the Radicalization of Protest OR a Change of Government OR Assembly OR Anxiety OR the Level of Organization OR Value Conflicts OR Processions OR Escalation of Conflict)

On the basis of the request was received this training text sample.
A fragment of the digest from the texts of SA "Protest activity"

> This Federal law is aimed at ensuring the implementation of the rights of Russian Federation citizens established by the Constitution of the Russian Federation to assemble peacefully, without weapons, hold rallies, meetings, demonstrations, marches and picketing.
> As you know, rallies of community activists and opposition under the name of Euromaidan pass in Kiev and other Ukrainian cities, which were triggered by the government's decision on the suspension of European integration November, 21, 2013.

Here is an example of an intermediate frequency dictionary on the basis of which the decision on the inclusion of the word (phrase) in a Dictionary of SP is made:

A fragment of intermediate frequency dictionary

• The MAIDAN (931)***
• ON THE MAIDAN (359)
• VECHE (171)
• OPPOSITION (591)***
• ACTIVISTS (254)***
• FOLK (92)
• POPULAR ASSEMBLY (89)
• IN KIEV (145)
• ON THE MAIDAN NEZALEZHNOSTI (87)
• RALLY (331)***
• ACTIVISTS (100)***
• MAIDAN ACTIVISTS (52)
• PROTEST (161)
• OFFENSIVE (25)
• PEACE APPROACH (18)
• ATTACK OF THE ACTIVISTS (15)
• IMPLEMENTATION OF REQUIREMENTS (15)
• DEMONSTRATION (93)***
• PROTESTS (111)***
• THE LEADERS OF THE OPPOSITION (35)
• POWER (149)
• MEMBER (41)
• PROCESSION (47)***
• FRACTIONS (31)

Here is an example of the ontology of the subject area.
The upper level fragment of the ontology of SA "Protest activity":
Objects: events (protests), people (activists, politicians), organization (the party, the opposition, Euromaidan), information sources (agencies, web sites), the time, place.
Protests: protests, demonstrations, picketing, campaign of civil disobedience, strikes, meetings, processions.

8 Conclusion

A tool for the automated generation of SAAP is a promising direction for creating systems of knowledge extraction. In addition, this tool is suitable for solving various other tasks.
A comprehensive approach to automatic processing of NL-texts based on a combination of statistical methods, corpus linguistics and distributional semantics to identify significant phrases (SP) and their association to the formation of semantic context space (SCS). It is implemented in the form of iterative methods of formation of the SAAP, an innovative and relevant research on the Russian language.
The obtained results can be used to solve the problems of the automated semantic search on very large data volumes, to extract knowledge from unstructured NL-texts and for automated generation of interactive domain-specific encyclopedias. Especially promising field of application

SAAP approach is the building of ontology, followed by the construction of database schema and knowledge.
The construction of the SAAP and the SCS using associative links between the selected terms (meaningful phrases), as well as the mapping of terms to the categories calculated by the formula cosine measures, has led to the following results:
- developing the complex of methods of text classification using methods of statistics, distributional semantics, corpus linguistics, machine associative links and some others, which allow to classify texts, initially containing the significant terms and phrases (collocations) from the training set;
- creating the software that implements the proposed methods for the selection of relevant texts on the topic of the subject area;
- forming by the volume of the electronic body correlated with a particular subject area NL-texts with automated partitioning of texts in the subject area;
- creating the body of dictionaries of terms of significant phrases (SP) and subject dictionaries, accumulated by processing large volumes of data from the Internet;
- developing the methods for creating visual maps for different subject areas with the use of multidimensional scaling" technology;
- further development of the method of semantic search NL-texts for the purpose of directed data extraction from network texts (big data) of encyclopedic information.

At present the authors' team continues to work on the development of methods of SAAP and on the creation of software tools for its implementation.

9 Acknowledgements

This work is supported by the Russian Foundation for Basic Research, grant #13-07-00272 “The methods for automatic creation of associative portraits of subject domains on the basis of big natural language texts for knowledge extraction systems”.

10 References

[1] Charnine, M. M.. V. Somin, I. P. Kuznetsov, I. Morozova, I. V. Galina, E. B. Kozerenko. Statistical mechanisms of formation of associative portraits of the subject areas on the basis of natural language texts of large amounts of systems for knowledge extraction. // Informatics and its application. 2013. Vol. 7. No. 2. Page.92-99.

[2] M. Charnine, A. Petrov, I. Kuznezov. Association-Based Identification of Internet User Interests. // Proceedings of the 2013 International Conference on Artificial Intelligence (ICAI 2013). V. I. WORLDCOMP’13. July 22-25, 2013. Las Vegas Nevada. USA. CSREA Press. C. 77-81.

[3] M. Charnine, V. Protasov. Optimal Automated Method for Collaborative Development of Universiry Curricula. // Proceedings of the 2013 International Conference on Artificial Intelligence (ICAI 2013). V. I. WORLDCOMP’13. July 22-25, 2013. Las Vegas Nevada. USA. CSREA Press. C. 96-100.

[4] M. Charnine, N. Somin and V. Nikolaev. Conceptual Text Generation Based on Key Phrases. // Proceedings of the 2014 International Conference on Artificial Intelligence (ICAI 2014). Vol.I. WORLDCOMP'14. July 21-24, 2014. Las Vegas Nevada, USA. CSREA Press. C. 639-643.

[5] O. Zolotarev, M. Charnine, A. Matskevich. Conceptual Business Process Structuring by Extracting Knowledge from Natural Language Texts. // Proceedings of the 2014 International Conference on Artificial Intelligence (ICAI 2014). Vol.I. WORLDCOMP'14. July 21-24, 2014. Las Vegas Nevada, USA. CSREA Press. C.82-87.

[6] V. Protasov, M. Charnine, and E. Melnikov. The Crowdsourcing Linguistic Technology for Experts Assessment. // Proceedings of the 2014 International Conference on Artificial Intelligence (ICAI 2014). Vol.I. WORLDCOMP'14. July 21-24, 2014. Las Vegas Nevada, USA. CSREA Press. C. 656-661.

[7] T. N. Borisov, A. E. Bronetz, S. V. Klimenko, V. V. Rykov, M.M. Charnine.. Autonomous underwater vehicles: automatic generation of associative portrait subject area. // Proceedings of International scientific conference "Situational centers and IAS for monitoring and security. SCVRT2013-14". Moscow-Protvino. Publishing house of the Institute of physics and technology (ICPT). 2013-2014. P. 38-43.

[8] Charnine M. M., Somin N. In., Galina I. V., Rodina I. V. Meaningful generate: a method of generating informative texts of relevant phrases. // Proceedings of the XIX international conference CPT'2014. Larnaca, Cyprus. may 12-18, 2014. Publishing house of the Institute of physics and technology (ICPT) (in press).

[9] O. Zolotarev A., M. Charnine. Methods of knowledge extraction from natural language texts and the construction of models of business processes on the basis of allocation processes, objects, their relationships and characteristics. // Proceedings of the XIX international conference CPT'2014. Larnaca, Cyprus, may 12-18, 2014. Publishing house of the Institute of physics and technology (ICPT) (in press).

[10] Johnson, O., M. Charnine. Methods of knowledge extraction from natural language texts and the construction of models of business processes on the basis of allocation processes, objects, their relationships and characteristics. // Proceedings of the XIX international conference CPT'2014. Larnaca, Cyprus, may 12-18, 2014. Publishing house of the Institute of physics and technology (ICPT) (in press).

[11] Protasov, V. I., Charnine M. M., Melnikov, I. E. the Use of technology self-managed crowdsourcing for certification experts. // Proceedings of International scientific conference "Situational centers and IAS for monitoring and security.

SCVRT2013-14". Moscow-Protvino. Publishing house of the Institute of physics and technology (ICPT). 2013-2014. P. 262-266.

[12] Charnine M. M., Rodina I. V. Mechanisms of formation of linguistic and statistical portrait of the subject area monitoring of public opinion for the technology of big data. // Proceedings of the Third International scientific-practical conference "Social computing: fundamentals, technology, Humanities and social effects" (ISC-14)". September 18-19, 2014 RITZ MGGU. Sholokhov. P. 66-70.

[13] O. Zolotarev, O. Kozerenko E. B., M. Charnine. Principles of construction of models of business processes of the subject area based on the processing of natural language texts // Bulletin of ROSNOU. - M.: RosNOU, 2014. No. 4. P. 82-88.

[14] O. Zolotarev, Process approach to the management of projects in the implementation of corporate information systems. // Bulletin of ROSNOU. - M.: RosNOU, 2014. No. 4. P. 89-92.

[15] O. Zolotarev. Methods of extraction processes, objects, relations from natural language texts. // Security problems of the Russian society. - Smolensk: Svitok. 2014.

[16] O. Zolotarev. Control in implementation projects of distributed Enterprise Resource Planning Systems. Bulletin of the Russian New University, Moscow, RosNOU, 2012. No. 4. P. 78-80.

[17] O. Zolotarev. Innovative solutions in the formation of functional structure of the subject domain. Bulletin of the Russian New University, Moscow, RosNOU, 2013. No. 4. P. 82-84.

Business Intelligence Processing on the Base of Unstructured Information Analysis from Different Sources Including Mass Media and Internet

O. Zolotarev[1], M. Charnine[2], A. Matskevich[2], K. Kuznetsov[2]
[1]Russian New University, Moscow, Russia
[2]Institute of Informatics Problems, Federal Research Center, Russian Academy of Sciences, Moscow, Russia

Abstract - This article describes approaches to the business intelligent processing on the base of unstructured information analysis. The problem of reputation management of companies is considered. The review of linguistic, statistic and semantic instruments of text analysis is presented. Several examples of natural language texts analysis are discussed. The information is extracted from the natural language texts from different sources. Entities, relations and processes are built in the form of semantic network. The questions of model constructing of a subject domain are discussed. The models are presented as fragments of semantic network built on the base of the declarative instrument DECL.

Keywords: Business Intelligence Processing, Business process, semantic networks, fragments of knowledge, objects, thesaurus, Big Data

1 Introduction

The functioning of a modern enterprise is connected with challenges and risks, both external and internal. Being in a tough competition, companies are forced to use a variety of analytical and intellectual methods to protect their reputation, to save the confidential information, to identify different kinds of leaks of documents, to minimize the potential damage from unauthorized activity on the part of their employees and the representatives of competing firms. For this purpose data is obtained from a variety of sources, including the Internet. Significant processes, activities and events are tracked; the links between individuals are checked. Related companies with different kinds of information resources are determined by the degree of participation of their individuals in different actions that may harm the enterprise.

Ongoing studies are considered from the point of view of tactical and strategic goals. The modern approach to the analysis of company's activity is based not only on obtaining direct information about its work, enterprise processes, documentation, monitoring the work of the enterprise, but also it is based on a serious study of open-source information. Every year Internet has more and more influence on this process. The most important things are information analysis, the formation of conclusions based on the results of the analysis and the information filtering due to the huge volumes of data in the public domain. Of particular importance is the information posted on the Internet, because in this case, we can process it automatically that couldn't be possible until the information resources were transferred into electronic format.

To reduce the space of search for relevant information the subject area associative portraits (SAAP), subject-specific dictionaries (SSD) and thesauri are generated [7]. As a result the knowledge base is formed. This allows us to identify certain objects and processes of the subject area, to significantly increase the usefulness retrieved from the Internet information. For this purpose, texts retrieved from the Internet are processed with the linguistic processor [1].

As a result of information processing there are a large amount of data for SAAP, thesauri and subject dictionaries that are constantly replenished with new data improving the quality of filtration of texts from the Internet to improve the reliability of analytical results.

The analytical research of the company background can be done by a group of analysts who explore the results of automatic processing of information from the following sources [2]:

- information from the Internet;
- materials from various databases;
- materials of analytical centers;
- electronic documents of the company.

There are significant issues in automatic processing of data from public sources:

- huge amounts of information including spam;
- difficulty in identifying specific objects, processes, situations on the basis of analysis and comparison of information from different sources;
- unreliability of data;
- explicit and implicit misinformation;
- incomplete data;
- constant volatility of environment, situations;
- historical variability;
- data aging, etc.

As a result of continuous improvement of the recognition mechanism of identification of objects and processes, and also because of the accumulation of knowledge that can be measured against the experts, the efficiency of the process of determining leaks and identifying the actions of individual actors, directed against the organization, can be substantially improved [3].

One of the objectives of any enterprise is the task of forming a positive image of the company. For this purpose can be used the results of the analysis of the external and internal environment of the enterprise to form purposeful influences on the organization background for the correcting its image.

The company's reputation is a kind of evaluation of a group of individuals about the company, group of people or individual on the basis of certain criteria. Moreover, there is a clear dependence of the capital of the company from its reputation. Goodwill or Reputational Capital is directly related to the financial condition of the company [14]. One of the goals of the company is minimizing the amount of negative information about the firm in Internet. Now there are a lot of companies that offer Internet services to correct negative image of the organization, for example, Online Reputation Management (ORM) companies and Search Engine Reputation Management (SERM) companies [4-6].

2 Linguistic, statistic and semantic instruments of text analysis

Currently the market offers a large variety of tools for analyzing texts. There are a lot of informational instruments for Internet research called textual-analytical tools or processors of data collection. There are quite a number of companies that offer tools for analyzing texts on the Internet aimed at identifying leaks of information, analysis of reputation, etc. Below is a description of some of these tools.

The InfoTracer program packet represents the toolkit for online public records searches. This firm is a member of the Private Investigator Union proposes several types of searches such as Comprehensive Background Report, Criminal Records Search, People Search, Property Search, Email Search, Company Background Search and so on. The most interesting thing about InfoTracer searching is the discovering of criminal backgrounds, past addresses, possible relatives, property ownership information, public profile information, registration information, birth, death, marriage and divorce record information, citizenship or even marital status.

Taiga is the French processor for extracting information within Internet from patent databases, news reports and publications on scientific conferences. It allows gaining valuable information about the most promising scientific developments and areas where competition is relatively small so far. The system analyzes not only text, but also drawings, diagrams and graphs. It collects any requested information as soon as it appeared on the Internet. The information about links between two countries can be discovered in a few hours.

Tropes software suggests high performance text analysis. It’s designed for semantic classification, keyword extraction, linguistic and qualitative analysis.

Tropes can detect contexts, themes and principal actors, through the application. This system can extract objects and subjects, time, place and purpose from the situation. Tropes carries out a chronological analysis of a text with the help of Natural Language Ontology Manager based on Semantic Networks and Natural Language Text Analysis technologies, supplied with several ready-to-use classifications.

It’s a searching engine that includes: topic summaries, categories, disambiguation, official sites. It can be used to define people, places, things, words and concepts; to provide direct links to other services; to list related topics; and to give official sites when available.

This company aims to provide intelligent systems based on Semantic Technologies and Big Data. Product Features: Automated semantic sentiment analysis for brand/products; Trend analysis and consumer profiling; Broad coverage and customization support; Semantic Search (Search by meaning not by keyword).

The urgency of the task of analytical intelligence is not in doubt. A large number of groups engaged in similar issues. Using methods of semantic analysis is practiced all over the world, which allows substantially to protect the enterprise from external and internal threats and to manage the organization's reputation in the media.

3 Texts procession, Information retrieval and related objects allocation

Now for the formalization of domain knowledge and building the structure of the business processes are used different methods of knowledge representation. This article discusses an approach for the formation of domain knowledge in the form of a semantic network, which is represented as nodes and relations between them [13]. Usually in the form of nodes are represented certain objects (subjects or entities) of the subject area (SA) [8], and in the form of relationship - operations, in which data objects involved or links between these objects. A semantic network is described in the form of fragments [1,2,3,9]. Below presented fragments of a semantic network describing the abstract objects and relationships between them:

R1(A1,A2/N1) R2(A3,A4/N2) R3(A5,A6/N3) (1)

Here R1,R2,R3 are the names of relations;

N1, N2, N3 - are the names of fragments (N1 - the name of the whole fragment R1(A1,A2); thus it can be omitted, if not required while processing);

A1 - A6 - object names.

For example, as a result of analysis of information from the Internet, there was extracted the following text fragment:

"Bondarev took part in a project to purchase equipment for the company Nogos". For convenience all the insignificant details were omitted (name and patronymic). As a result of the processing of this phrase is creating a corresponding fragment of a semantic network [10]:

Purchase of equipment (Nogos, Bondarev) (2)

Thus, it is known that Bondarev does not work at the Nogos enterprise and can't be formally involved in the project for purchasing equipment. But miss Kravtsova works at the Nogos company and she is a sister to Bondarev. In this case, the following information is placed in the system in the form of a semantic network (or the network should be built on the basis of the analysis of text information of the enterprise)[11]:

Sister(Kravtsova, Bondarev) (3)

Worker(Nogos, Kravtsova) (4)

A given set of predicates corresponds to the following fragment of a semantic network (Fig. 1):

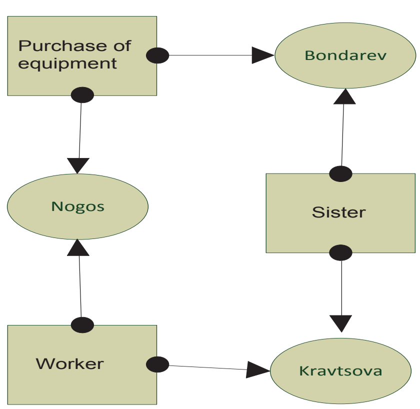


Fig. 1. A fragment of semantic network (1) - (4)

So we can conclude that the company's employee Kravtsova used his family ties to promote a project, either for personal gain from the transaction of funds or as a contractor of the project of purchasing the equipment.

In this case, we can talk about the using of neighborhood approach to the analysis of knowledge. Every object has its own surrounding with other objects (entities) or nodes in the framework of a semantic network. The neighborhood of the object is defined by the presence of "close" semantic relations. There are several levels of proximity within the vicinity of the object - the first, the second, etc. For example, for the object "Kravtsova" mr. Bondarev (her brother) is adjudged to be in the neighborhood of the first order, i.e. they are connected directly through one of the elements of a semantic network. In this case it was done by means of fragment Sister(Kravtsova, Bondarev). These fragments correlate with those objects built in the system ontology, which contain information on the relationships (sister, brother), classes of objects (classes of people). In addition, if the ontology has already the described above information (about Bondarev, Kravtsova, their family relationships), then identification of objects is done, for example, there are already a lot of objects from Internet included in the ontology, which substantially simplifies the situation recognition [12].

The described above case can be considered as ideal, in other words, in this situation, there is no difficulty with doing the conclusions. But there are much more complicated situations, when the analysis of the semantic proximity is done not only by family ties and indirect ties but with other aspects too. Consider the following example, making changes in the situation:

"Mt. Petrov took part in a project to purchase equipment for the company Nogos" and we have no direct information about the relationships of Mr. Petrov. If there is information on the Internet, in social networks, that Petrov is a friend of Bondarev, then will be built the following semantic network fragments:

Sister(Kravtsova, Bondarev) (3)

Worker(Nogos, Kravtsova) (4)

Purchase of equipment (Nogos, Petrov) (5)

Friend(Petrov, Bondarev) (6)

As a result, we have the following. The information leaked from the enterprise, perhaps in this case, the company incurred additional losses in the form of payments to the person concerned, or higher prices for the equipment. The situation is already described by the second order neighborhood where there are no direct links between objects. As a proxy fragment is the predicate "Friend".
A given set of predicates corresponds to the following fragment of a semantic network (Fig. 2):

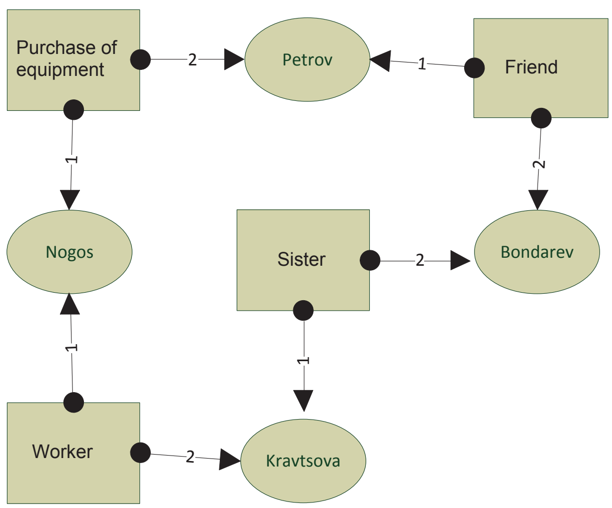


Fig. 2. A fragment of semantic network (1) - (6)

Neighborhoods of vertices can be based on different grounds when there are not direct relationships between objects. You can determine the presence or absence of communication between objects in a number of ways, for example:

- by finding objects in the same place (several times);
- by finding objects in the same place at the same time (several times);
- by training in the same school;
- by occurrence in the same organization (club, car);
- by the similarity of interests;
- by localization of the place of residence or work (in a broad sense).

This situation can be augmented with additional information found on the Internet, if two objects are in the same place at the same time, and can be described by fragment type:

Place(PlaceName, Object, Month, Year) (7)

For example, if there was extracted the text like this "Petrov had a vacation in June 2015 in Sochi" and at the same time there was extracted the following text "Bondarev usually has a vacation in Sochi in August, then we can add to fragments already constructed two more fragments (excluding the fragment Friend(Petrov, Bondarev)):

Place(Sochi, Petrov, June, _) (8)

Place(Sochi, Bondarev, June, 2015) (9),

This will provide some connection between these two individuals in case if Petrov takes part in the procurement project tj buy the equipment (see above). As a result will be built the following semantic network (in this fragment presents a synthesis of information from the ontology of the subject area) (Fig. 3):

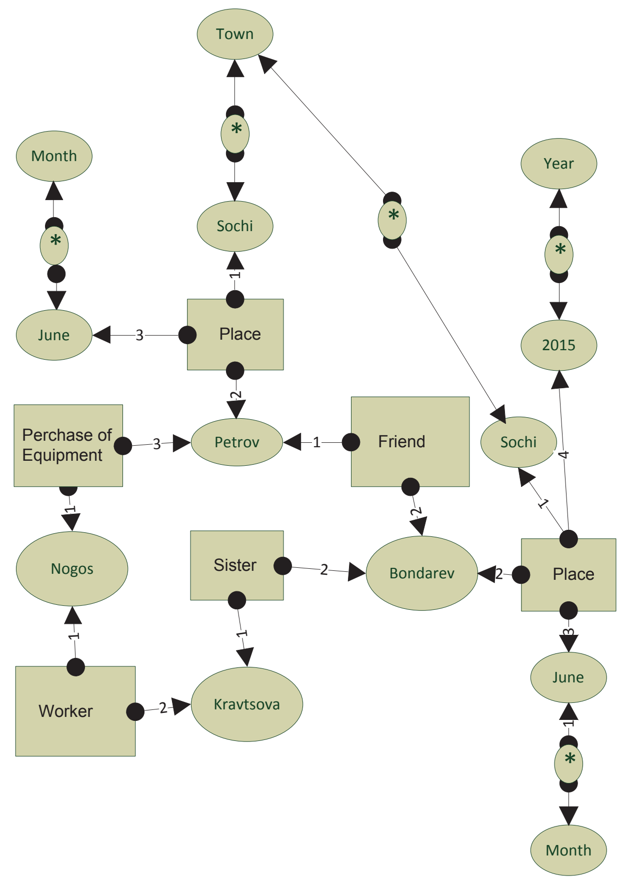


Fig. 3. A fragment of semantic network (1) - (9)

The following fragment presents a semantic network fragment with the information from ontology in the form of a series of generalizing concepts (categories), namely, "month", "year", "city", which was described above in verbal form. Such categories forms polyhierarchy and it is extraordinarily useful for performing inference in the process of natural language processing. A particularly interesting special case of polyhierarchy categories, which is allocated in the main tree contains all vertices of polyhierarchy. This polyhierarchy with a dedicated tree is called Keywen-hierarchy. Keywen hierarchy allows to find higher categories for each concept, which increases the flexibility of the representation of the categories system. At the same time Keywen hierarchy allows us to construct an unambiguous highlighted path to the root from all the nodes of the hierarchy, for example, Russia > city>Sochi. The pathway reminds address that's easy to navigate and to control the correctness of the structure of categories. Methods for constructing Keywen-category hierarchy described in [7,8].

4 Conclusion

The discussed in this article approach allows not only to make the organization more secure, but also to determine the most relevant and promising directions of its development. We are not only interested in people's actions that can harm the business, but with the opportunities as a result of changes in the environment. New interesting directions are revealed. New firms and stakeholders are extracted from specific subject areas, with analysis of the connection between them and areas for development. Priority activities are detected. The analysis highlighted the strategic goal of enterprise development [15].

Constant monitoring of open source allows us to gain advantages in the following aspects:

- analysis of possible risks and opportunities for development;
- development of a proactive plan of action to win competitors;
- identifying new areas of development,
- detecting the emergence of new competitors;
- analysis of the reputation of the company in the community;
- definition of channels of information leakage;
- formation of positive image of the company;
- identifying allies and competitors.

5 Acknowledgements

This work is supported by the Russian Science Foundation, grant #15-11-30040, and by Russian Foundation for Basic Research, grant #13-07-00272 “The methods for automatic creation of associative portraits of subject domains on the basis of big natural language texts for knowledge extraction systems”.

6 References

[1] Kuznetsov I.P. Semantic Representations // Moscow: "Nauka", 1986. 290p.

[2] Kuznetsov I.P., Matskevich A.G. The System for Extracting Semantic Information from Natural Language Texts // Proceedings of the Dialog International Workshop "Computational Linguistics and its Applications", Vol.2, Moscow: Nauka, 2002.

[3] Kuznetsov I.P. Natural Language Texts Processing Employing the Knowledge Base Technology // Sistemy i Sredstva Informatiki, Vol.13, Moscow: Nauka, 2003, pp. 241-250.

[4] Somin N.V., Solovyova N.S., Charnine M.M The System for Morphological Analysis: the Experience of Employment and Modification // Sistemy i Sredstva Informatiki, Vol. 15 Moscow: Nauka, 2005, pp. 20-30.

[5] Web site “Knowledge extraction for Analytical Systems”: http://Iiranlogos.com/english/

[6] Kuznetsov, I.P., Kozerenko E.B. Semantic Approach to Explicit and Implicit Knowledge Extraction // Proceedings of ICAI’11, WORLDCOMP’11, July 18-21, 2011, Las Vegas, Nevada, USA. - CRSEA Press,
USA, 2011.

[7] Charnine, Michael Vladimir Charnine. Keywen Category Structure.// Wordclay, USA, 2008, pp.1-60.

[8] Charnine, M. M., “Keywen: Automated Writing Tools”, Booktango, USA, 2013, ISBN 978-1-46892-205-9.

[9] Charnine, M.M., I. P. Kuznetsov, E. B. Kozerenko,. Technological peculiarity of knowledge extraction for logical-analytical systems. WORLDCOMP’12 July 16-19, 2012. Las Vegas, USA.// CSREA Press, pp. 49-55, 2012.

[10] Zolotarev, O., M. Charnine, A. Matskevich. Conceptual Business Process Structuring by Extracting Knowledge from Natural Language Texts. Proceedings of the 2014 International Conference on Artificial Intelligence (ICAI 2014), vol.I, WORLDCOMP’14, July 21-24, 2014. Las Vegas Nevada, USA. CSREA Press, pp.82-87.

[11] Zolotarev, O., Kozerenko E. B., M. Sharnin. Principles of construction of models of business processes of the subject area based on the processing of natural language texts // Bulletin of ROSNOU. - M.: RosNOU, 2014. No. 4. P. 82-88.

[12] Zolotarev, O. Process approach to the management of projects in the implementation of corporate information systems. // Bulletin of ROSNOU. - M.: RosNOU, 2014. No. 4. P. 89-92.

[13] Zolotarev, O. Methods of extraction processes, objects, relations from natural language texts. // Security problems of the Russian society. - Smolensk: Svitok. 2014.

[14] Zolotarev, O. Control in implementation projects of distributed Enterprise Resource Planning Systems. Bulletin of the Russian New University, Moscow, RosNOU, 2012. No. 4. P. 78-80.

[15] Zolotarev, O. Innovative solutions in the formation of functional structure of the subject domain. Bulletin of the Russian New University, Moscow, RosNOU, 2013. No. 4. P. 82-84.

Evaluation for morphologically rich language: Russian NLP

S. Toldova[1], O. Lyashevskaya[1], A. Bonch-Osmolovskaya[1] and M. Ionov[2]
[1]School of Linguistics, National Research University "Higher School of Economics", Moscow, Russia
[2]Department of Theoretical and Applied Linguistics, Moscow State University, Moscow, Russia

Abstract - *RU-EVAL is a biennial event organized in order to estimate the state of the art in Russian NLP resources, methods and toolkits and to compare various methods and principles implemented for Russian. Russian could be treated as an under-resourced language due to the lack of free distributable gold standard corpora for different NLP tasks* (*each team tried to work out their own standards*)*. Thus, our goal was to work out the uniform basis for comparison of systems based on different theoretical and engineering approaches, to build evaluation resources, to provide a flexible system of evaluation in order to differentiate between non-acceptable and linguistically "admissible" errors. The paper reports on three events devoted to morphological tagging, dependency parsing and anaphora resolution, respectively.*

Keywords: NLP resources, evaluation, Russian, morphological taggers, dependency parsing, anaphora resolution.

1. Introduction

The NLP evaluation forum RU–EVAL started in 2010. A strong motivation for initiating the event was the need to independently review the stateoftheart of pos-taggers, parsers, and other NLP modules for Russian. The evaluation campaign is open both to academic institutions and industrial companies, and its major objective is to assess the current trends in the field and to promote the development of NLP technologies. This paper presents three evaluation campaigns held in 2010, 2012 and 2014, their task sets and results.

Although Russian computational linguistics history started more than 50 years ago (cf. the first MT research in 1955, Bar-Hilel 1960), up to the beginning of the 21 century the NLP teams, both research groups which inherited the Soviet tradition and new commercial industry labs, had been working as isolated units, with no interaction among them. The evaluation of standard NLP tasks such as pos-tagging, lemmatization, parsing, etc. requires certain unified standards and principles of annotation. In this respect, Russian could be treated as an under-resourced language since there were no free distributable gold standard corpora for various tasks. The problem was that each team tried to work out their own standards. The disjoint development of Russian NLP teams led to the co-existence of many different theoretical and engineering approaches. This plurality in approaches, tagsets, annotation principles, etc. makes the task of the evaluation very difficult. The paper summarizes our experience in working out the uniform basis for comparing different systems, building gold-standard evaluation resources, as well as providing flexible protocols for evaluation. The comparison of different approaches as well as working out unified standards reveal the controversial issues related to the nature of language data that are hardly amenable to formalization. For this reason we differentiate between non-acceptable and linguistically "admissible" errors which is helpful for determining the bottlenecks in language modeling for different levels of language description.

The paper is structured as follows. The remainder of Section 1 reviews the aims of the evaluation events for Russian, the topics and participants of the RU-EVAL tracks, and resources created by these events. Then we discuss the events on morphological tagging (Section 2), dependency parsing (Section 3), and anaphoric and co-reference relations detection (Section 4). Section 6 concludes.

1.1. Background

In the last decades, a special attention has been given to the evaluation of NLP systems. One of the main aims for such events is to provide the general grounds for independent evaluation via suggesting shared tasks for various NLP modules. The influence of such events on the NLP technology progress is indispensable. Among them are CLEF [1], Morpho Challenge [2], GRACE (for French, see [3]), Senseval/Semeval [4], MaltEval (for parsing, see [5]), CoNLL (for parsing, Named Entities recognition, anaphora and coreference resolution, [6]), ACE program [7], MUC7 [8], ARE (Anaphora Resolution Exercise, see [9]), etc. The majority of tasks are based on the testing sets for English (training and testing collections annotated according to shared tasks conditions, evaluation metrics and evaluation scripts etc.), though there are regular events for some other languages such as French (EASY) and Italian (EVALITA, [10]) as well as minor languages involved into multilingual task events. While preparing the testing sets evaluation procedures and resources for RU-EVAL events we took into consideration the experience of corresponding events (EVALITA as the most close to our conditions). However, we had to modify the tasks and evaluation principles for we have no opportunity to rely upon resources for Russian language for they are not easily accessible for the majority of NLP teams and are not recognized by the majority as standards.

1.2. Aims of the evaluation campaigns for Russian: state of the art

As it has been mentioned above there are a lot of teams working with Russian language both those that are working on NLP tasks for more than 20-30 years and just newly organized start-ups, scientific research groups and business companies as well as educational groups from High School institutions. The majority of such teams are working in disjoint modes. Many of these teams start new NLP modules for Russian from the ground up.

The disjoint mode of development has led to the high diversification of standards and annotation schemes used in different NLP tasks by different teams. Thus, for instance we count more than 1000 different labels for dependencies for seven participants. The overlap in syntax tagsets was three-four tags only.

When the RU-EVAL campaign started in 2010 there were no generally distributed test collections, which could serve as a basis for systems comparison. There were some available resources (e.g. Russian National Corpus) or such morphological taggers as AOT or MyStem[1]. However, there was a need in a gold standard collection that consider discrepancies in tagging principles and allow comparison of systems based on quite dissimilar theoretical assumptions.

Thus, the aims of RU-EVAL events are: to consolidate the isolated NLP teams dealing with Russian language; to suggest gold standard collection that could serve the basis for comparison; to suggest principles of annotation for different tasks in Russian; to enumerate the discrepancies in existing theoretical and practical approaches; to suggest an evaluation scheme; to measure the basic level for different NLP tasks for Russian.

1.3. RU-EVAL events

In this perspective, the aim of the RUEVAL initiative is to promote the development of language technologies for the Russian language, by providing a shared framework to evaluate different systems and approaches in a consistent manner.

The NLP Evaluation forum RU–EVAL started in 2010. The participants are both academic teams and industrial companies. Organized on a fully voluntary basis, RU-EVAL was aimed at systematically proposing standards for Russian starting with the lower levels of linguistic analysis. The first three events were devoted to morphological tagging (2010, see [11]), parsing (2012, see [12]), anaphora and coreference Resolution (2014, see [13]).

[1] It's worth mentioning the Open Corpora project that nowadays suggests an open collection for morphological tagging training and evaluation.

The first NLP Evaluation forum focused on morphological taggers (see http://ru-eval.ru), bringing together 15 participants from Moscow, Saint-Petersburg, Yekaterinburg, Ukraine, Belarus and UK. We had seven different tasks in total. These were four tasks for tagging without disambiguation (lemmas, pos-tags, full set of grammatical tags and a special track for rare words) and three tasks for tagging with disambiguation (lemmas, pos-tags, full sets). The number of participants for each track is shown in Table 1.

Table 1. Participants of RU-EVAL

Task	Number of participants	Year
Lemmatisation	13	2010
Pos-tagging	13	2010
Full gram. Set	12	2010
Rare words	8	2010
Lemmatisation (disambig)	7	2010
Pos-tagging (disambig)	7	2010
Dependency parcing	7	2012
Anaphora	7	2014
Coreference	3	2014

In 2011-2012, syntactic parsing technologies were evaluated. Eleven participants expressed their interest in participation, seven of them submitted their answers. The task was to submit dependency parsing results. Only relations (head-dependent were taken into consideration) irrespective of assigned relation labels. There were two tracks depending on text types: general text collection and the News subcorpus.

The last event (2014) was devoted to the tasks of anaphora and coreference resolution and had seven participants in total. All of them submitted the anaphora resolution results and three of them participated in the coreference resolution track.

All the events had the preliminary discussions of evaluation details (corpora, formats, standards) with prospective participants. During the «Dialog-2011» (International Conference on Computational Linguistics in Russia[2]), a meeting on problems of syntactic parsers evaluation with the leading experts in the field was organized. In 2013 the round-table on anaphora and coreference resolution took place.

The complete cycle of the forum starts with working out mark-up scheme of the Gold Standard via analyzing the international practice of similar evaluation events, testing annotation schemes for Russian provided by potential participants, evaluating the inter-annotator agreement on preliminary test sets and ending with the final paper preparation. Our forum has an educational

[2] http://www.dialog-21.ru/en/

component: the expert group includes students who plan to work in the field of computational linguistics [14]. The evaluation cycle serves as a basis for a course in computational linguistics. It is worth mentioning that students not only do the routine assessment procedure, but take part in creating the forum design.

To sum up, the RU-EVAL has brought together a considerable number of IT companies and academic groups that work on Russian, and made it possible to assess the state-of-the-art in the field (so far, mostly in Russia).

1.4. Gold standard resources and evaluation schemes

As results of three RU-EVAL campaigns three sets of resources were created.

For each campaign the testing corpora include texts of different genres. Corpora consisted of fiction, news, non-fiction (science, law etc.) and texts from social networks (5%). We have 1 million tokens test sets for each of events. The gold standard sets are tagged manually by two annotators, the discrepancies being discussed and checked by a supervisor.

For morphology tagging we have a gold standard set of 3316 tokens and a testing list of rare words. For parsing we used manually annotated collection of 600 sentence (16000 tokens). The anaphora/coreference gold standard corpus at present consists of 185 texts (97 texts were used as learning set), containing 199681 tokens. The text length was from 5 up to 100 sentences, the longest one being 170 sentences long. These texts include 2900 anaphoric chains with 14405 total elements were annotated.

The forum devoted to parsers (RU-EVAL'2012) suggested the evaluation of existing syntactic parsers. It brought about several significant outcomes such as creation of a manually tagged and assessed gold standard treebank (800 sentences, available freely from http://rtb.maimbava.net/res01/rtb.php), treebank with parallel (1 mln. tokens, annotated by four participants, available from http://rtb.maimbava.net/res01/rtb.php). Besides the variations in theoretical and practical decisions between existing parsers have been analyzed.

The coreference/anaphora gold standard texts were split into sentences, pos-tagged with TreeTagger for Russian (we used a TreeTagger-based ([15]) part-of-speech tagger, a lemmatizer based on CSTLemma [16]. The learning set corpora is available from http://ant1.maimbava.net/.

Below we discuss each NLP task in more details from the point of view of language specific features and theoretical traditions which influence the development of NLP for Russian.

2. Morphology tagging

2.1. Russian as a morphologically rich language

The controversial issues we faced while working out the evaluation routine for Russian could be explained primarily by the fact that Russian like other Slavic languages is a morphologically rich language with a rather free word order. It is well known that the morphological richness increases the complexity of tagging [15]. Russian has a considerably large morphological tagset (cf. more than 4 592 unique full tags reported in [15], from which the top-1000 tags have each more than 40 occurrences in the 6M corpus [17]; cf. also 829 simplified tags used in [18]). There are 6 to 12 forms for nouns, ca. 30 forms for adjectives and more than 160 synthetic forms for verbs, including adjective-like participles.

Besides that, Slavic languages are inflectional with high index of fusion [19]. A bulk of grammatical categories is encoded in one short affix and this leads to high index of potential homonymy of word-forms. The word order is not as helpful in this case as in English and other German languages, for it is rather free in Russian, see Section 3.

Taking into consideration the diversity of existing taggers (see Section 1.2) the preparation stage of the event included their comparison. While comparing various approaches we come across the following linguistically motivated issues that need special treatment.

The size of verb paradigm could vary depending on whether two stems are organized into one paradigm or not. Many categories differentiate morphosyntactic classes and, thus, pertain to a stem and not to a certain morpheme. For example, there is no regular affixation for expressing aspect (imperfective VS perfective) in verbs, so aspect is a characteristic of a stem. Thus, the systems vary in whether each stem is lemmatized as a separate item, or a pair of stems are assign to one lemma of a fixed aspect (for some systems it is imperfective variant and for others perfective variant). The lemmatization of the so-called reflexive verb pairs is also non-trivial for some of the lexemes have regular pairs and are united under one lemma while in other cases the semantic relation between a verb and its derivative with the reflexive affix *–sya* is dubious.

The principles of pos-assignment could differ depending on the theoretical assumptions whether a pos-tag is determined by a lexeme paradigm (a set of affixes) or a word syntactic position. Thus, there is a sufficient variation in pronouns pos-tagging. For this reason, we do not take this class into account in evaluation procedure. One more complicated issue as far as the pos-tagging standard is concerned is the differentiation of conversion cases and the regular forms with the corresponding syntactic function: e.g. participle vs. "verbal" adjectives, adverbs vs. predicatives etc. The latter oppositions present a problem even for manual tagging.

The main problem both for theoretical modeling and for the method choice (c.f. context-based methods such as HMM or rule-based methods based on morphological parsing) is the need of optimal balance between context and word-structure criteria.

2.2. Gold standard annotation and evaluation principles

The majority of Russian systems use rule-based methods without disambiguation since they disambiguate POS-tags in parsing modules, if needed. For this reason, we have four tracks for tagging without disambiguation.

We use a simplified list of parts of speech: separate parts of speech for inflected words such as Noun, Verb, Adjective, two syntactically defined classes, namely, Conjunction and Preposition, and ADV that includes adverbs, particles and other non-inflected words. We did not take into consideration pronouns tagging. We reduced the number of problematic stem-based tags such as aspect and voice for verbs (excluding voice for participles) and we do not take an opposition short vs. brief form for adjectives.

We had the only one tag for a token in gold standard. A system response was considered as true positive in case any of those given for the token matched the gold standard tag. We use accuracy as an evaluation measure.

2.3. Rare words track

A special track on rare words presupposed analysis of tokens which are, with a high probability, not included in the grammatical dictionaries of the dictionary-based systems. We put sentences with rare words into the common test set but evaluate responses separately.

A special word set for testing was comprised of 75 words. They represented the following classes of rare words: (a) those referring to productive word-formation types and rhyming false stimuli, eg. *frendjata* ‘little friends’, lemma *frendjonok* vs. *arrabjata* ‘arrabiata’ and those invented by authors, eg. *uvazila, slipkix*; (b) complex words with a dictionary-based second part, eg. *ul’trazhensvennoj* ‘ultra - feminine’; (c) simple stem words with standard inflection affixes, eg. *turbijona* ‘tourbillon.Gen.sg’; (d) rare and substandard forms of declination, eg. *visju* ‘I hang’, *derevjannee* ‘more wooden’; (e) abbreviations.

2.4. Results and discussion

The general level of lemmatization and pos-tagging was considerably high for tagging without disambiguation. The median for lemmas task was 96.5 (max 99.3, min 72.8), pos-tagging 97.1 (max 99.4, min 72.8) and full tagging 94.8 (max 97.3, min 31.9). There were predictably lower scores on the rare words track, the median being 62 with maximum 72 and 4 as minimum.

Figure 1 shows results of tagging with disambiguation, the median being 94.5 for lemmas and 95 for pos-tagging.

The analysis of typical mistakes and systems drawbacks revealed some more language features that presented additional problems for highly inflected free-word order languages.

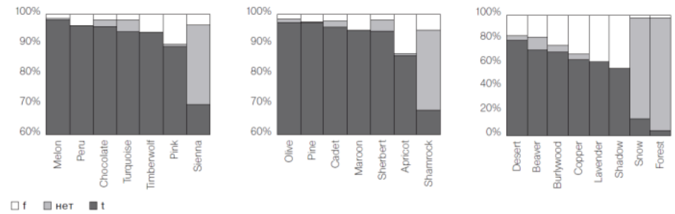


Figure 1

On the one hand, the rich inflectional system reduces the homonymy caused by conversion. On the other hand, it induces new homonymy types. Thus, there are many regular cases of grammatical forms homonymy such as genitive of masculine vs. nominative of feminine. Moreover, many systems generate potential forms that are never met in real texts (e.g. Russian preposition *dlya* ‘for’ is homonymous to a converb of the verb *dlit*’ ‘to prolong’, that hardly could occur in real texts, it never occurs in National Russian Corpus). Consequently, the resulting tagging have a high coefficient of homonymy, e.g. 2.5-3 tags per token.

The majority of systems build up their analysis relying on inner word structure. The typical mistakes are the wrong analysis based on wrong detection of morpheme boundaries or matching it to a wrong morpheme annotation. Another class of multiple mistakes concerns a case when a token with clear morphological structure (e.g. adjectival suffixes) occupies a syntactic position of another part of speech (e.g. substantivized nouns, surnames with adjectival inflection system in Russian, etc.).

3. Dependency parsing

3.1. Challenges for the Russian parsing

The second forum took place in 2011-2012 and it was dedicated to syntax parsing.

Several properties of the Russian syntax make the syntactic parsing more difficult compared to English. The most important one is the free word order. In fact, word order (e.g. the order of major constituents such as subject, object) is mostly triggered by information flow (e.g. topic-focus hierarchy, prominence of participants in a profiled frame, emphasis, etc.). However, the order within individual constituents is more fixed, e.g. demonstratives and numerals usually precede nouns (but not always).

Subject is not obligatory in a finite clause in Russian, e.g. there are a lot of different types of impersonal constructions (c.f. [16] where adaptation of Universal set for dependency annotation to Slavic languages is discussed). A finite verb also could be omitted in a sentence (c.f. zero copula in the Present Tense). In this case it is unclear what could be chosen as a root for a syntactic tree. There are constructions in Russian, for example quantificational (numeric) groups, for which a controvertial evidence exists on what the relation direction should be. As a result some decisions concerning relation directions vary through systems.

3.2. Principles of syntax evaluation

Since different syntactic parsers employ different formalisms under the hood, it was very important to choose the right representation for the results. A preliminary study on the existing systems showed that most of them use dependency grammar representation. Depndencies were chosen as an output format. Participants who used representations other than dependency trees, were asked to convert their results. There was no unified treebank, every team had to start from scratch. Therefore, it was impossible to use a unified tag set, so we decided not to evaluate prediction of syntactic relations types. Only correct head detection for a node was evaluated.

Another important decision was to ignore some types of differences between the gold standard and markup provided by participants. The main assumption of an assessment procedure was that there is no 'correct' answer in some situations. Only divergences motivated neither by theoretical nor practical decisions were counted as mistakes.

The evaluation corpus consisted of untagged texts of various types: fiction, non-fiction, news and texts from social networks. Since systems performed different text preprocessing procedures, the corpus had been tokenized beforehand. A small part of the corpus (600 randomly selected sentences) was used as a gold standard for the assessment. It was manually tagged by two annotators independently.

Since verb-argument structure relations are mainly encoded by grammatical case and prepositions, the role of word order in the recognition of semantic-syntactic relations shrinks dramatically.

The results are presented in table Table 1

Table 2

System Name	P	R	F1
Compreno	0,952	0,983	0,967
ETAP–3	0,933	0,981	0,956
SyntAutom	0,895	0,980	0,935
SemSyn	0,889	0,947	0,917
Dictum	0,863	0,980	0,917
Semantic analyzer group	0,856	0,860	0,858
AotSoft	0,789	0,975	0,872

The best results have been achieved by the systems based on the manual rule-based approach. Both have a thoroughly elaborated ontologies and lexicographic resourses. However, low-time-consuming systems, such as SyntAutom, have also proved to be reliable. One of the systems, Russian Malt (presision 0,912), was based on the machine-learning technology. It used the SynTagRus Treebank as a learning corpus and achieved the third-highest results (the results are not shown in the chart since the system participated outside the competition).

This event has shown that although Russian is a free-word order language with a rich morphology, the quality of syntactic parsing is quite high. The majority of Russian parsers override the difficulties by developing semantic components and integrating statistical approaches into the rule-based systems.

4. Anaphora and coreference resolution

4.1. Anaphoric and coreference relations in Russian

Anaphora and coreference resolution event in 2014 started with the discussion of what types of relations we would like to detect.

Besides various general complications (annotation of appositive NPs like in *Petrov, the director of* …, annotation of abstract notions coreference) etc. Russian has some specific properties. It lacks the definiteness as a grammatical feature. A bare noun phrase without any determiner (demonstrative or a possessive pronoun) is a standard noun phrase (NP) type for non-first referent mention. There could be no clue in an NP for whether it is a newly introduced referent or before mentioned. For this reason, there are three nearly equal possibilities for such a NP interpretation: (a) a NP refers to one of the before mentioned referents (belonging to an existing coreference chain), (b) it introduces a new specific referent or (c) it is a generic NP. The differentiation of these three types is difficult even for human annotators. Another complication is due to free word order in Russian. It is not a rare event when a reflexive pronoun as *svoj* precedes its antecedent (as in *Svoji fotografii Petrov nikomu ne pokasyvajet* – lit.‘[his own]$_1$ photos Petrov$_1$ to nobody shows’.

Thirdly, Russian is a so-cold pro-drop language, there are cases when a zero pronoun is used to refer to a subject of a clause, the omitted overt referent mention could influence the overestimation of distance between an NP and its coreferring NP from previous discourse.

There are also syntactic zeros for non-finite constructions. These are so called PROs, which are controlled by an NP from another clause and whose overt expression is ungrammatical. The relative frequency of the latter in Russian texts influence significantly on the coreference chains properties. A preliminary comparative research of coreference in Russian, Czech and English [20] has shown that the number of chains in Russian differs from those in English and Czech. According to the authors, the possible explanation is that the number of non-finite subordinate clauses such as infinitival or converb constructions have PRO in the subject position. Thus, the difference in coreference chaining could be strongly influenced by the clause structure of a sentence.

4.2. Gold standard annotation and evaluation principles

The campaign was devoted to both coreference chains extraction and anaphora resolution. The main aim was to track pronominal or all the mentioning of one and the same entity through the text. As previous events this event was the first pilot run for Russian. The tasks were limited to the non-event anaphora; no implicit relations between corresponding NPs (such as part-whole, team-member etc.) were involved.

There were three participants in the first track and seven participants with total 17 runs for the anaphora resolution track. We prepared a little manually annotated training corpus consisting of nearly one hundred texts. Since each participating system has its own NLP pipeline, they used no predefined common standards for morphological and syntactic tagging learning set has no prerequisite morphological and syntactic annotation (including NP annotation).

In our annotation scheme, we addressed the identity relation between coreferential NPs. For there is no grammatical encoding for definiteness in Russian, special cases of distinguishing between discourse-new and discourse-old mentions as well as specific and generic reference were discussed in the annotation guidelines. We excluded from annotation procedure split antecedent cases, abstract notions, some classes of generic NPs. We took into consideration apposition and predicative NPs. However, the latter two NP types did not participate in evaluation. We took as markables NPs of maximal size. We mark potential semantic heads. There were participants who detected only heads as referring expressions. Moreover, the NP heads could vary through systems. Thus, in *admiral Pavel Nakhimov* the head could be *Pavel, Nahimov, admiral*. NPs matching evaluation was based on the head matching criterion in evaluation procedure.

The training data was distributed as a set of texts and a file with anaphoric chains information. In the anaphora dataset a chain consists of two elements: a pronoun (3rd person, possessive and reflexive, demonstratives and the relative pronoun kotoryj 'that'). In the coreference dataset a chain consists of all the NPs—mentions of the same entity with a set of attributes in the training corpus and without attributes in the testing set. While mapping system response NPs to Gold standard NPs we use soft criteria for NP boundary matching, that is a potential head matching principle. We used standard measures for anaphora resolution track. These are precision, recall and F-measure. We use MUC-score for coreference resolution.

4.3. Results and discussion

The forum has shown that there are competitive teams that develop high-level (discourse level) NLP components on a considerably high level (some systems manifest nearly 80% precision for anaphora resolution). However, the task of anaphora resolution is complicated for Russian due to free word order and the absence of overt markers of NP referential status. The absence of free semantic resource as WordNet and freely distributed syntactic parsers make the task more difficult for NLP start-ups and new small teams. The anaphora and coreference resolution tracks have shown the impact of high quality lower level linguistic analysis to the quality of discourse analysis tasks.

5. Conclusions

The RU-EVAL 2014 has brought together a number of IT companies and academic groups that work on Russian NLP tasks (pos-tagging, parsing, anaphora and coreference resolution), and made it possible to assess the state-of-the-art in the field (so far, mostly in Russia). The forum has shown that there are competitive teams that develop NLP components on a considerably high level. However, these tasks have some peculiarities and complications due to high inflectional and fusional properties of Russian language, its free word order and the absence of overt definiteness markers for NP. The absence of free semantic resources as WordNet and freely distributed syntactic parsers make the task even more difficult for newly organized NLP small teams. However, the event was the challenge for those teams that conduct the experiments on various machine-learning techniques.

The event has the following practical outcomes:

- the baselines for three NLP tasks were evaluated;
- the guidelines for tagging according to GS principles have been compiled and tested for Russian;
- new anaphora resolution systems for Russian arises at stretch due to the RU-EVAL 2014 campaign;
- the manually tagged standard sets for morphological tagging, parsing and anaphora and coreference resolution arises;
- new resourses for anaphora and coreference annotation (RuCor) are made available through http://gs-ant.compling.net/ and http://ant1.maimbava.net/ (the latter is to be moved to the former URL);
- RPTB - the Russian Treebank with parallel annotation of four systems (1 million tokens) is available at http://otipl.philol.msu.ru/~soiza/testsynt/
- a new Treebank for Russian (RTB) with manually annotated Subject, Object, Attributive modifier relations has come into being (http://rtb.maimbava.net/res01/rtb.php);
- the created corpora includes the wide variety of genres and various types of coreference relations.

The organizers hope that these corpora would be helpful for other NLP teams for the experiments on coreference resolution algorithms.

6. Acknowledgments

We acknowledge support from the Russian Foundation for Basic research (grant No. 15-07-09306). We thank the Lomonosov Moscow University students RU-EVAL team, and Dmitrij Gorshkov for software support, the RuCor and Treebanks interfaces creation.

7. References

[1] URL: http://www.clef-campaign.org/

[2] URL:http://research.ics.aalto.fi/events/morphochallenge/

[3] Gilles Adda et al. “The GRACE French Part-of-Speech Tagging Evaluation Task”. LREC, 1998.

[4] URL: http://www.senseval.org/

[5] Jens Nilsson, Joakim Nivre. “MaltEval: An Evaluation and Visualization Tool for Dependency Parsing”. Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC’08), Marrakech, Morocco, May. 2008.

[6] URL: http://ifarm.nl/signll/conll/

[7] URL: http://www.nist.gov/speech/tests/ace/

[8] Hirschmann L. “MUC-7 coreference task definition. Version 3.0”. Proceedings of the 7th Message Understanding Conference, 1997.

[9] Orasan C., Cristea D., Mitkov R., and Branco A. “Anaphora resolution excersice: an overview”. Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC’08), Marrakech, Morocco, May. 2008.

[10] Magnini, Bernardo, et al. “Evaluation of Natural Language Tools for Italian: EVALITA 2007”. Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC’08), Marrakech, Morocco, May. 2008.

[11] Lyashevskaja Olga, Astaf'eva Irina, Bonch-Osmolovskaja, Anastasia, Gareyshina Anastasia, Grishina Julia, D'jachkov Vadim, Ionov Maxim, Koroleva Anna, Kudrinskij Maxim, Lityagina Anna, Luchina Elena, Sidorova Evgenia, Toldova Svetlana, Savchuk Svetlana., Koval' Sergej. “Evaluation of the automated text analysis: POS-tagging for Russian”. (Ocenka metodov avtomaticheskogo analiza teksta: morfologicheskije parsery russkogo jazyka). Proceedings of the International Conference on Computational Linguistics Dialogue-2010. [Komp’iuternaia Lingvistika i Intellektual’nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii “Dialog 2010”], pp. 318-327.

[12] Gareyshina Anastasia, Ionov Maxim, Lyashevskaya, Olga, Privoznov Dmitry, Sokolova Elena, Toldova Svetlana. (2012). RU-EVAL-2012: Evaluating Dependency Parsers for Russian. Proceedings of COLING 2012: Posters. pp. 349-360. URL: http://www.aclweb.org/anthology/C12-2035.

[13] Toldova S., A. Roytberg, A. A. Ladygina, M. D. Vasilyeva, I. L. Azerkovich, M. Kurzukov, G. Sim, D. Gorshkov, A. Ivanova, A. Nedoluzhko, Y. Grishina. "RU-EVAL-2014: Evaluating anaphora and coreference resolution for Russian". Proceedings of the International Conference on Computational Linguistics Dialogue-2014. [Komp’iuternaia Lingvistika i Intellektual’nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii “Dialog 2014”], pp. 681695.

[14] Bonch-Osmolovskaya A., Lyashevskaya O., Toldova S. “Learning Computational Linguistics through NLP Evaluation Events: the experience of Russian evaluation initiative”. ACL 2013, Fourth Workshop on Teaching Natural Language Processing. Sofia, 2013, pp. 61-65.

[15] Dalal A. et al. “Building feature rich pos tagger for morphologically rich languages: Experience in Hindi”. ICON, 2007.

[16] Jongejan B., Dalianis H. Automatic training of lemmatization rules that handle morphological changes in pre-, in- and suffixes alike. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1* (ACL '09), Vol. 1. Association for Computational Linguistics, Stroudsburg, PA, USA, 145-153.

[17] Sharoff S., Nivre J. The proper place of men and machines in language technology: Processing Russian without any linguistic knowledge. Computational linguistic and intellectual technologies, 2011, pp. 591-604.

[18] Sokirko A.V., Toldova S.Ju. “Sravnenie effektivnosti dvukh metodik sniatiia lexicheskoi i morfologicheskoi neodnoznachnosti dlia russkogo jazyka [The comparison of two methods for the morphological umbiguity resolution in Russian language]”. Internet-matematika. Moscow, 2005.

[19] Marszałek-Kowalewska K., Zaretskaya A., Souček M. Stanford Typed Dependencies: Slavic Languages Application. Advances in Natural Language Processing. 9th International Conference on NLP, PolTAL 2014, Warsaw, Poland, September 17-19, 2014. Proceedings. Springer International Publishing. 2014, pp 151-163.

[20] Nedoluzhko A., Toldova S., Novak M. Coreference Chains in Czech, English and Russian: Preliminary Findings. Proceedings of the International Conference on Computational Linguistics Dialogue-2014. [Komp’iuternaia Lingvistika i Intellektual’nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii “Dialog 2015”].

SESSION

MACHINE TRANSLATION, NATURAL LANGUAGE PROCESSING AND RELATED METHODS

Chair(s)

TBA

Information Extraction of ICARTT Data Using Natural Language Processing

Elliot Rieflin[1], Dali Wang[1], John Kusterer[2], and Gao Chen[2]
[1]Christopher Newport University, Newport News, VA, United States
[2]Science Directorate, NASA Langley, Hampton, VA, United States

Abstract - *This paper presents an approach for autonomously identifying instruments within airborne measurement data collected by NASA. The data files use ICARTT (International Consortium for Atmospheric Research on Transport and Transformation) format. Instrument information is a required part of ICARTT data files. However, the text containing the information is frequently garbled and incomplete; it often fails to form complete sentences or provide necessary information. In addition the instruments often occur in multiple locations within the same ICARTT file, causing inconsistencies and further complicating the problem. Existing techniques fails to deliver a satisfactory solution due to the complexity of the problem. The proposed solution utilizes a combination of natural language processing and record matching. The algorithm is applied to a sample of NASA data collections from three airborne campaigns over the last 10 years, and has generated satisfactory results.*

Keywords: natural language processing, record matching, object identification

1 Introduction

NASA airborne tropospheric chemistry studies have generated a great wealth of data which need to be held by ASDC (Atmospheric Science Data Center) dating back to 1980s. The data sets have been increasingly used in model assessments in terms of evaluation of physical and chemical processes as well as vertical distributions. NASA's goal is to make aircraft data readily available for scientific studies on climate and air quality issues.

The International Consortium for Atmospheric Research on Transport and Transformation (ICARTT) developed the ICARTT standard [1] in an effort to create consistency between the various experiments and missions performed by members of the group. The format becomes the de-facto standard for NASA airborne data and has been widely adopted by atmospheric scientific research community in archiving their data collections.

In order to ingest the raw measurement data files in ICARTT format into ASDC data archive, properties within each ICARTT file needs to be extracted; each property need to be mapped to a well-defined attribute in order to make the data accessible through a common interface across different files and data collections. The goal of this research is to automate the process.

One of most challenging part of the task is to extract the type of instrument used for each measurement and classify it into one of the predefined list of measurement devices. The instrument information is embedded in the ICARTT files, appears in one or multiple locations, and takes many different forms. All of these contribute to the complexity of the problem.

In its most common form, the instrument information is given in one of more sentences in a narrative or descriptive form. Our problem bears some similarity with the research work that has been classified as record linkage, data cleaning, and object identification in the literature. On record matching, textual similarity comparison is a common approach to decide on matched and non-matched records [2, 3, 4, 5]. In [3], a fuzzy match algorithm assisted with a similarity function has been used to match records such as pre-recorded addresses and product names. The similarity function involves weights for each part of matching target. Given the number of instrument types to be matched, and the variation of forms each instrument may appear in an ICARTT file, it becomes impractical to implement a similar approach.

Graphical models based approaches are another type of well-studied method [6, 7]. An example of such work is by Pasula et al [6]. Their approach uses directed graphical models as a representation of the matching problem. It involves parsing of the references into fields, which is quite complex. In particular, it is a generative rather than discriminative approach, requiring modeling of all dependencies among all variables, making the learning task very difficult.

Statistical and probabilistic approaches were pioneered by Fellegi and Sunter [8] and refined by many researchers [9, 3]. The methods usually involve elaborative statistical models, and ad-hoc cutoff thresholds on matches or non-matches. Their methods typically apply to well-defined records in a relational database, which do not exist in our case.

The goal of this research is to extract the instrument information from the raw ICARTT document through an automated process. After this introduction section, the background and objectives of the work are given in section 2. An in-depth description of our research methods is provided in section 3. Results using real NASA ICARTT files are presented in section 4, followed by the conclusion in section 5.

2. Problem Overview and Background

2.1 ICARTT data format

The ICARTT format has two defined parts: the first is a collection of metadata describing the properties of the measurement, and the second is the measurement data itself. The metadata section is source of information for this study. It contains the information about the data collection (date, organization) as well as information on the format of the stored data (a list of variables, and units).

Unfortunately adherence to the format was adopted inconsistently, causing both missing and redundant data in ICARTT files. An example of one such inconsistency can be seen in the data source description line. This line is intended to hold information such as the instrument name/platform name and model name. However the contents of this line vary greatly, creating difficulties during information extraction.

Close to the end of the metadata there is a section for normal comments. The normal comments section requires information to be paired with a set of tags defined in the ICARTT standard [1], as shown in Figure 1.

PI_CONTACT_INFO: Phone number, mailing/email address
PLATFORM: Platform or site information.
LOCATION: including lat/lon/elev if applicable.
ASSOCIATED_DATA: File names with ship data, etc.
INSTRUMENT_INFO: Instrument description,
DATA_INFO: Units and other information regarding data
UNCERTAINTY: Uncertainty information,
ULOD_FLAG: -7777 (Upper LOD flag, always -7's).
ULOD_VALUE: Upper LOD value (or function)
LLOD_FLAG: -8888 (Lower LOD flag, always -8's).
LLOD_VALUE: Lower LOD value (or function)
DM_CONTACT_INFO: Data Manager – contact info
PROJECT_INFO: Study start & stop dates, web links, etc.
STIPULATIONS_ON_USE: (self-explanatory).
OTHER_COMMENTS: Any other relevant information.
REVISION: R# See file names discussion.
Figure 1. Comments section of ICARTT standard

These lines often help mitigate the inconsistencies throughout the rest of the metadata. For example the INSTRUMENT_INFO flag seen in figure 1 can be leveraged in order to extract instrument information without relying on the much more variant data source description line.

2.2 Objectives

The inconsistent adherence to the ICARTT format has rendered the data contained near impossible to analyze. Furthermore, new ICARTT files are constantly being submitted containing new errors and styles. In order to solve this problem the Toolsets for Airborne Data (TAD) group was formed within NASA. TAD's goal was to consolidate the information contained in the ICARTT format and present it in a manner conducive to further study.

TAD decided to approach the problem in a two-part solution shown in Figure 2. The first part (the Data Ingest System) autonomously converts ICARTT data into an extended form before storing in a database. The second part focuses on the presentation of the data in a user friendly manner [11].

There are two processes within the first part as shown in Figure 2: an information extraction and label classification process which is the focus of this work, and the file reformatting process which parses the files into a well-defined format.

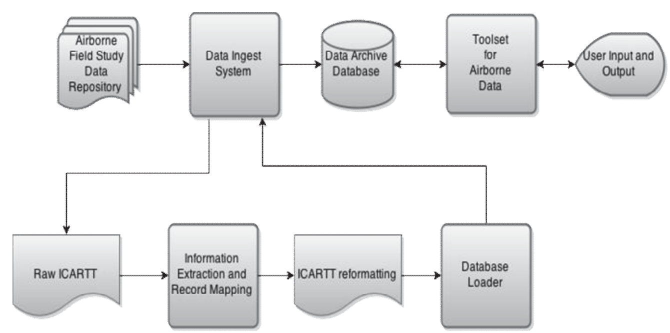


Figure 2: TAD Data Flow

The goal of this research is to identify the instruments that are used for the data measurements from an ICARTT file's metadata section, and to uniquely map each variable in the file to a known instrument from a predefined list of about 100 instruments. The Instrument is one of the most important attribute of each measurement. Identification of instruments in ICARTT files will allow different measurement data files across different field campaigns to be associated through the common instruments they used. As a result, an end-user will able to search, merge, or subset data based on instruments used through a web interface shown as the second part of Figure 2.

3. Proposed solution

In order to automatically determine the instrument, two lines from the file are processed using a series of natural language processing tactics. These two lines are the data source description line and the instrument information flagline. Because the data source description line was frequently used to contain other information during early iterations of the ICARTT format, the instrument flagline is searched for matches first. The approach utilizes the following natural language processing techniques.

3.1 POS tagging

The first step in instrument detection involves determining which grammatical part of speech each word in the text holds. Part of speech (POS) tagging is a well explored field within natural language processing [12].

Tagging parts of speech is accomplished via training on segments of text. Part of speech tagging also involves viewing the context, considering the preceding and following word as well as larger groups of words in n-grams (where n is the size of the word grouping). In practice this involves a cycle of POS predictions on a large corpus of text, followed by corrections based on correct POS roles previously identified manually.

The POS tagger leveraged to solve this problem is a pickled (previously trained) Maxent Treebank POS (MTP) tagger trained for the English language on the University of Pennsylvania Treebank project. A MTP tagger focuses on maximum entropy, that is it minimizes the amount of information stored by the tagging model, and tags words based off of a tag set also used in the Penn Treebank project [13].

Adjective: JJ, JJR, JJS
Noun: NN, NNS, NNP, NNPS
Verb: VB, VBD, VBG, VBP, VBZ
Preposition: IN
Determiner: DT
Coordinating Conjecture: CC
Cardinal Number: CD
Adverb: RB, RBR, RBS
Misc: TO, SYM,

Figure 3: Common Penn Treebank

Using a real example from our application, the POS tagger takes sentences such as “CNgt3nm is measured with a TSI CPC-3025” and outputs the following [('CNgt3nm', 'NNP'), ('is', 'VBZ'), ('measured', 'VBN'), ('with', 'IN'), ('a', 'DT'), ('TSI', 'NNP'), ('CPC-3025', 'NNP')], binding each word in the sentence to a part of speech descriptor.

This tagging methodology is selected because it showed great accuracy during preliminary testing on ICARTT files, despite the possibility of custom training a POS tagging algorithm on the ICARTT data format. A custom trained POS tagger could result in more accurate predictions; however, such training was outside the scope of this problem due to the accuracy observed from the MTP tagger used.

3.2 Chunking

Chunking is, in many ways, the most important step in this process. By grouping grammatical phrases based on the sequence of associated position tags determined during POS tagging, the text can be considered as a series of phrases. This allows for matching entire instrument phrases, instead of matching word by word. Without this step, instruments such as Aircraft Instrument (AI) would be detected in almost every file, as ICARTT data frequently is measured by many types of instruments on aircrafts.

NP: {<DT|JJ|CC|CD|NN.*>+}
VP: {<VB.*|TO>+}
PP: {<IN><NP>}
IP: {<NP><VP><PP|NP>}

Figure 4: Chunking Rules and Instrument Phrase

In order to chunk the info lines, a set of grouping rules must be defined. These rules are similar to regular expressions in nature, but involve part of speech tags instead of standard regular expression arguments.

The chunking stage is where this approach differs from the standard, in addition to chunking based off of common English phrases, such as noun phrases, more specific chunks must be developed. For the purpose of ICARTT files, rules for an “Instrument Phrase” (IP) are created in this work. This instrument phrase contains verbs which suggest measurement or data collection, in addition to two noun phrases or a noun phrase and a prepositional phrase. The format can be expressed in a condense form seen in figure 4 as IP, where NP is a noun phrase, VP is a verb phrase, and PP is a prepositional phrase consisting of a preposition and a noun phrase.

Applying chunking rules to the previous, POS tagging, example gives the following result. The sentence “CNgt3nm is measured with a TSI CPC-3025” is first chunked as such, “CNgt3nm” is the first noun phrase, followed by the verb phrase “is measured”, the preposition with and another noun phrase “a TSI CPC-3025”. Then the sentence is identified as an Instrument Phrase by chunking together the preposition with the second noun phrase.

This chunking process addresses edge cases where the writer mentions multiple instruments within the same file. Occasionally the text will mention a list of instruments located on the aircraft. Often only one of these instruments is actually used to measure variables in that file.

Because of the need to distinguish between results, instrument phrases where the verb indicated some form of measuring or collecting are identified. This identification is later used when considering the strength of an instrument match.

3.3 Stopwords

Stopwords are specific words which are identified as common occurrences. These words occur so often that they can be ignored in certain contexts. An example of stopword detection can be observed in one of the files where a Chemical Ionization Mass Spectrometer (CIMS) was used to measure variables. In this case the phrase “a CIMS” matches the CIMS abbreviation in the instrument dictionary. This match is a perfect match, as 'a' is always a stop word and therefore does not affect the distance between the two phrases.

3.4 Lemmatization

Lemmatizing is the process of reducing words to their base forms. After chunking and POS tagging, word lemmatization can assist in the matching process. By finding the root of the word, the variance of possible matches is reduced. The words “measure”, “measured” and “measuring” would all match as the word “measure” after lemmatization. Lemmatization is accomplished by utilizing the WordNet lexical database [14] developed by Princeton University.

This process is important as reducing the possible variance of the two strings greatly improves the accuracy of any attempt to match them. For example, this increase in accuracy is especially visible when attempting to record match sentences to the Laser Induced Florescence (LIF) instrument. By lemmatizing the text and the instrument name, strings such as “the data was collected by inducing florescence with a laser” can match the LIF variable. This would not have matched in its original state, as “Laser Induced” would have been modifying “florescence” and “induced” would not be considered a verb.

3.5 Semantic role labeling

Determining which noun phrase in the instrument phrase contains the instrument requires semantic role labeling.

If the roles in the phrase are not identified, it would be possible for one instrument to measure another, and the identification would be incorrect.

This process historically involves training on a large corpus before attempting to label roles [15]. This process is fairly intensive. In addition, the ICARTT collections were produced by different scientist from different organizations at different time. These make a statistical approach less attractive unless a large corpus with consistence can be found.

If the prepositional phrase detection between the POS tagger and the chunker is accurate, the resulting prepositional phrase can be used to determine information about the attached noun phrase. This process is accomplished via utilizing the types of the neighboring chunks determined by the chunker, as well as considering the preposition itself. A survey of the problem reveals that the instrument will either occur in a solo noun phrase chunk, or as a part of a larger instrument phrase as described above.

The execution of this role labeling can be best seen through the following example using the Particle Soot Absorption Photometers (PSAP) instrument. First the phrase “PSAP were used to measure soot” is in the form <NP> <VP> <NP> and thus the first Noun phrase in this sentence is the instrument.

A Second, similar phrase “soot was measured by PSAP” is in the form <NP><VP><PP> so the prepositional phrase must be considered in order to determine which noun phrase is the instrument and which is being measured.

At this point consideration of the preposition is required. If the statement was “PSAP measured in MM-1” then the nature of the preposition “in” suggests that the instrument is the first noun phrase. This is revealed by the fact that a noun cannot act “in” an instrument. However, in this example “the soot was measured by PSAP”, the preposition “by” suggests that the second noun phrase is the instrument.

Therefore, instead of training for an entire file, the role labeling code need only to identify relevant noun-phrase relations for the much smaller list of possible prepositions.

A set of early ICARTT files were selected to test the classification of roles, as these files had the most variance and more likely contained abnormal cases that would otherwise be missed. For these files, a list of symbolic rules for each preposition achieved the greatest prediction accuracy. The rules were edited in order to fix oversights in the initial ruleset and then applied to related instruments for a much larger selection of ICARTT files.

3.6 Matching Instruments

The process of matching instruments is not simple. The instrument is broken into two parts, abbreviated name and full name. In addition to matching both of these parts, the ICARTT text frequently contains typographical errors and missing data. Because of the nature of this problem a multi-part matching algorithm was implemented.

Both the abbreviated name and the full name were compared against the text in the same manner. This comparison searches for each word in the abbreviated name within each identified phrase. If a similar word is found, the Levenshtein distance formula is applied to determine the similarity. Once the entire text is considered, the closest match for each word is retained.

The best match for each word in the abbreviated name is then used to calculate the result in in following formula.

$$Match\ Confidence = \frac{1}{1 + (\frac{\sum \text{Distance}}{length})}$$

Where distance is the closest distance match for an individual word and length is the number of words in the instrument name. This equation results in a confidence of 1 if the levenshtein distance of all words is 0, and a confidence which approaches 0 as the distance grows across all words.

Finally, the confidence values calculated from both the abbreviated name and the full name are compared. And the highest confidence is taken as the match confidence for the instrument being considered.

3.7 Comparing matches

Once a match is made after the POS tagging, chunking, and role labeling process, it must be compared to other matches that have occurred. This is a difficult process, as it is possible for more than one match to occur in the file at full confidence. In order to improve prediction accuracy a significance coefficient was determined for each match. The significance coefficient is a value which identifies the strength of a match based on its relative position in the text. This significance coefficient, multiplied by the previously calculated confidence, became the new confidence of a match reported between a specific phrase in the text and an instrument. The algorithms final detection is the instrument with the highest match confidence overall.

4. Results

In order to verify the accuracy of this method, a sample set of files from six NASA airborne missions were run, containing a total of 1092 variables. These missions contained data files collected by multiple researchers over a time period from year 2006 to the present.

The output was manually checked against instrument mappings generated by human experts. This information was evaluated on both per mission and per instrument basis.

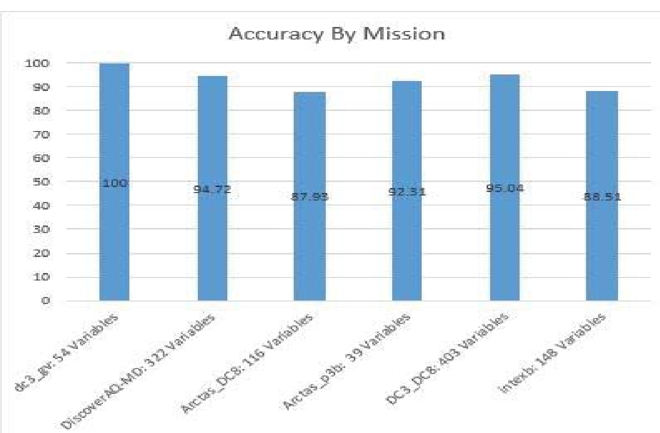


Figure 5: Detection Accuracy

As shown in figure 5 there was slight variance in the success of the approach related to the associated mission. In particular the Arctas DC8 and Intex-B files had an abnormally low success rate. During evaluation of these two missions, along with the rest of the missions, a set of common errors were identified.

The analysis also revealed situations in which the approach was accurate when supplied particularly dirty input. The software was able to correctly predict the instrument in use even when multiple unrelated instruments were mentioned across multiple paragraphs, but not used to measure any variables in the file. Furthermore several files had conflicting information within the INSTRUMENT_INFO line and the data source description line, these files were properly handled by the record linking process and an accurate prediction was possible.

These common errors in the described approach can be described in the following three categories, derived variables, thermal dissociation variants of other instruments, and multiple instruments. These three issues accounted for 54 of the 71 incorrect identifications.

4.1 Multiple Instruments

As mentioned before, matching instruments in files mentioning unrelated instruments causes a conflict with files containing multiple instruments. The singular nature of the INSTRUMENT_INFO line further complicates this issue, as there are no clear divides by which to identify multiple variables within one file. To show the difficulty of this problem a few examples are provided.

The first example shows measurements utilizing different varieties of the Chemlum instrument which are considered to be separate instruments.

Example 1: Multiple types of Chemlum

variable name	Instrument guess	correct instrument
NO_pptv_Weinheimer	Chemlum-O3	Chemlum-O3
NOy_pptv_Weinheimer	Chemlum-O3	Chemlum-O3
NO2_pptv_Weinheimer	Chemlum-O3	Chemlum-O3
O3_ppbv_Weinheimer	Chemlum-O3	Chemlum-NO

Figure 6: Multiple Instruments Example

INSTRUMENT_INFO: “DC8 in situ NO, NO2, NOy, O3”.

While the instrument information does contain information suggesting that both forms of the Chemlum instrument were used, the sentence is very unclear. Similar uncertain predictions occur in cases when only a single instrument is measured, such as the example 2 shows.

Example 2: APS as single instrument

In this second example, the instrument for all variables is the Aerodynamic Particle Sizer. The relevant INSTRUMENT_INFO line is given below.

INSTRUMENT_INFO: “TSI APS Model 3321
-comparisons with other sizing instruments such as SMPS often requires a
-conversion from aerodynamic to geometric diameter.”

Since SMPS is also an instrument, the software must determine whether APS or SMPS is used to measure each variable.

Because of the similarity of the INSTRUMENT_INFO lines in these two examples, attempts to detect multiple instruments resulted in false matches for currently accurate variables. This resulted in a net lowering the program's accuracy across the files.

4.2 Derived variables

Derived variables presented a similar problem, as the text below shows.

“Scattering data are measured with a TSI Nephelometer model 3563 and were corrected for truncation errors using Anderson and Ogren AS&T, 29, 57-69, 1998”

When analyzing this text the Nephelometer instrument is detected at perfect confidence. However the data file contained mostly derived variables, with only one variable listed as measured by a Nephelometer.

Because of the conflicts encountered when approaching these two problems, derived variables and multiple instruments where considered within acceptable error for the scope of this problem.

4.3 Results organized by instrument

Out of the 40 instruments found in the six missions, a total of 27 where matched with perfect accuracy. Only 9 instruments were matched less than 75% of the time. 8 of these 9 instruments only appeared in files where either no instrument was mentioned in any descriptive line, or multiple instruments where listed and only one of them matched.

The last instrument, Thermal Dissociation Ionization Mass Spectrometer (TD-CIMS), always matched as CIMS and forms the entirety of the missing prefix group of errors. Presumably, this error occurs because the thermal dissociation phrase was frequently separated from the instrument name.

4.4 Final results

Overall the testing showed a 93.4% accuracy across all missions and instruments, with four categories of errors, as shown in figure 7.

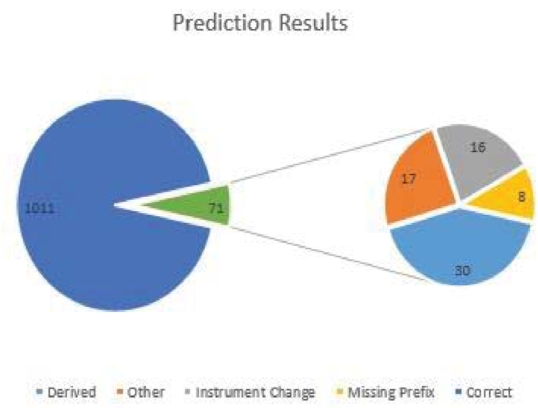


Figure 7: Breakdown of Results

5. Conclusion

The task of instrument type extraction was previously performed manually by domain experts. Due to the sheer volume of the data archive, it becomes imperative for NASA to automate the process. This research provides a viable solution by utilizing natural language processing combined with other object identification techniques. Our experiment shows that the algorithm is both efficient and accurate, taking only a small handful of minutes to run through multiple gigabytes of ICARTT files.

The accuracy of this approach is around 93%, exceeding the aim initially set for the project. This research is under an ongoing umbrella project: Toolsets for Airborne Data (TAD). Future research has the potential to bring us the secondary information we need (i.e., the existence of multiple instruments and derived variables) in order to solve remaining misidentified cases.

Acknowledgments

This work is supported by NASA Langley’s Research Center, especially its Science Directorate and ASDC, under a NASA grant. We would like to thank Lindsay Parker, Amanda Benson and Aubrey Beach for their guidance and support. We would like to acknowledge the contribution of CNU graduates Nathan Typanski and Matthew Rutherford for helping to create a library which provided a method by which to extract specific lines in an ICARTT file, such as the instrument information line.

References

1. ICARTT File Format Standards V1.1, International Consortium for Atmospheric Research on Transport and Transformation, 2013
2. WilliamW. Cohen, Pradeep Ravikumar, and Stephen E. Fienberg, A Comparison of String Metrics for Matching Names and Records, Proceedings of 2003 SIGKDD, page 73-78, August 2003.

3. Surajit Chaudhuri, Kris Ganjam, Venkatesh Ganti and Rajeev Motwani, Robust and Efficient Fuzzy Match for Online Data Cleaning, Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 313 – 324, June 2003, San Diego, CA.
4. Aron Culotta, Andrew McCallum, Joint deduplication of multiple record types in relational data, Proceedings of the 14th ACM international conference on Information and knowledge management , Pages 257-258
5. C. V. Trappey, Fu-Chiang Hsu, D.W. Hsiao , A Fuzzy Ontological Knowledge Document Clustering Methodology, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Volume:39 , Issue: 3, Page:806 – 814, 2009
6. H. Pasula, B. Marthi, B. Milch, S. Russell, and I. Shpitser, Identity uncertainty and citation matching, Advances in Neural Information Processing, 15, 2003.
7. Parag and Pedro Domingos, Multi-relational record linkage, Proceedings of the KDD-2004 Workshop on Multi-Relational Data Mining, pages 31–48, August 2004
8. I. P. Fellegi and A. B. Sunter, "A Theory for Record Linkage," Journal of the American Statistical Association, Vol. 64, pp. 1183-1210, 1969
9. Winkler, W. E., "Frequency-based Matching in the Fellegi-Sunter Model of Record Linkage,"
Proceedings of the Section on Survey Research Methods, 1989, American Statistical Association, 778-783.
11. Matthew Rutherford, Nathan Typanski, Dali Wang and Gao Chen, "Processing of ICARTT data files using fuzzy matching and parser combinators", 2014 International Conference on Artificial Intelligence, pp. 217-220, Las Vegas, July 2014
12. Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer, Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network, Proceedings of HLT-NAACL 2003, pp. 252-259.
13. Adam L. Berger, Vincent J. Della Pietra, and Stephen A. Della Pietra, A maximum entropy approach to natural language processing, Computational linguistics, 22(1):39–71, 1996.
14. George A. Miller , WordNet: A Lexical Database for English, Communications of the ACM, Vol. 38, No. 11, pp. 39-41, November 1995
15. D. Gildea and D. Jurafsky, Automatic labeling of semantic roles, Computational linguistics, *28*(3), 245-288.

Evaluation of a Standalone Language-independent Dialogue Framework

Arnaud Jordan[1] **and Kenji Araki**[1]
[1]Graduate School of Information, Science and Technology, Hokkaido University, Sapporo, Japan

Abstract—*This paper presents an innovative dialogue agent designed for textual casual chatting, which can handle any language. The system acquires knowledge from a non-annotated corpus and then represents all the language aspects into a graph. Using previously acquired knowledge it splits sentences into sub-nodes to proceed to a flexible output generation. Moreover, it uses graph clustering to generate node categories without using any grammar-related tags, and uses these categories to induce new knowledge. The system uses the same processing regardless of the language, that makes the system able to handle any language without any adaptation task. In addition, since the system uses only a limited number of resources, it can be set up as a standalone system in order to preserve the user privacy. We carried out dialogue correctness experiments in Chinese, English and Japanese and obtained results comparable to a more language-specific multilingual system.*

Keywords: natural language processing, multilingual system, spoken dialogue agent, real-time, graph clustering

1. Introduction

Nowadays, a large number of spoken dialogue agents have been proposed, such as ALICE [1], or are still under development. Some of them focus on non-task-oriented dialogues, while others focus on providing information or achieving a particular task. In this paper, we focus on non-task-oriented dialogues because we consider it as the first step to build a complete system which at the end may be able to handle both task and non-task oriented dialogues at the same time.

Many non-task-oriented dialogues systems have already been proposed. With progress in research and systems improve, spoken dialogue agents are able to handle more and more situations, like the Multimodal Multi-domain Spoken Dialogue System [2], for example. However, to reach this objective they use many high-level operations such as word categorization or case grammar. Consequently, to handle these complicated processes most of systems require very language-specific resources such as dictionaries or grammatically tagged corpora. For example, many systems work only in a specific language, such as Japanese [3]. These kinds of systems cannot be easily adapted to another language without a lot of work. A solution would be to create a multi-lingual model that handles all languages [4]. However, this is a hard task since each language has specific aspects that are not used in other languages. Nevertheless, it is possible to try to implement the most common behaviors to cover a maximum of languages. However, the result will be incomplete and not optimal for each specific language. That is why we opt for implementing only very basic processes used universally in all languages.

In this paper, we propose a framework that has been developed with the aim of handling any language, and which consequently uses no language-specific resources to keep a maximal generality. For example, the system must be able to handle a newly constructed language using only some samples of this language. In addition, the proposed framework includes no copyright covered elements and as a result can be easily implemented and adapted in various environments. Moreover, it can be considered as a base framework for a system focusing any specific language.

However, in order to check our algorithm before adding new processing we focus on very simple dialogues. We will improve the system in future, for example, we will increase the speed of the output generation to be able to handle more knowledge.

Developing an algorithm that is not dependent on language is a complicated task, and the results may not be better than the current best language-specific systems. However, it would be useful to achieve many different objectives such as those listed below.

- Handling and acquiring the meaning of new terms, such as words used by young people.
- Minority language support, languages for which specific natural language tools are not available.
- Foreign language learning using casual dialogue as training.

Moreover, since the system uses no external tools, it can be easily distributed or installed on a mobile device and works without any network connection as a standalone system. Consequently, it can also provide a full privacy protection to the user.

In addition, the system could be set up to handle non-verbal input such as sign language too. Since the system can handle any kind of input such as gestures, specials tags or texts, it can be considered naturally multimodal.

2. Outline

The proposed system uses graph traversal to generate and select the optimum responses to the user's inputs. Consequently, the system is composed of two main parts: graph construction which replaces the use of external resources, and graph parsingm, which is used to generate the system's responses.

In order to handle more complex dialogues, we will need to improve the first phase to, for example, automatically acquire knowledge in a similar way as is performed in a lexical database such as WordNet [5].

2.1 Graph representation

All the acquired knowledge is represented in a graph using nodes and directed links, as shown in Figure 1. The system generates many different links, however to keep the figure easily readable we only represent some of them.

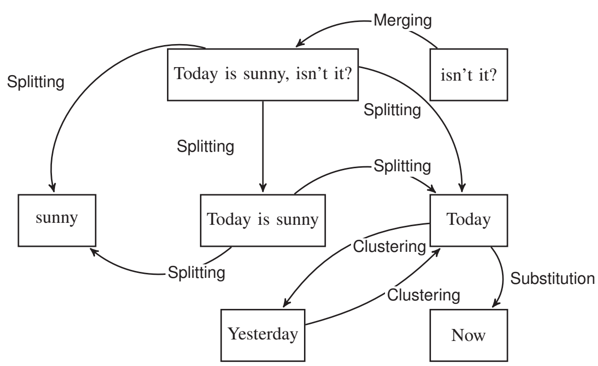


Fig. 1: A simple graph

2.2 Basic links

We used only basic links, which are necessary for output generation [6], in order to preserve the generality of the system.

2.2.1 Splitting

A *splitting link* is generated between a node and its sub-node. Here, sub-node refers to a node whose value is contained in another node value. For example, the node "I like pears" contains the node "pears" and is consequently related to it.

2.2.2 Merging

A *merging link* is the opposite of a splitting link. It is set between a node and a super-node. For example, there is a merging link between "How" and "How are you?".

Concretely, a *merging link* is set between a sentence's part sub-node and all the complete sentences which include this sub-node. Using these links, the system can retrieve complete sentences that are eligible for output.

2.2.3 Substitution

A *substitution link* is provided between a *node A* and a *node B*, if the *node B* can be used instead of the *node A* in the output. This substitution can be considered as a similar process to association in psychology [7].

For example, when the user inputs "Hello", the system can answer "Hello", "How are you?" or "How do you feel?". Consequently, they are *substitution links* from "Hello" to "How are you?" and to the other possible responses.

However, if the input is "How are you?" the system may reply "I am fine" and "I am tired" at the same time, which would not be coherent behavior. To avoid this kind of unexpected action it is possible in a future version of the system to implement emotional concepts; the node "I am fine" can be connected to a good emotion, i.e. a node representing this emotion, and the node "I am tired" to a bad one, and then the system can be set up to output only nodes related to the same emotion when the user inputs a question. These emotional nodes are not related to language, since the same basic emotions are used by all humans [8].

2.2.4 Clustering

The system uses the MaxMax algorithm [9] to create nodes clusters and generate *cluster links* in order to be able to generate more various responses to the user's input. The MaxMax algorithm has been made to suit tasks such as Word Sense Induction (WSI). It is a non-parametrized and graph applicable algorithm which is very easy to implement. However, other clustering algorithms that work on graphs can be easily adapted to be used in the system.

Concretely, for example, "apples" and "oranges" can be related by a *cluster link*. The system attempts to replace the nodes of the sentences with others from the same cluster to generate a new sentence. If "apples" and "oranges" are in the same cluster and if the system learns the sentence "I eat apples", then it will generate the sentence "I eat oranges".

2.3 Node generation

The system uses training samples (cf. 2.6) to generate nodes in the graph before the dialogue starts. Firstly, each sentence of the samples is converted into a node called an *input node*. For example, the sentences "Hello" and "How are you?" are converted into two distinct nodes. Then, the system proceeds to generate the sub-node.

2.3.1 Sub-node generation

In natural language processing, one of the most common tasks is to identify words present in a sentence. However, in the context of a multilingual system we cannot use a morphological analysis tools such as JUMAN [10] which are only available in specific languages, such as Japanese.

A solution would be to use unsupervised word segmentation [11]. However, we need a real-time and fast adaptive algorithm. This is why we develop our own algorithm to identify parts of the sentence. We use already existing nodes to try to split new ones. For example, the system uses the node "I like" to split the node "I like peaches" into "I like" and " peaches". The generated sub-nodes can represent several words, e.g. "like peaches", as well as a single word like "peaches" or a part of a word like "ach".

This method will generate a lot of noise, i.e. many nodes that are not useful for output generation, as well as useful ones. However, as has been proven for stochastic resonance [12], it could also help the system to generate many correct and useful responses. Concretely, the system may access many ineffective nodes which will not be used to generate the output of the system, because they are regularly related to all the other nodes. As a result, their influence on the choice of the output is limited.

2.4 Link characteristics

Each kind of link between nodes has its own characteristics. These are used during the graph traversal to calculate the node's score and the link's cost.

- **Node score** denotes the importance of a node.
- **Link cost** refers to how much power is needed to take the link and go to the pointed node. This value is used to limit the graph traversal.

All links that have a link cost exceeding a defined value (arbitrarily set to 5) are ignored by the system.

Each kind of link has the following three characteristics.

- **Weight** denotes the value of the linked node; links that bring a lot of information such as a substitution link have a high value.
- **Distance** denotes the information difference. Splitting links only remove a part of the information; consequently their distance is small.
- **Base cost** is used to calculate the cost of the link.

Changing these characteristics will change the system's behavior. For example, we can make the system generate more sentences[1], but these will not be all correct or make the system take a more careful behavior[2] and only output sentences that are definitely correct.

Equation (1) is used to calculate the node score, and the weight and distance are calculated by aggregating the total values of the links used to arrive at this node from the user's input.

We use the exponential function, to limit the number of parsed nodes. For example, we want to avoid a path which uses many small distance links.

$$S_n = \frac{\sum weight}{e^{\sum distance}} \tag{1}$$

- S_n is the score of a specific node.

[1]Decrease the *distance value* or the *base cost*
[2]Increase the *base cost*

In addition, we use Equation (2) to calculate the link cost.

We use the number of links to decrease value of very frequent nodes in a similar way to the tf-idf method [13]. This is often the case of nodes resulting from the noise of the splitting algorithm. In addition, we use a logarithm to reduce the difference between two nodes that only have a small difference in number of links, and consequently can be considered similar.

$$C_l = c \times (1 + log(n_{link})) \tag{2}$$

- C_l is the cost of the link.
- c is the base cost of the target link type.
- n_{link} is the number of links of the corresponding type from the same node.

The clustering links are used to create new nodes, but are not used during the graph traversal.

Table 1 contains the empirically defined values for each type of link.

Table 1: Links' characteristic

Link	Cost	Distance	Weight
Splitting	1.5	0.75	2
Merging	0.99	1	3
Substitution	2.5	2	5

2.5 Output generation

As shown in Figure 2, the system checks each input node of the graph to look for all the nodes that match, include or are included in the user's input[3]; all the matching nodes' score is increased.

```
ALGORITHM visitingGraph(input)
    FOR EACH inputNode OF inputNodes
        IF inputNode = input
            inputNode.increaseScore()
            folowLink(inputNode, 0)
        ELSE IF inputNode contains input
            inputNode.increaseScore()
            folowLink(inputNode, 0)
        ELSE IF input contains inputNode
            inputNode.increaseScore()
            folowLink(inputNode, 0)
```

Fig. 2: Algorithm used to find node related to the input

Then, as shown in Figure 3, using the previously-acquired links, the score of all the nodes that are related to a matching input node will be increases too in function of their links' characteristics. All the nodes will be accessed until the link cost exceeds a defined value. The link cost of each link is added to the previous link cost, and as a result the cost used in the comparison increases each time the system follows a link.

[3]For example, the input "I like eating" includes "eating" and is included in "I like chocolate". As a result these two nodes' scores are increased.

```
ALGORITHM folowLink(node, currentCost)
    FOR EACH link ELEMENT OF node.links()
        cost:=currentCost + link.cost()
        IF cost < MAXCOST
            linkedNode:= link.getNode()
            linkedNode.updateScore()
            folowLink(linkedNode, cost)
```

Fig. 3: Algorithm used for graph traversal

Finally, the node that has the best score, which exceeds the trigger value (cf. 2.5.1), is selected and output.

For example, if the input sentence is "I like making cookies", the nodes "I like" and "cookies" are included and their score will be increased. Both of these are related to the node "I like eating cookies" by a merging link, and its score will be increased too. If the score of the node exceeds the trigger value, the system will output "I like eating cookies".

If no node's score exceeds the trigger value after all of the graph has been traversed, the system will output an apology sentence such as "I am sorry, I cannot reply". Those apology sentences are present in the training samples (cf. 2.6).

2.5.1 The output trigger value

In the aim to create a real-time system, the system has to reply in a minimum amount of time like a human would, but with maximum relevance, i.e. the best possible response. To produce this kind of behavior, the system uses a dynamic trigger, which value decrease in function of the time spent, using Equation 3.

$$V_t = V_i - t \times k \tag{3}$$

- V_t is the value of the trigger at t.
- V_i is the initial value.
- t is the time since the initial value.
- k is a defined coefficient.

This equation makes the trigger value decrease continuously using a single parameter that is empirically set. The system periodically checks if a node score exceed the trigger value. Then, it will output the node which has the higher score to the user. After each iteration, the score of the outputted node is set to 0 and the score of all the remaining nodes is decreased.

Figure 4 shows an example of the trigger evolution.

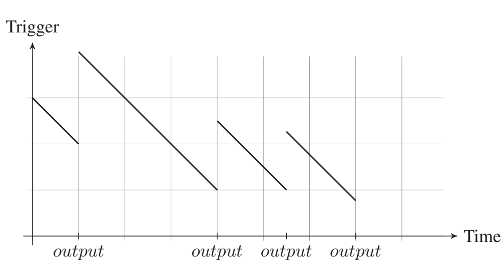


Fig. 4: Example of trigger evolution

After the system selects an output, the trigger value is reinitialized. This new value is calculated using Equation 4.

$$V_i = \left(\sum_{k=i-5}^{i-1} S_k\right)/5 \times 2 \tag{4}$$

- S_k is the score of the output k.
- i is the output number.

Concretely, the system uses the mean of the last five outputs' score corresponding to the output node score to calculate the new trigger value. This method allows the system to adapt to the nodes' scores automatically.

2.5.2 Example of output generation

Using the graph of the Figure 5 the system can generate several responses.

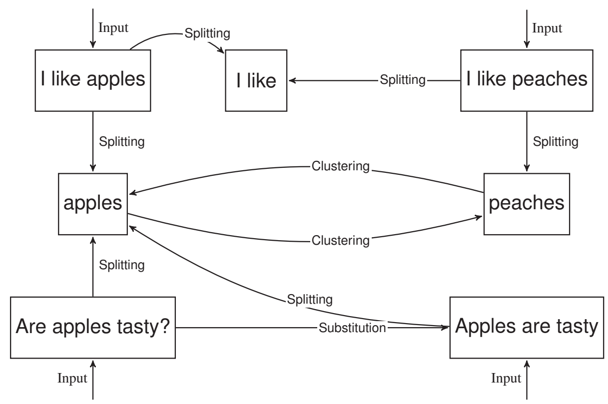


Fig. 5: Example of graph used to generate an output

Some of possible responses are listed below.

- If the input is "Are apples tasty?" the system can directly output "Apples are tasty".
- If the input is "I like apples?" the node "I like apples" will be selected as output, since it is included in the input.
- If the input is "Are peaches tasty?" the system can use the cluster link and output "Peaches are tasty".

The system will output all the nodes which are complete sentence and which score exceed the trigger value, for one input several outputs are possible.

In addition, a dialogue is a real-time process [14], to make the system enable to receive inputs at any time we implement each operation in a different thread executed in parallel. Since, there are no blocking operations, the system can continue to receive inputs while it is generating an output. Moreover, the new input will influence the current output generation.

2.6 Training samples

To generate the graph, the system uses two kinds of basic resources, which contain no grammar information and need no complex creation processes. They can be, for example, extracted from a dialogue between two humans or from any kind of text such as books or screenplays.

Compared to a common system based on AIML [15] corpus, the corpus of the presented system contains no tags and all the rules are automatically acquired from the samples. For example, wild-cards, which are often present in corpus-based chat-bots, are not present in the samples. They have to be statistically induced[4] by the system.

2.6.1 Dialogue samples

The dialogue samples contain some very simple dialogues such as those shown in Figure 6, used to acquire substitution and splitting links in the target language. Concretely, a substitution link is set between an utterance and its response and between all the sub-nodes of the utterance and all the nodes of the response which are not present in the utterance.

U1: what do you drink? U2: I drink milk

Fig. 6: Example of a dialogue sample

2.6.2 Knowledge samples

The knowledge samples are a list of simple sentences such as "I like cookies" or "The president of the USA is Obama".

The knowledge samples are used to increase the possible outputs of the system. These samples can be collected easily, since they consist of a list of simple sentences that are not contextually related. They can be collected from a text such a Wikipedia article or from users' dialogues.

3. Experiments

We used the same protocol as the evaluation of generality of SeGA-ILSD [16]. However, for our system we do not use an ELIZA-type system to generate part of the responses. We also do not use morphological analysis tools as the baseline system does.

In order to fit the baseline experiment process, we used a speech input tool. However, this kind of speech recognition tool uses a lot of language-dependent resources and they are only provided for a limited number of languages. That is why for the experiment we only consider the speech recognition as an input tool which replace the keyboard and which is not a part of the presented system itself.

We use the Google speech recognition implemented in an Android[5] application to get the user's inputs and evaluations.

We asked subjects to evaluate each response of the system as below.

- **Correct reply** Meaning is correct, and expression is natural.
- **Semi-correct reply** Meaning is correct, but expression is not natural.
- **Erroneous reply** Meaning is not correct.

The aim of the evaluation is simply to check whether the system's responses are grammatically correct and correspond to the user input. Nevertheless, we asked the subjects to evaluate the system's response as erroneous if the system does not reply to the input question. For example, if the input is "What will you do tomorrow?" and the response is "I don't know", it is considered to be erroneous even if the output is grammatically correct and a human could reply in a such way.

3.1 Baseline

We used the SeGA-ILSD system as a baseline for this experiment. This spoken dialogue system uses an inductive learning method based on genetic algorithms with sexual selection. Concretely, it acquires rules automatically from pairs consisting of an utterance and its associated reply, and attempts to crossover two rules to create a new one. Rules that generate erroneous output are progressively removed from the system using user feedback.

In order to crossover two rules, the system needs to identify each word in the sentences and in consequence, for which it uses a morphological analysis tool[6].

In addition, when no rules are found to reply to the input, the system uses an ELIZA-type system to generate the output. The ELIZA-type system contains manually created rules that are different for each language.

Moreover, the baseline uses Microsoft Japanese recognizer (Version 6.1), Microsoft English Recognizer (Version 5.1) and Microsoft Simplified Chinese Recognizer (Version 5.1) as speech recognition tools[7].

3.2 Preparation of the experiment

We asked three native speakers each of Chinese, English and Japanese to imagine each one a simple casual dialogue of about 40 sentences in order to create the dialogue samples.

The Japanese knowledge samples were directly extracted from our previous research. For this research we asked subjects to teach some common knowledge to train a spoken dialogue agent. The same samples were also manually translated into the two other languages by a native speaker.

For Chinese (Mandarin) we used simplified Chinese characters. We did not make any distinction between different kinds of English. The used corpus can be considered as small, however in order to be able to compare the system comportment in the three languages we prefer to favor the corpus unity than the corpus size to carry out first experiments.

[4] A word that has many substitution links can be considered to be a kind of wild-card.

[5] http://developer.android.com

[6] JUMAN Version 5. for Japanese, Apple Pie Parser Version 5.9 [17] for English and ICTCLAS for Chinese [18]

[7] The version 6.1 stems from Microsoft Office 2003 and the version 5.1 is extracted from the package Microsoft Speech SDK 5.1: http://www.microsoft.com/en-us/download/details.aspx?id=10121

3.2.1 Splitting parameter

To avoid the generation of too many nodes in languages using Latin characters, we set a minimal character length of four to split a string in English.

We also did not use the sentence starting capital letters to increase the node matching rate. For example, in the sentences "Cats are cute" and "I like cats", "Cats" and "cats" are the same word, but they will be considered different words by the system because of the capital letter. However, we kept meaningful capital characters, such as in the case of proper nouns.

It is important to note that the knowledge required to know if a word needs a capital or not depends of the language. With a bigger corpus we think this task can be avoid without an important impact to the system since the number of nodes will be sufficient to split all the words with and without a capital letter.

Characters depend on the language, but they do not make the system language-dependent. The user can input any character into the system; the output generation process will not be affected. For example, a word such as "t%&3=f" can be learnt by the system in the same way as all other words.

3.3 Experiment settings

Table 2 shows details about the subjects who participated in the evaluation of the proposed system.

Table 2: Subject's information

	Chinese	English	Japanese
Subject	7	4	13
Male	2	2	5
Female	5	2	8
Student	7	2	9
Worker	0	2	4
Age [year]	21.9	21.5	23.0

Table 3 summarizes information about the system knowledge. The number of nodes refers to the number of nodes created before the user starts the dialogue. To count the number of words, we split sentences using spaces for English, for Japanese we used the JUMAN tool [10], and for Chinese we use the Google Translator's[8] word segmentation.

Table 3: Corpus' information

Number	Chinese	English	Japanese
of nodes	1410	715	1623
of words	1246	1029	1268
of characters	1818	4853	2514
of sentences (knowledge)	100	100	100
of sentences (dialogues)	88	69	85

We can see that more a language has words more generated nodes are numerous. In addition, the number of nodes in the system is small. However, we think that bigger graphs can be easily handling as well using parallel processing.

[8]https://translate.google.com/

3.4 Experiment results

3.4.1 Speech recognition precision

The speech recognition tool does not recognize punctuation. Consequently, the subjects was not able to input point or question mark in their inputs.

The precision is calculated using Equation 5. This formula is the same as that used in the baseline.

$$P = \frac{n_c}{n_c + n_i} \tag{5}$$

- P is the precision.
- n_c is the number of correct recognitions.
- n_i is the number of incorrect recognitions.

Table 4 shows the precision for each language.

Table 4: Speech recognition precision

	Chinese	English	Japanese
Precision	93.4%	86.7%	89.3%

3.4.2 Evaluation of responses

Figure 7 shows an example of dialogue with the proposed system. Evaluation by the subjects is presented in parentheses. The sentence starting capital letters are added after the output generation.

User: What do you like?
System: I like studying *(Correct reply)*
System: Do you like studying? *(Correct reply)*

Fig. 7: Example of a dialogue with the system

The system firstly outputs "I like studying", then because the score of the nodes "studying" and "like" are high, it also outputs the sentence "Do you like studying?". This comportment can generate natural dialogues as well as erroneous outputs. That is why we will try to enhance it in further research using more parameters such as emotions and contextual nodes for the output generation.

Table 5 contains the experiment results compared to the baseline [19]. For the baseline, the values in parentheses correspond to the evaluation results for the ELIZA-like output.

Table 5: Experimentation results [%]

Language	Correct	Semi-correct	Erroneous
Chinese	25.9	17.0	57.1
English	39.1	14.2	46.7
Japanese	31.7	16.2	52.1
Baseline* (ELIZA's responses)			
Chinese	25.6 (16.0)	13.6 (30.4)	4.4 (10.0)
English	4.0 (8.4)	16.0 (53.2)	15.2 (3.2)
Japanese	14.6 (38.2)	2.5 (13.5)	8.9 (22.3)

* Results without the ELIZA's responses.

However, both parts of the baseline are evaluated in a single run. ELIZA's responses are used when no other response is available.

3.5 Results analysis

We can see that the results of the three languages are similar. In addition, they exceed the baseline's results if we exclude the ELIZA's responses. We consider ELIZA's responses as language dependent, since they are manually inserted in the system for each target language. The proposed system is able to answer most of the greetings and some questions of the user. It does not simply look for a matching rule, but it decomposes the input and analyses the nodes related to each parts in order to output the best responses.

3.5.1 Used resources

The baseline uses a morphological analysis tool and an ELIZA-type system, both of which are language-specific. However, the other parts of the output generation do not depend on language. Consequently, the system can be adapted to other languages with a minimal amount of work for any language for which such tools are provided. However, if one of these tools is not available, the adaptation task becomes much more complicated.

In comparison, the proposed approach only needs language samples to be trained and then be able to handle a dialogue in the targeted language. These samples can be simply extracted from the user's own chat logs.

To achieve a fully end-to-end language-agnostic dialogue system, it is possible to start the system without any knowledge and to allow it to acquire knowledge from the users' inputs. However, in this case the teaching process will be very annoying for the user. A better method would be to make the system assist with a dialogue between two humans and acquire knowledge in a similar way to a child hearing people around him and finally becoming able to speak. The dialogue samples used in this paper can be considered to be dialogue heard by the system during its "childhood".

4. Conclusion

In this paper, we used an unique language free algorithm to provide a real-time spoken dialogue agent to the user. We carried out experiment in Chinese, English and Japanese, and obtained similar results in all these languages. Moreover, the precision obtained exceed the baseline if we exclude ELIZA's responses.

The SeGA-ILSD system handles several languages; however it needs to be adapted to each one. In contrast, the proposed system needs no special work to be adapted to another language. For example, we can input into the system both Chinese and Japanese training samples at the same time, and the system will be able to output Chinese as well as Japanese sentences. However, it cannot preserve contextual information from one language to the other.

In our future research, we will add emotional nodes [20] to the graph in order to enable the generation of more outputs using more parameters. In addition, sharing knowledge between users [21] would help the system to acquire many different kinds of knowledge directly from the users.

References

[1] Richard S. Wallace. 2009. *The Anatomy of A.L.I.C.E.*, pp. 181-210. Parsing the Turing Test.

[2] Ridong Jiang, Rafael E. Banchs, Seokhwan Kim, Kheng Hui Yeo, Arthur Niswar and Haizhou Li. 2014. *Web-based Multimodal Multi-domain Spoken Dialogue System*. Proceedings of 5th International Workshop on Spoken Dialog Systems.

[3] Arnaud Jordan and Kenji Araki. 2013. *Spoken Dialog Processing for Acquiring Taste and Knowledge*. Proceedings of PACLING2013.

[4] Ryan T. McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith B. Hall, Slav Petrov, Hao Zhang, Oscar Täckström and others. 2013. *Universal Dependency Annotation for Multilingual Parsing*, pp. 92-97. Proceedings of ACL2013.

[5] George A. Miller. nov. 1995. *WordNet: A Lexical Database for English*, Vol. 38, No. 11, pp. 39-41. Communication of the ACM. Association for Computing Machinery.

[6] Arnaud Jordan and Kenji Araki. 2014. *A Framework for Multilingual Real-time Spoken Dialogue Agents*, pp. 24-29. Proceedings of iCAST2014.

[7] Howard C. Warren. 1921. *A History Of The Association Psychology*. Charles Scribner's Sons.

[8] Paul Ekman. 1972. *Universals and Cultural Differences in Facial Expression of Emotion*, pp. 207–283. J. Cole ed. Nebraska Symposium on Motivation. University of Nebraska Press

[9] David Hope and Bill Keller. 2013. *MaxMax: a graph-based soft clustering algorithm applied to word sense induction*, pp. 368-381. Computational Linguistics and Intelligent Text Processing. Springer.

[10] Kurohashi and Kawahara lab. 2012. *Japanese Morphological Analysis System JUMAN version 7*. Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University.

[11] Daichi Mochihashi, Takeshi Yamada and Naonori Ueda. mar. 2009. *Bayesian Unsupervised Word Segmentation with Hierarchical Language Modeling*, pp. 49-56. IPSJ SIG Notes 2009(36). Information Processing Society of Japan (IPSJ).

[12] Luca Gammaitoni, Peter Hänggi, Peter Jung and Fabio Marchesoni. 1998. *Stochastic resonance*, volume 70, pp. 223. Reviews of modern physics. American Physical Society (APS).

[13] Gerard Salton and Michael J. McGill. 1986. *Introduction to modern information retrieval*. PMcGraw-Hill, Inc.

[14] Masashi Takeuchi, Norihide Kitaoka and Seiichi Nakagawa. 2004. *A spoken dialog system activating the natural response timing using prosodic and linguistic information for chat-like conversation*, pp. 87-92. IPSJ SIG Notes 2004(15). Information Processing Society of Japan.

[15] Richard S. Wallace. 2003. *The elements of AIML style*. Alice AI Foundation.

[16] Kenji Araki and Michitomo Kuroda. 2006. *Generality of spoken dialogue system using SeGA-IL for different languages*, pp. 72-77. Proceedings of the Second IASTED. Computational Intelligence.

[17] S. Sekine and R. Grishman. 1995. *A Corpus-based Probalistic Grammar with Only Two Non-Terminals*. Proceedings of the Fourth International Workshop on Parsing Technologies.

[18] Hua-Ping Zhang, Hong-Kui Yu, De-Yi Xiong and Qun Liu. 2003. *Q.HHMM-base Chinese lexical analyzer ICTCLAS*, volume 17, pp. 184-187. Proceedings of the Second SIGHAN Workshop on Chinese Language Processing.

[19] Kenji Araki and Michitomo Kuroda. Jan. 2007. *Evaluation of Generality of SeGA-ILSD for a Chat Using Different Languages*, volume 2007, pp. 79-85. IPSJ SIG Notes 2007(7). Information Processing Society of Japan.

[20] M. Ptaszynski, P. Dybala, R. Rzepka and K. Araki. 2008. *Effective analysis of emotiveness in utterances based on features of lexical and non-lexical layer of speech*, pp. 171-174. Proceedings of the Fourteenth Annual Meeting of the Association for Natural Language Processing.

[21] Arnaud Jordan and Kenji Araki. 2013. *Comparison of two Knowledge Treatments for Questions Answering*, pp. 55-62. Proceedings of SNLP2013.

Automated Scoring of Levels of Integrative Complexity Using Machine Learning and Natural Language Processing

A. Kannan Ambili and K. Rasheed
Institute for Artificial Intelligence, University of Georgia, Athens, GA, USA

Abstract- *Conceptual/Integrative complexity (IC) is a construct used in political psychology and clinical psychology to gauge an individual's ability to consider different perspectives on a particular issue and subsequently form a conclusion that draws from the said perspectives. Presently IC is scored from text manually which is time-intensive, laborious and expensive. For a rater to be qualified to score IC, it is standard that he/she go through a rigorous training program. Consequently, there is a demand for automating the scoring, which could significantly reduce the time, expense and cognitive resources. Any algorithm that could achieve the above with a reasonable accuracy could assist in researchers who are interested in broadening the horizon for IC research. Furthermore, such a development could also assist in the design of intervention systems for reducing the potential for aggression, systems for recruitment processes and even training personnel for improving group complexity in the corporate world. In this study we used machine learning and natural language techniques to predict IC levels from text. We developed an intelligent feature called Semantic Paragraph Coherence for the prediction of IC levels in text. We achieved over 83% accuracy in a three way classification.*

Keywords: Integrative complexity, multi-class classification, semantic similarity, natural language processing

1 Introduction

Conceptual/Integrative Complexity is a construct in psychology that measures in a particular sample of text or speech the extent of differentiation and integration exhibited by the author [1]. Differentiation is the author's ability to examine differing perspectives on an issue; the higher the number of perspectives being examined on a particular, the higher is the differentiation. Integration refers to the author's skill in considering the possibly connected perspectives at hand and using these connections to form well-reasoned conclusions.[1] It has been claimed as the most used and widely validated measurement of complex thinking. Regardless of the content contained in a text sample, IC is a measure used to capture the cognitive strategies used to formulate the structure of thought of the author.

IC has been used to predict aggression in political psychology [2]. It has also been found to be an efficient predictor of performance and corporate social responsibility [3.4]. Studies have found that liberal or left-leaning politicians often tend to have high IC [2] [5] [6] Decision making can sometimes be hampered due to high IC [7]. Considering all these applications, it seems to be of immense importance that an efficient automated scoring method for IC be developed.

1.1 The need for Intelligent Features

The fact that the level of IC in a text is contingent upon the relationships that connect different perspectives could be used as a heuristic in determining the level of IC. In machine learning, the problem could be better solved with the consideration of predictors (features) that indicate in text, the amount of differentiation and integration. However, current literature has not defined algorithms that can accurately measure said constructs in an efficient manner. Linguistic Inquiry and Word Count (LIWC) [8] is a program that counts words that belong to two psychologically meaningful categories: exclusion words and conjunctions. Exclusion words (e.g. but, without, exclude) are helpful in making distinctions among different sentences. Conjunctions (e.g. and, also, although) join multiple sentences and contribute to measuring Differentiation [9]. The authors of the current body of work had performed research on the automation of the scoring of IC and were successful in obtaining accuracies of approx. 78% [10]. The authors hypothesized that the inculcation of a pre-designed NLP feature in the previously adopted machine learning methodology could improve performance and accuracy. This body of work focuses on proving that particular hypotheses as well as on improving the performance of the automated scorer for Integrative Complexity. While most text classification problems are easily solved through a bag of words approach, this particular problem requires a deeper understanding of the interlinking of arguments (or in other words, perspectives) in a given fragment of text. Consider the example given below (Taken from Peter Suedfeld's integrative complexity training workshop [1]):

> *"Advances made in the chemistry of antiseptics and the techniques of surgery are not wholly responsible for the new standards of lifesaving in war. An alert and courageous system of fully equipped yet highly mobile surgical units following close behind the assault troops has resulted in an immense saving of time between the battlefield and the operation table. In surgery time-saving is akin to lifesaving."*

The thesis for the instance is that '*the new standards of lifesaving in war*' cannot be just attributed to '*antiseptics and the techniques of surgery*'. Following it, is a contributing perspective that "*the alert and courageous system of fully equipped yet highly mobile surgical units following close behind the assault troops*. Subsequently, the author makes the differentiation more substantial with the declaration that this has "*resulted in immense saving of time*". This differentiation is immediately followed by the integration-bearing declaration that "*time-saving is akin to lifesaving*." thereby giving the thesis further support. Since there is minimal differentiation and integration, this text sample could be scored as having moderate Integrative complexity, which is equivalent to a score of 3- 5.

Therefore for a text to be qualified as having high integrative complexity, numerous differentiations have to be made, subsequently followed by integration. In other words, differentiating statements relate to each other with a non-zero amount of semantic similarity. Most differentiating and integrating statements would intuitively be semantically similar in content to an extent. In this particular example, the differentiating statements do have some semantic similarity. The semantic content of the reference made in the thesis sentence "*lifesaving in war*", is referred to semantically in content in the subsequent differentiating statement as "*the assault troops*" and "*the battlefield*" and "*immense saving*". In the final integrating conclusion, we can determine a semantic similarity to "*lifesaving*". It is this property that is exhibited by the 'integrative-ly complex' that could be exploited in the prediction of levels of integrative complexity.

1.2 Semantic Paragraph Coherence as a feature

Measuring semantic similarity between sentences could be translated into measuring the semantic similarity of words that carry the most information in these sentences. Most often the semantic content in sentences comes from the nouns, verbs and adjectives and to a lesser degree on adverbs, prepositions and the rest. Traditionally, semantic similarity between sentences would be limited to analyzing the similarity between shared words [11], which worked reasonably well in texts of longer lengths. However, for shorter texts, a method which focused on the semantic meaning of the word rather than the word itself was required.

1.2.1 Semantic similarity between words.

The method for calculating semantic similarity between words in this paper is based on Li, Bandar & McLean's work in 2003 [12], where the similarity of two words is calculated using a hierarchical semantic knowledge bases (e.g. WordNet [13][14][15] The work presented in this paper calculated semantic similarity as a function of path length (the minimum number of words lying between the considered words in the hierarchical knowledge base) and depth (the depth of the subsumer in the hierarchy). Path length and depth are both derived from a lexical knowledge base. α and β are parameters that are used to scale the contributions of path length and depth respectively. Let the semantic similarity between two words w_1 and w_2 be noted by $S(w_1, w_2)$. Then according to the word similarity measure proposed in [12]:

$$S(w_1, w_2) = f(l).f(h) \qquad (1)$$

In the above measure, $\alpha \geq 0$ and $\beta > 0$. The proposed optimal values are: $\alpha = 0.2$, and $\beta = 0.6$[12].

1.3 Implementation Details

This section describes the method used to calculate the semantic similarity between two words. Only a brief account is given here, for further explanation, please refer to the original paper [12] [16]. The method was coded in SWI Prolog [17], since WordNet version 3.0 [13] [14] [15] was also available in Prolog.

1.3.1 Contribution of path length

The path length between two words in a hierarchical knowledge base can vary between 0 to large numbers. Hence the function should be designed so that it will have values ranging from 0 to 1. This function will depend on three cases: In the first case, $f(l) = 1$; if w_1 and w_2 belong to the same concept. In the case that the two words do not belong to the same concept, but have the same word linking them, their semantic similarity is calculated as:

$$f(l) = e^{-\alpha l} \qquad (2)$$

l is the sum of the number of words leading up from both words to the same word.

1.3.2 Contribution of depth

The first common hypernym between w_1 and w_2 is called the subsumer of w_1 and w_2. h is the depth of the subsumer in the hierarchical semantic nets. For example consider the path between 'boy' and 'girl', the path is 'boy-male-person-female-girl', then 'person' is the subsumer for 'boy' and 'girl'. The depth is calculated by counting the levels from the subsumer level to the top of the lexical hierarchy. The subsumer of the shortest path is considered in deriving the depth of the subsumer in case of polysemous words.

$$f(h) = \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \qquad (3)$$

1.3.3 Calculation of semantic similarity between words

The semantic similarity between two words w_1 and w_2 be noted by $S(w_1, w_2)$ (i.e a product of (1) and (2)) [12]:

$$S(w_1, w_2) = f(l).f(h) = e^{-\alpha l}.\frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \qquad (4)$$

1.4. Calculation of Semantic Paragraph Coherence

Our work proposes the use of Semantic Paragraph Coherence as a feature to predict Integrative Complexity. Scoring of Integrative complexity involves scoring the text by determining the levels of differentiation and integration. A text that has been scored extremely low on Integrative Complexity can be seen as a series of unconnected discourse, or as a paragraph that focuses on a single thesis with descriptive statements. Such a fragment of text wouldn't necessarily make references to the semantic information present in the thesis later on in the paragraph to discuss different perspectives or come to a well-reasoned plausible conclusion. It is this assumption that is behind the development of the proposed feature.

The proposed method calculates Semantic Paragraph Coherence by calculating the semantic similarity between the first sentence in a sample text and the rest of the sentences in the sample text. The calculation of semantic similarity between the sentences is limited to nouns and verbs. The assumption behind this choice is that nouns and verbs carry the most semantic information, and at the same time this keeps the number of calculations to a smaller number, thereby reducing computational complexity.

The calculation of Semantic Paragraph Coherence is a two-step process. Initially, the calculation of all the semantic similarities of the words in the first sentence with every other word in the rest of the sentences in the paragraph is performed. This step itself is composed of two steps. For each word, w_j present in the first sentence (otherwise named as the topic sentence), the semantic similarity between itself and every relevant word, w_i in the rest of the paragraph is calculated. Let this value be s_{ij}. Here m is the maximum value of i, *i.e.* the total number of relevant words present in the paragraph (with the exception of the topic sentence). Therefore for a word w_j present in the topic sentence, the associated semantic similarities with the rest of the paragraph is formulated as below. Let $g(w_j)$ be this measure. Then:

$$g(w_j) = \frac{1}{m}\sum_{i}^{m} s_{ij} \qquad (5)$$

Let n be the total number of relevant words in the topic sentence. Then the total associated semantic similarity value of the paragraph could be treated as, sum.

$$sum = \sum_{j}^{n} g(w_j) \qquad (6)$$

Semantic Paragraph Coherence, P could be calculated as:

$$P = e^{-1*sum} \qquad (7)$$

Consider the below statement, taken from Dr. Suedfeld's Integrative complexity training workshop page [1]:

> *' The experience of life's hardships and comforts fosters an awareness of both the value and impermanence of the moment ; all of these influence and are influenced by the meaning we make-which is further negotiated over time and through interaction with others-and manifested in our autobiographies..'*

The sample text (shown above) scores high on Integrative Complexity. Our proposed method scored a P value of 0.30. Whereas the text given below [1] scores low on Integrative complexity. And has a P score of 0.930:

> *'So much for my apologies. There are plenty of them, perhaps too many. Were it not for your letter I should feel myself almost guiltless. But since you apparently went on thinking about the purse and possibly even searching for it, all apologies are of course inadequate and I must resort to asking you not to spoil my pleasure in finding the purse, by being angry with me for my negligence. For that would be-even though the purse contained 900 crowns (which may explain my haste in telling you) a tremendously high finder's fee which I would be obliged to pay to lucky chance. You won't do that I'm sure."*

It has to be noted that for some samples Semantic Paragraph Coherence may not be the best predictor. These instances could be identified as outliers. From these examples, it could be inferred that Semantic Paragraph Coherence could act as a predictor for scoring Integrative complexity with non-zero error.

2 Approach

2.1 Data Selection and Experimental Setup

The data for the project consisted of 83 text samples along with the scores provided by manual scoring by trained coders. The data was taken from Suedfeld's Complexity Materials Download Page [1], where the data has been made available free for download for scorers who want to practice scoring.

Each instance has been scored on a 1-7 scale. The first step in Pre-processing involved binning the instances into three bins. Instances that have been given IC scores of 1 or 2, were classified as having low IC and therefore given a class label of 'low'. Similarly instances that have been scored IC scores of 3, 4 or 5 were classified as having medium levels of IC, and were given a class label of 'mid'. Subsequently, instances that were scored IC scores of 6, 7 were classified as having high IC and were given class labels 'high'.

The code for extracting the value of the Semantic Paragraph Coherence feature was written in SWI Prolog [18]. The code made use of WordNet 3.0 [13] [14] [15] written in Prolog to design the feature. Then the code was run on each instance to calculate the Semantic Paragraph Coherence of each instance. The code for calculating the length of a paragraph (in words) was also calculated in Prolog.

Then, the data was cleaned and converted into an ARFF (Attribute Relation File Format) file format for use in Weka. [17]. Feature selection methods played a huge role in this text-classification problem. Using the String to Word Vector filter in Weka [17], the string in the text attribute of each instance is converted to a set of attributes representing word occurrences, where each word is converted to lowercase before processing. Along with the bag of word features, we also included the Semantic Paragraph Coherence measure and length of the text sample. The number of attributes were reduced significantly using Attribute Selection methods.

2.2 Learning Methods

The project used several machine learning algorithms for experimenting with the data. For this purpose, the open source machine learning software, Weka [17] was used. The algorithms that are mentioned here, are the ones which have reported some of the best performances. They are Bagging, the Multinomial Logistic Regression model, Multi-layer perceptron, AdaBoost.M1 and the Multi-class classifier.

Adaboost (short for Adaptive Boosting) is a boosting algorithm that can be used to significantly improve classifier performance given that its weak learners can predict with a rates a little better than random guessing. A weak learning algorithm is run on different parts of the distribution of the training data and then combined to form a composite classifier, this is the basis of boosting [19]. AdaBoost.M1 is a special case of AdaBoost where easy examples that are correctly classified by the weak learning algorithms are given less weightage than examples that get misclassified by the weak learning hypotheses.

The Multinomial Logistic Regression Model is often used in Natural Language Processing applications because they do not assume statistical independence of features, as is often the case with text. The model is a generalization of the logistic regression model for multi-class problems. The probabilities describing the outcomes of an instances are modeled as a function of its features, using a logistic function.

Another classifier that we experimented with was the Multi-class classifier- suitable for the multi-class classification problem. The Meta classifier used binary classifiers to solve the 3 –class classification problem. The binary classifiers used for experimentation were the logistic regression and the multi-layer perceptron. Popular multi-classification methods like 1-against-1 and pairwise classification were used.

The Multilayer Perceptron (MLP) was also used in the experimentation part. An MLP consisting of multiple layers of nodes in a directed graph, uses a supervised learning techniques called backpropagation for training the classifier. The MLP used in this work contained only nodes that had sigmoid functions as activation functions. The learning rate was set at 0.3 and momentum was set at 0.2.

Bagging (also known as Bootstrap Aggregation) is ensemble meta-learning algorithm that is used to reduce variance and over-fitting. This algorithms grants 'votes' to base classifiers that are trained on different bootstrap samples. A final classifier is built from all the base classifiers trained on all the bootstrap samples, whose prediction is based on the most predicted by its base classifiers.

3 Evaluation

The performance of the multi-class classification methods is tested through stratified 10-fold cross-validation. Considering the limited amount of data, especially in the context of a multi-class classification problem, the standard way of predicting the error rate of a learning technique is to use stratified 10-fold cross-validation. Classification accuracy has been used as one of the performance measures for this problem. However emphasis should be given to performance measures such as precision, recall and F-1 measures, as they tend to be better measures when evaluating small classes (Manning et al., 2008).

4 Results

Results obtained were promising. Table-II shows the classification accuracies and r. Table-I show the precision, recall and F-1 measures of the classifications. Overall, higher values for classification accuracies and effectiveness measures have been reported for the proposed approach. The highest classification accuracy was reported by the Multinomial Logistic Regression Model with a ridge estimator-II. The same classifier also reported the highest precision and recall. Some of the classifiers have high precision (1.000) for the class high. While some of them have high recall for the class mid. Classification accuracies of 80% to 83% are at par with the human rater reliability of 80%. But since the dataset is relatively small, more focus should be given to the Precision, Recall and F-1 measures.

The addition of the Semantic Paragraph Coherence and length of the text sample as features have influenced the performance of the algorithms in a positive manner. The combination of these features along with a bag of word approach have produced a decent performance. Experiments conducted to evaluate the contribution of the newly designed feature produced favorable results. Results showed that the Semantic Paragraph Coherence feature was able to assist in the three-way classification.

Table I: Precision, Recall and F-1 measures for Bagging, Multi-Class Classifier and Multinomial Logistic Regression Model with a ridge estimator-II

	Bagging			**Multi-Class Classifier**			**Multinomial logistic regression with a ridge estimator-II**		
Class	**Precision**	**Recall**	**F-1 measure**	**Precision**	**Recall**	**F-1 measure**	**Precision**	**Recall**	**F-1 measure**
low	0.800	0.500	0.615	0.923	0.500	0.649	0.857	0.500	0.632
mid	0.790	**1.000**	0.883	0.762	0.980	0.857	0.803	**1.000**	0.891
high	**1.000**	0.600	0.750	0.857	0.600	0.706	**1.000**	0.800	0.889
Weighted. Avg.	**0.818**	**0.807**	0.790	**0.820**	0.795	0.779	**0.843**	**0.831**	**0.816**

Figure 1: Performance comparison for multinomial logistic regression model-II (with a ridge estimator)

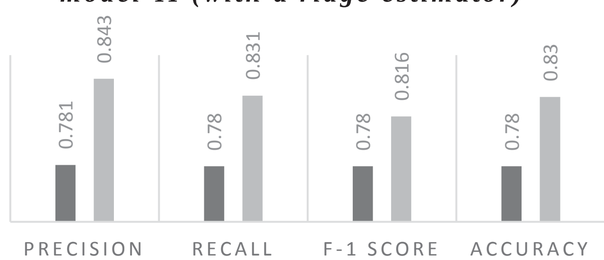


Figure 2: Performance comparison for bagging

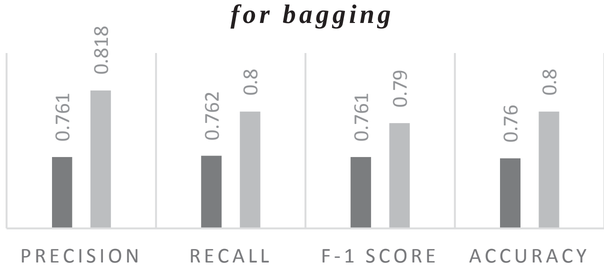


Figure 3: Performance comparison for multi-layer perceptron

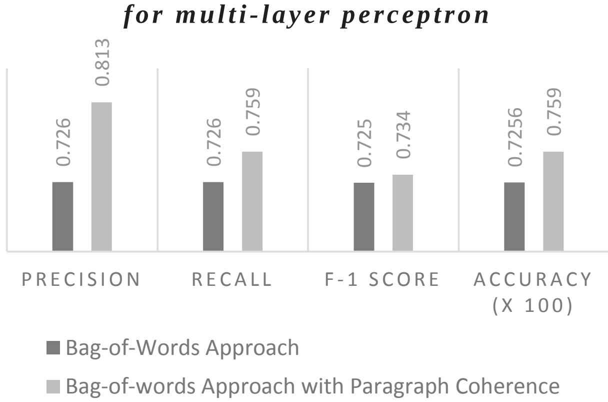


Table II: Classification accuracies

Classifier	**Specifications and comments**	**Accuracy**
Multi-layer Perceptron	*Backpropagation algorithm*	75.9%
AdaBoostM1- I	*Base classifiers and their weights:* *Random forest of 10 trees, each constructed while considering 5 random features.*	73.5%
AdaBoostM1- II	*Base classifier: SMO with Polykernel*	77%
Multi-Class Classifier	*Base classifier: Multinomial logistic regression with a ridge estimator* *Method: 1-against-all*	79.5181 %
Bagging	*Base classifier: Multinomial logistic regression with a ridge estimator*	80.7229 %
Multinomial logistic regression with a ridge estimator-I		80.7229 %
Multinomial logistic regression with a ridge estimator-II	*Uses Conjugate Gradient Descent for search for paramenters*	**83.1325%**

Figure. 1, Figure 2 And Figure 3 show performance comparisons for three classification algorithms with the inclusion and exclusion of the Semantic Paragraph Coherence feature. From these figures, it can be easily seen that an approach involving the Semantic Paragraph Coherence feature has superior prediction accuracies than an approach without it. The effectiveness measures reported by the proposed approach are also higher than the basic bag-of-words approach.

5 Conclusion

The contribution of the NLP feature, Semantic Paragraph Coherence together with length of the text cannot be overlooked in the light of the results obtained. That a pure bag of word approach may be limiting and that it should be combined with a knowledge engineering approach is particularly insightful. Classification accuracies that are at par with expert rater-reliability are unheard of in the work done on the automation of integrative complexity scoring. However, the precision, recall and F-1 measure are better performance measures in this body of work, considering the size of the dataset.

Future work could experiment with a much larger dataset. The development of NLP-focused features to aid in the detection of differentiation and integration could not only assist in the scoring of Integrative Complexity, but also help us attain a deeper understanding of human language, and the structure of thought.

6 References

[1] P. Suedfeld, P. E. Tetlock, & S. Streufert, Conceptual/integrative complexity. In C. P. Smith, J. W. Atkinson, D. C. McClelland, and J. Veroff (Eds.), *Motivation and personality: Handbook of thematic content analysis* (pp. 393-400). New York: Cambridge University Press, 1992.

[2] P. E Tetlock, “Cognitive style and political ideology”, *Journal of Personality and Social Psychology*, vol. 45, pp. 118-126, 1983.

[3] D. H. Gruenfeld & A. B. Hollingshead, “Sociocognition in work groups: The evolution of group integrative complexity and its relation to task performance”, *Small Group Research*, vol. 24, pp. 383-405, 1993

[4] G. J. Feist, “Personality and working style predictors of integrative complexity: A study of scientists' thinking about research and teaching”, *Journal of Personality and Social Psychology*, vol. 67, pp. 474-484, 1994.

[5] P. E. Tetlock, “Cognitive style and political belief systems in the British House of Commons”, *Journal of Personality and Social Psychology*, vol. 46, pp. 365-375, 1984.

[6] P.E. Tetlock., K. A. Hannum, & P. M. Micheletti, “Stability and change in the complexity of senatorial debate: Testing the cognitive versus rhetorical style hypotheses”, *Journal of Personality and Social Psychology*, vol. 46, pp. 979-990, 1984

[7] P. E. Tetlock, & R. Boettger, “Cognitive and rhetorical styles of traditionalist and reformist Soviet politicians: A content analysis study.” *Political Psychology*, vol. 10, pp. 209-232, 1989.

[8] Pennebaker, J. W., Booth, R. J., & Francis, M. E. (2007). Linguistic Inquiry and Word Count: LIWC [Computer software]. Austin, TX: LIWC.net

[9] Y. R. Tausczik & J. W. Pennebaker, “The psychological meaning of words: LIWC and computerized text analysis methods*.” Journal of Language and Social Psychology*, vol. 22, pp. 24-54, 2010.

[10] A. K. Ambili, and K. M. Rasheed. "Automated Scoring of the Level of Integrative Complexity from Text Using Machine Learning." In *Machine Learning and Applications (ICMLA), 2014 13th International Conference on*, pp. 300-305. IEEE, 2014.

[11] C. T. Meadow, B. R. Boyce, and D. H. Kraft, *Text Information Retrieval Systems*. 2nd. Ed. Academic Press, 2000

[12] Y. Li, Z. A. Bandar, & D. McLean, “An approach for measuring semantic similarity between words using multiple information sources”, *Knowledge and Data Engineering*, IEEE Transactions on, vol. 15(4), pp. 871-882, 2003.

[13] C. Fellbaum, C. *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[14] (2010) WordNet: An electronic lexical database.. [Online] Available: http://www. cogsci. princeton. edu/wn

[15] A. G. Miller, “WordNet: a lexical database for English”, *Communications of the ACM*, vol. 38(11), pp. 39-41, 1995.

[16] Li, Y., Bandar, Z., McLean, D., & O'Shea, J. (2004). A Method for Measuring Sentence Similarity and its Application to Conversational Agents. In FLAIRS Conference (pp. 820-825).

[17] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann & I. H. Witten, “The WEKA Data Mining Software: An Update”*; SIGKDD Explorations*, vol. 11, Issue 1, 2009.

[18] J. Wielemaker, T. Schrijvers, M. Triska & T. Lager, “Swi-prolog”, *Theory and Practice of Logic Programming*, vol. 12(1-2), pp. 67-96, 2012.

[19] Y. Freund & R. E. Schapire, “Experiments with a new boosting algorithm”, In *ICML*, vol. 96, pp. 148-156. 1996.

Semantic Characterization of Academic and Occupational Profiles Based on Competencies

Alexandra González Eras[1] **Jose Aguilar** [1,2]
[1] Computer Sciences Department, Universidad Tecnica Particular de Loja, Ecuador
[2] CEMISID, Engineering Faculty, Universidad de Los Andes, Mérida, Venezuela
acgonzalez@utp.edu.ec, aguilar@ula.ve

Abstract - *This research shows an architecture for extraction, comparison and feedback of competencies from the characterization of its components. The main product is a scheme that facilitates skills and knowledge detection in documents, as well as the identification of more complex concepts, such as competencies. This research is a first contribution to the development of a system for comparison and update profiles, that is adaptable to the context, and facilitates understanding of the competency dynamics on education and employment environments.*

Keywords: Natural Language Processing, linguistic patterns, Competency.

1 Introduction

For universities and employers manage the competencies is a two way process, including skills identification to create new graduate profiles and determine qualified professional skills to fill a working place. However, in reality it is almost impossible to compare competencies, mainly due to incompatible profiles [Fazel-Zarandi and Fox, 2009] [Stevens, 2013]. In addition, the information published on university websites and work platforms is unstructured, ambiguous, and sometimes incomplete [Fazel-Zarandi, 2013]. Looking for a solution, models and platforms have been proposed in order to profiles standardization [Draganidis and Mentzas, 2006] and comparison through competency frameworks[1]. Nevertheless, these tools are rarely used by the actors, or it have only been proposed for one language context without a real application in others. As a result, universities can not identify the requirements of employers, and in turn, employers can not identify new graduate profiles aligned with their job offers.

The goal is to establish a management system of academic and labor profiles, allowing extraction, comparison and update of competencies, based on components characterization. First, we give an overview of the system architecture, and then focus on the characterization module, which models skills and knowledge, and proposes how they are connected to form more complex concepts, such as competencies. The resulting scheme is the management system cornerstone, which is applied on competency profiles in Spanish.

Notably, we initially focused on modeling skills and knowledge, because they are the most common concepts associated with the definition of competency in both contexts: labor and academic; in addition, skill and knowledge patterns are trackable in the profiles. This allows us to perform a less subjective characterization of competencies compared to other constituent elements, such as actitude, and value [Yahiaoui et al., 2006].

Thus, this research displays a overview of the system architecture detailing the characterization module, then we conducted two experiments, one on competency profiles to determine the ability of the scheme in recognizing patterns, and second analyses the ability of patterns to comparing the profiles against two bodies of knowledge (SWEBOK and DISCO). Finally, we made preliminary conclusions and future work of our research.

2 Academia versus occupational environment

Training for work is a mix between education, work experience and specific training acquired throughout life, hence competencies are defined and constructed in social practice, as a joint effort between companies, workers and educators. From a business standpoint, c*ompetence management* is the systematic development of human resources in organizations, and in that sense, competence management systems should support this systematic development [Lindgren et al., 2004].

Typically, professional skills and knowledge are adquiridad at university, where students develop a competence profile. In [Dorn and Pichlmair, 2007] states that through this profile, we may perform a gap analysis of the student, and investigate which additional knowledge and skills should be acquired, to achieve professional profile defined by the university or industry groups. Also, the student can use the profile to create a summary and seek employment. Similarly, companies seek candidates based graduate profile of universities to also consider other aspects

[1] e- Qualifications Framework, available online at http://www.ecompetences.eu/

such as additional training and experience [Dorn et al., 2007].

The search for suitable candidates for job openings, or careers covering work places, has always been a complex task, mainly due to different interpretations that each actor has about skills, and hence its many forms representing [Fazel-Zarandi and Fox, 2009]. For example, the term competence in the work context can be used sometimes to refer to actions and their consequences, sometimes as cognitive abilities and personal characteristics [Stevens, 2013]; whereas in academia competencies are expressed in terms of qualifications and certifications (such as diplomas) [Malzahn et al., 2013], or learning outcomes within education processes [Paquette, 2007].

On the other hand, the profiles clearly not describe the competencies or the competency elements, so we can not make a comparison between them [Paquette et al., 2012]. Job offers meet job characteristics such as activities or roles, rather than skills [Bizer et al., 2005]. Moreover, the curriculum of a university degree generally has dependencies between courses, and a general description of the learning objectives to be met to achieve a score [Dorn and Pichlmair, 2007].

Therefore, we need to establish a common environment to achieve a comparison. Through similarity measures and a description of the context around the notion of competencies, we can define a scheme which covers aspects such as extraction, comparison and prediction of competencies. This proposal pursues this objective, which will be explained in the following sections.

3 Architecture

The figure 1 shows the architecture of our proposal, in which the academic and job profiles get into an iterative pipeline process comprising 4 phases: characterization, extraction, comparison and updating.

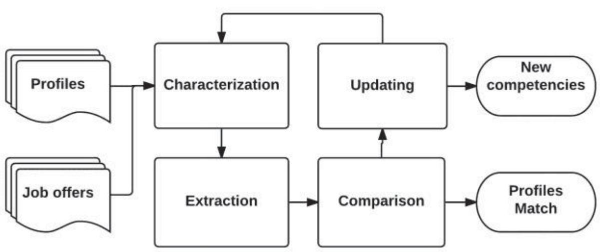


Figure 1: System's architecture

3.1 Characterization

In this phase we propose a scheme based on standard patterns, taxonomies and logical descriptions, which allows the recognition of the competency descriptor elements such as skills and knowledge, to achieve a first scheme of characterization. The characterization phase is based on linguistic patterns, semantic rules and indicators of similarity / dissimilarity, which are the basic input for the following phases.

3.2 Extraction

The extraction phase applies the scheme characterization over the documents. To develop this, low level NLP is combined with the linguistic patterns and semantic rules (defined in the preceding stage) in order to label text pieces in university and job profiles. This phase provides text segments (patterns) which will be compared to determine their similarity.

3.3 Comparison

At this stage we establish measures of similarity between skill, knowledge and competency patterns (identified in the previous phase), achieving levels of closeness between them. To do this, we combine different similarity measures and clustering techniques [Huang, 2008]. This phase will have two output: the first are the groups (clusters) of similar competencies, and the second comprising the remaining competencies. In addition, a set of similarity/dissimilarity indicators for each group are presented.

3.4 Updating

In this phase we analyse groups of similar and dissimilar competencies to determine the inter/intra cluster relationship, looking for several things, among others, those competencies that represent outliers inside and outside the clusters. The output of this phase is recommendations such as: groups of new competencies, more common skills, new knowledge domains, etc.

In this research we focus on the characterization phase, in order to establish our first analysis schema of profiles.

4 Characterization Module

The figure 2 displays the activities carried out in the characterization phase. First, we defined the axioms underlying the scheme. To do this, we used competency concepts and its elements (skills and knowledge), applied in each domain (academic and professional) to identify common contexts. Then, we suggest logical descriptions to characterize these contexts (patterns). To this end, we assumed that there are many definitions of competency and of its components; therefore, we performed a selection process based on a criterion. This criterion is determined by the purpose of comparison.

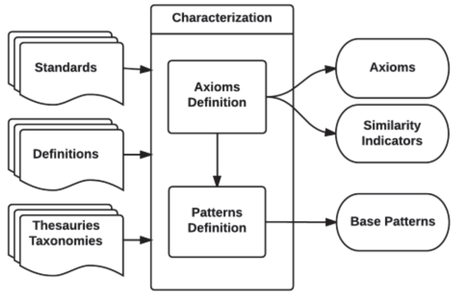


Figure 2: Characterization module

For instance, if the objective is to compare Software Engineering profiles, from South American universities against a set of job offers, the approach includes choosing skill and knowledge concepts, and identify standards, thesauri and taxonomies related to academic and professional context proposed. With them, we would define patterns and initial indicators on Software engineering domain to analyze the academic and work competencies. Thus, we see that the pattern definitions will be determined by generic aspects related to the definition of term competency (axioms), and by issues related to the domain where we want to make the comparison (in our previous example, Software Engineering)

4.1 Axioms definition

The axioms play the role of establishing a generic framework for competency analysis. We carried out the basic definitions of competency elements to consider (knowledge and skills), and used a formal description to make them computable. In the case of university profiles, the axioms that contextualize skills in the academic domain were:

Definition 1: In reference to [Paquette et al., 2012], competency is the ability to use knowledge and skills in work or study situations. According to DISCO competency thesauri [Muller-Riedlhuber, 2009], competencies represent the skills and knowledge of an individual in a specific domain. Taking up the above proposed criteria, examples of these definitions in the domain of software engineering are:

Example 1: Demostrar conocimientos de algorítmica y programación
Example 2: Utilizar los algoritmos de procesamiento de datos para almacenar, acceder y analizar información.

Figure 3: Example of academic competencies

Definition 2: According to DISCO competency thesauri [Muller-Riedlhuber, 2009], skill is the ability to apply knowledge and use know-how to complete tasks and solve problems. In [Paquette, 2007] skills has a taxonomical structured according to the knowledge cognitive level, and can range from generic to specific. For the same example of Software Engineering skills could be:

Example 3: Skills related to the generic skill ***producir***

Generic skill	Level	
	1	2
Producir	analizar	Deducir, clasificar, predecir, diagnosticar
	sintetizar	Planear, modelar, diseñar

Table 1: Example of skills

Definition 3: Citating DISCO [Muller-Riedlhuber, 2009], knowledge comprises all items related to a field of work or study. According to SWEBOK[2] [Guide in Spanish], knowledge is a hierarchical framework of entities. For the same example of software engineering, knowledge is presented as follows:

Example 4: Knowledge in Software Engineering

Knowledge area	Knowledge sub-area
Requerimientos de Software	• Fundamentos de los requisitos • Captura de los requisitos
Diseño de Software	• Estructura y arquitectura de Software • Análisis de la calidad del Software

Table 2: Example of knowledge

For the employment domain, the axioms that characterize the competencies are:

Definition 4: In reference to [Yahiaoui et al., 2006], competency comprises the domain/proficiency levels required in a particular knowledge area. In [Bourque et al., 2003], they affirm that competencies are complemented by the experience, and according to that, competency levels in knowledge areas are suggested. Continuing with the supposed case of Software Engineering domain, competency examples would be:

Example 5: Different proposals of competency levels found in the definitions and standards are:

Proposal	Competency
Competence levels [Yahiaoui et al., 2006]	Basic (B o 20%), Application (A o 50%), Master (M o 70%) or Expert (E o 90%).
SW Engineering profiles [Bourque et al., 2003]	NG (New Graduate), G+4 (Graduate with four years of experience), EWSE (Experienced software engineer working in a

[2] SWEBOK Guideline, available online at http://www.swebok.org

	software engineering process group)
Competence proficiency levels (e-CF)[1]	e-1:EQF3 , e-2: EQF 4 and 5, e-3:EQF6, e-4:EQF7 and e-5:EQF8

Table 3: Example of competency

Definition 5: According to DISCO [Muller-Riedlhuber, 2009], skill is the ability and determination to play a role or function. In the context of e-CF[1], the skills are defined in relation to five general action areas which comprise their own abilities. In [Bourque et al., 2003], skills are represented as taxonomic structures that reflect six cognitive levels with their own synonyms, as in the case of Bloom's taxonomy [Capdevila, 2011]. The following example explains this definition:

Example 6: Different skill proposals found in definitions and standards are:

Proposal	Skill
Areas[1]	manage, enable, run, build and plan
Taxonomical levels [Bourque et al., 2003]	knowledge, comprehension, application, analysis, synthesis and evaluation

Table 4: Example of skill

Definition 6: Knowledge is an object of competency with a specific level of competency[1]. According to [Yahiaoui et al., 2006], the objects of competency may be technological topics belonging to specific knowledge area or software artifacts. To Sicilia [Sicilia et al., 2005], an artifact is "an at least partially tangible thing which was intentionally created by a person", on the other hand, competency levels can be defined based on the examples given in the table 3. The following example explains this definition:

Example 7: Artifacts in Software Engineering

Area	Knowledge
Requerimientos de Software	Documentos de requisitos
Diseño de Software	Patrón de diseño, traza de requerimiento, programa

Table 5: Example of knowledge

4.2 Patterns definition

Once raised the axioms, we proceed to define the linguistic patterns for each of them. In the Spanish language, the basic structure of a sentence consists of two parts, subject and predicate, being the core of the subject noun and the verb predicate core. In Computational Linguistics, the subject of the sentence is associated with a noun phrase (Noun phrase), and predicate with a verb phrase (Verb Phrase) [Manning and Sch¨utze, 1999]. In the case of profiles, the expressions used to describe competencies are descriptive sentences, but with the difference that they have a tacit subject. The Figure 4 presents the linguistic analysis of a classical competency sentence: “Gerenciar centros de cómputo”.

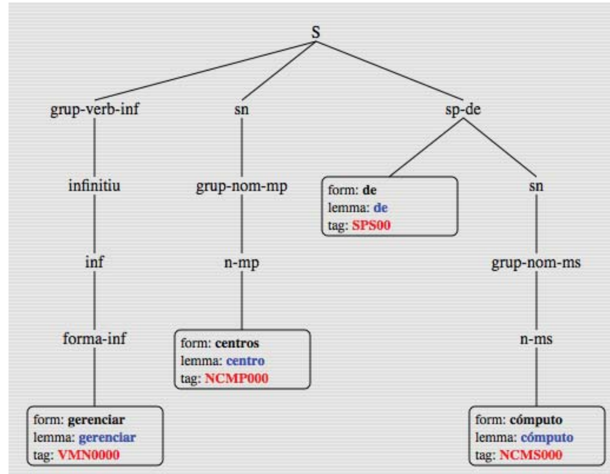


Figure 4: Linguistic analysis of a classic competency sentence in Freeling[3]

Based on the axioms defined in paragraph 4.1 for the academic context, we can say that competency is the union of a verb phrase and noun phrase, where the noun phrase (Noun phrase) represents knowledge and verbal phrase (Verb phrase) represents skills. Table 6 presents the proposed patterns according to the axioms for the academic context, as well as examples found in the profiles.

Element	Pattern	Example
Knowledge	Noun Phrase: (NP) [(NP)(NP)][(NP)(Prep)(NP)]	-Proyecto -Sistema Operativo -Programa de software
Skill	Verb Phrase: (VP)[Flexión Verbal(NP)]	-Diseñar -Gestionar -Gestión
Competency	Noun Phrase + Verb Phrase: (NP) + (VP) [(NP)(NP)][(NP)(Prep)(NP)] + (VP)[Flexión Verbal(NP)]	-Diseñar + programas de software -Gestión + de sistemas operativos

Table 6: Academic competency patterns

Similarly, we can define a competency characterization to occupational context (see Table 7), where competences are defined as noun phrases (Noun phrase). Furthermore, the

[3] Freeling, available online at http://nlp.lsi.upc.edu/freeling/demo/demo.php

skills are presented as the combination of verbal and noun phrases, while knowledge is formed based on noun phrases, as in the academic context.

Element	Pattern	Example
Knowledge	Noun Phrase: (NP)|[(NP)(NP)] | [(NP)(Prep)(NP)]	-Java -Sistemas Operativos -Oracle -Bases de Datos -Patrones de diseño
Skill	Verb Phrase + Noun Phrase: (VP) + (NP) (VP)[Flexión Verbal(NP)] + [(NP)(NP)][(NP)(Prep)(NP)]	-Gerenciar + redes de computadoras -Gestión + de servidores Linux
Competency	Noun Phrase: (NP) (NP) | [(NP)(NP)] | [(NP)(Prep)(NP)]	-Desarrollador Máster -Director de Proyectos Junior

Table 7: Occupational competency patterns

5 Experimentation

5.1 First Experiment: Text analysis

With the purpose to analyze the scheme patterns in both contexts, we make a first approximation of its uses in the profiles taken from universities and industrial contexts. The Figure 5 shows an example of our characterization's definitions in pieces of competency profiles, taken from universities and work contexts, where we see knowledge patterns highlighted in blue and skill patterns highlighted in red.

University profile	***Título***: ***Ingeniero en Software*** ***Función:*** **Diseñar, implementar y evaluar componentes, programas y sistemas.**
Job Offer	***Cargo***: **Desarrollador de Software** ***Descripción:*** Ingeniero de Software especializado en **programación de sistemas**. El candidato deberá tener la habilidad de **priorizar** actividades para **alcanzar** fechas de entrega. (Es deseable que tenga habilidades en la **gestión de proyectos**)

Figure 5: Pattern annotation over profiles

Since in both contexts competencies are interpreted differently, we can note difficulties in connecting skills in the texts. At first glance, we show that university profiles contain complex sentences with several words tagged as skills and knowledge. On the other hand, job offers contain a greater presence of knowledge descriptions, although some of them can be understood as skills, transforming these words in its verbal inflection (Programming → To program). In conclusion, there is a high degree of ambiguity in the text, which does not fully identify competencies or their components.

From this input, the comparison phase of our system may establish metrics, strategies to compare the information identified in the analysis of texts of competencies, obtained by our patterns, as the case that we come to show.

5.2 Second experiment: standards-based analysis

The Figures 6 and 7 show an example of alignment of the profiles presented in the first experiment (see figure 5), through two standards related to Computer Sciences domain. Figure 6 compares the competency of the university profile: "Diseñar, implementar y evaluar componentes y programas de sistemas" (Design, Implement and evaluate-system program components) with two standards: DISCO [M ̈uller-Riedlhuber, 2009] and SWEBOK [Sicilia et al., 2005].

To this end, we used the characterization patterns defined in section 4.2, and an iterative algorithm to compare the patterns present in the sentence against the lower levels of the hierarchy of each body of knowledge. We selected a candidate pattern based on pattern matching hits, which is now compared against the lower levels of the hierarchy to find a match. For this experiment, the pattern (system program) becomes the candidate pattern. Figure 6 shows the result of the iteration, where the path through the hierarchies gets some indicators such as: domains of knowledge, and the competence levels related to the pattern.

PATTERN	DISCO	SWEBOK
[system program]	Computing **(1st Level)** – Programming **(2nd Level)** **Term:** **System Programming** **Phrase** • create program modules and procedures	**KA:** Software Construction **(1st Level)** – **Breakdown topic** **(2nd Level)** Practical Considerations – **Construction languages** **(3rd Level)** – **Topic** **System Programming Notations**

Figure 6: Tracking academic patterns on DISCO and SWEBOK

We can see that there are ambiguities in text segments of profiles, clearly demonstrated by the different topics that form the comparison route (systems software, systems programming, systems programming notation). These three topics reflect different domains of knowledge in relation to the original pattern, although there is a certain similarity between them.

We repeated this analysis on job offer patterns. Figure 7 shows the result of the experiment. After several iterations, the candidate pattern was "programación de sistemas" (system programming). In this case, the comparison route through the hierarchy reveals a relative similarity between the candidate pattern in figure 7 (system program) and the second level of DISCO (Computing/Programming), and the third level of SWEBOK (Practical considerations/Construction Languages).

PATTERN	DISCO	SWEBOK
[software programing] [system programming]	Computing (1st Level) – Programming (2nd Level) –IT Consultancy (3rd Level) – SW. Evaluation – SW Design **Term:** System Programmes **Phrase** • analyse the efficiency of applied software programmes • perform tests on established software programmes	**KA:** Software Construction (1st Level) – **Breakdown topic** (2nd Level) Practical Considerations – **Construction languages** (3rd Level) – **Topic** **System Programming Notations**

Figure 7: Tracking occupational patterns on DISCO and SWEBOK

The results of this second set of experiments, gives us a first idea of the level of similarity between academic profile and the job offer profile. Therefore, we suppose that it is possible to use the standards as a means of comparing the profiles in both contexts [Rudzajs and Kirikova, 2011], thereby establishing a midpoint that solves the problem of ambiguity. The comparison phase has on these results an important input. For that, we need to refine concepts like "relative similarity", used on the comparison phase, based on the linguistic patterns defined in this paper.

6 Conclusions

Our proposal of a characterization module based on a modeling approach, allows that the Profile Management Architecture will be adaptable to the context. The definition of axioms and linguistic patterns for competencies and its basic elements, contributes to the development of the other modules of the architecture. For example, facilitates the process of pattern extraction (extraction phase), the matching of competencies (comparison phase), and also provide a framework for the upgrade of the competencies (updating phase).

The experiments lead us to think that the patterns proposed in the characterization module allow us the identification of knowledge and skill within documents, and provide a first approach towards defining a process for comparing the profiles. This approach will be evaluated in the comparison module through mechanisms of matching of profiles, using as input the patterns identified in the texts using our axioms and linguistic patterns.

The application of low-level linguistic patterns on the corpus of profiles, gives us the opportunity to experiment with machine learning algorithms under a semi-supervised approach, especially in the step of updating of competencies (learning skills, knowledge and competencies).

In order to improve this research, future works will be directed towards developing of the modules of extraction and comparison of our architecture, the formal definition of axioms and linguistic patterns using descriptive logic, and the implementation of machine learning algorithms for the phase of competency upgrading.

Acknowledgments

Dr Aguilar has been partially supported by the Prometeo Project of the Ministry of Higher Education, Science, Technology and Innovation of the Republic of Ecuador.

References

[Bizer et al., 2005] Christian Bizer, Ralf Heese, Malgorzata Mochol, Radoslaw Oldakowski, Robert Tolksdorf, and Rainer Eckstein. The impact of semantic web technologies on job recruitment processes. In Wirtschaftsinfor-matik 2005, pages 1367–1381. Springer, 2005.

[Bourque et al., 2003] Pierre Bourque, Luigi Buglione, Alain Abran, and Alain April. Bloom's taxonomy levels for three software engineer profiles. In Software Technology and Engineering Practice, 2003. Eleventh Annual International Workshop on, pages 123–129. IEEE, 2003.

[Capdevila, 2011] Javier P´erez Capdevila. Desde los mapas de conocimientos hacia los mapas de competencias. Avanzada Cient´ıfica, 13(1):67–75, 2011.

[Dorn and Pichlmair, 2007] Jrgen Dorn and Markus Pichlmair. A competence management system for universities. In ECIS, pages 759–770, 2007.

[Dorn et al., 2007] J¨urgen Dorn, Tabbasum Naz, and Markus Pichlmair. Ontology development for human resource management. In Proceedings of the 4th International Conference on Knowledge Managements, pages 109–120. Citeseer, 2007.

[Draganidis and Mentzas, 2006] Fotis Draganidis and Gregoris Mentzas. Competency based management: a review of systems and approaches. Information Management & Computer Security, 14(1):51–64, 2006.

[Fazel-Zarandi and Fox, 2009] Maryam Fazel-Zarandi and Mark S Fox. Semantic matchmaking for job recruitment: an ontology-based hybrid approach. In Proceedings of the 8th International Semantic Web Conference, 2009.

[Fazel-Zarandi, 2013] Maryam Fazel-Zarandi. Representing and Reasoning about Skills and Competencies over Time. PhD thesis, University of Toronto, 2013.

[Huang, 2008] Anna Huang. Similarity measures for text document clustering. Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. 2008.

[Lindgren et al., 2004] Rikard Lindgren, Ola Henfridsson, and Ulrike Schultze. Design principles for competence management systems: a synthesis of an action research study. Mis Quarterly, pages 435–472, 2004.

[Malzahn et al., 2013] Nils Malzahn, Sabrina Ziebarth, and H Ulrich Hoppe. Semi-automatic creation and exploitation of competence ontologies for trend aware profiling, matching and planning. Knowledge Management & E-Learning: An International Journal (KM&EL), 5(1):84–103, 2013.

[Manning and Schütze, 1999] Christopher D Manning and Hinrich Schütze. Foundations of statistical natural language processing. MIT press, 1999.

[Mena et al., 2007] Elena Mena, María Pareja, Miguel-Angel Sicilia, and Elena García-Barriocanal. Deriving competencies from the swebok ontology. In 2nd International Workshop on Ontology, Conceptualization and Epistemology for Software and System Engineering, 2007.

[Müller-Riedlhuber, 2009] Heidemarie Müller-Riedlhuber. The european dictionary of skills and competencies (disco): an example of usage scenarios for ontologies. In I-SEMANTICS, pages 467–479, 2009.

[Paquette et al., 2012] Gilbert Paquette, Delia Rogozan, and Olga Marino. Competency comparison relations for recommendation in technology enhanced learning scenarios. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012), volume 896, pages 23–34. Citeseer, 2012.

[Paquette, 2007] Gilbert Paquette. An ontology and a software framework for competency modeling and management. Educational Technology & Society, 10(3):1–21, 2007.

[Poblete, 2006] Manuel Poblete. Las competencias, instrumento para un cambio de paradigma. 2006.[Rieu et al., 2005] Christine Rieu, Cécile Million Rousseau, and C Roche. Gestion des compétences: un modèle opérationnel à base dontologie. In Colloque Du E-Management à la E-RH, 2005.

[Rudzajs and Kirikova, 2011] Peteris Rudzajs and Marite Kirikova. Mediated competency comparison between job descriptions and university courses. Scientific Journal of Riga Technical University. Computer Sciences, 44(1):48– 56, 2011.

[Schmidt and Kunzmann, 2006] Andreas Schmidt and Christine Kunzmann. Towards a human resource development ontology for combining competence management and technology-enhanced workplace learning. In On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, pages 1078–1087. Springer, 2006.

[Sicilia et al., 2005] Miguel- Angel Sicilia, Juan J Cuadrado, and Daniel Rodríguez. Ontologies of software artifacts and activities: Resource annotation and application to learning technologies. In SEKE, pages 145–150, 2005.

[Stevens, 2013] Gregory W Stevens. A critical review of the science and practice of competency modeling. Human Resource Development Review, 12(1):86–107, 2013.

[Yahiaoui et al., 2006] Leila Yahiaoui, Zizette Boufaïda, and Yannick Prié. Semantic annotation of documents applied to e-recruitment. In SWAP, 2006.

Comparative Study of Verse Similarity for Multi-lingual Representations of the Qur'an

A. Basharat[1], D. Yazdansepas[1], and K. Rasheed[1]
[1]Department of Comp. Sci, University of Georgia, Athens, GA, USA

Abstract—*Text similarity is a subject that has received great attention in recent years. However, the application of text similarity tools to Semitic languages such as Arabic faces unique challenges. Moreover, the increasing number of texts being made available online, not only in native languages but also in translation, adds further challenge to identifying similar portions of texts across different documents. In this paper, we explore the problem of text similarity in the context of multi-lingual representations of the Qur'an. Particularly, we use Arabic and English datasets of the Qur'an for comparative study and analysis of several similarity measures applied across different representations of the verses in the Qur'an. We provide useful insights into the impact of using different similarity measures applied to different features across different representations and linguistic characteristics of similar text.*

Keywords: similarity, document similarity, Qur'an, Arabic, cosine, Jaccard

1. Introduction

The Qur'an, considered as a concise data set, consists of less than 80,000 words, sequenced in 114 chapters (Surahs) and 6,236 verses (Ayahs) [1]. The original data format was spoken Classical Arabic. We treat the problem of computing similarity between the verses of the Qur'an as a special case of computing document similarity, a widely studied subject in literature. Document similarity measures are often utilized for the purpose of automatic text classifcation, and clustering [2] [3]. We treat a verse somewhat similar to a document. However, some verses may be as short as a single or few words long. The longest verse in the Qur'an does not span more than a single page in the standard manuscript writing. Researchers recognize that the determination of a pair of documents being similar or different is not always clear and is often context dependent [3]. The problem at hand bears resemblance to the context of short text classification and clustering [4] [5].

We are not only interested in analyzing verse similarity using the original Arabic script, but also take considerable interest in undertaking this similarity study within and across different languages. Qur'an is one of the most widely translated texts, translated into numerous languages. A recent open linked dataset called 'Semantic Quran' has been published [6], with more than 48 known translations obtained from Tanzil[1]. Research in other domains has revealed interesting findings when cross-language similarity studies were undertaken [7].

The problem of computing the similarity between the verses has not been widely studied, despite the fact, that the Qur'anic text has attracted much attention in recent years from the computational research, artificial intelligence and natural language processing researchers in particular. There have been studies conducted using various corpuses in Arabic, some of which are [8], [9]. Sharaf et al.[10] are among the pioneers to have conducted some similarity and relatedness studies on the the Qur'anic text. The QurSim corpus is a corpus that is marked with relatedness information judged by a human domain expert. The authors highlight the challenges the variation in the degree of relatedness between texts. While there are instances where lexical matching is evident between the terms, the authors believe that the majority of the related pairs require a deeper and more semantic analysis and domain specific world knowledge in order to relate the two texts in the pair. For a more detailed discussion on the relatedness and the extent of its broadness, we refer the reader to the original work of Sharaf and his collegues.

Our objective here is to use the Qur'an's text to compare and contrast the different classes of similarity that may result using different representations of the text, and also using different similarity measures. We believe that a benchmarking study towards standardizing similarity measures for the Qur'anic text would pave the way towards the vision of achieving the ongoing efforts of standardizing the Qur'anic knowledge map as described by Atwell and collegues [11], [1]. We intend to use this study as the basis of developing semantic networks of similarity and relatedness between not only verses of the Qur'an but also extending to other contextually relevant texts, which are considered indispensable when it comes to developing a comprehensive and coherent understanding of the Qur'anic verses.

In this paper, we evaluate a variety of document similarity measures, using multi-lingual, heterogenous representations of the Qur'an. In order to do this, we develop a similarity computation framework for the verses in the Qur'an (Section 2). We experiment with four different dataset representations, four similarity measures and three different feature repre-

[1]http://tanzil.net/

sentations and provide comparative insights into the results obtained (Section 3 and 4). We also shed light towards how we plan to extend this study further (Section 5).

2. Verse-to-Verse Similarity Computation Framework for the Qur'an

In this section, we present the design of our developed framework, which is meant to facilitate the process of similarity evaluation between the verses of the Qur'an, from the available datasets. The framework we have developed is generic enough to be readily used with other texts.

2.1 Datasets and their Characteristics

We have created a custom database for the experimentation which consists of the Qur'anic text in its original Arabic script in two forms: Firstly, with diacritics, i.e. case markings, and secondly, without diacritics, i.e. plain, simple and clean arabic script. In addition, the database also contains translations in several languages. We used the translation of the Qur'an to the English language (translation by Yusuf Ali) for this study. The last type of data representation we used was a dataset generated for the Arabic roots (stems). A summary of the datasets used is shown in Table 1.

Table 1: Dataset representations and samples

Abbreviation	Description	Example
Q-DIAC	Arabic text with diacritics (or case markings)	وَ لَمْ أَكُن بِدُعَائِكَ رَبِ شَقِيًّا
Q-NODIAC	Arabic text without diacritics	ولم أكن بدعائك رب شقيا
Q-ROOTS	Arabic roots dataset	كون، دعو، ربب ، شقو
Q-ENG	English Text	"...and never have I been in my supplication to You, my Lord, unhappy."

2.2 Preprocessing of Verses

For this purpose, we used the raw data for each dataset without any linguistic pre-processing, except tokenization. We created a workflow mechanism such that we specify an input configuration that encodes the input data representation, the feature selection method to be used, and the similarity method to be used for an experiment. The Verse-Set Generator as shown in Figure 1, uses the Query Processor to generate the dataset for experimentation from the database, and prepares the appropriate input representation to be used in later stages.

2.3 Feature Selection and Verse Representation

In order to adequately compare and contrast the different representations and the multi-lingual texts, we adopt a vector space model as means of verse representation for our experiments. For preliminary analysis we did not use any specific pre-processing techniques such as stemming. We relied on the traditional term-vector approach, which is the conventional approach adopted in most text mining applications.

We consider $V = \{v_1, v_2, v_3 ... v_n\}$ to denote the set of Verses in the Qur'an, and $\{T = t_1, t_2, ... t_m\}$ as the set of distinct terms occurring in V. We represent each verse, v, in the Qur'an data set as an m-dimensional vector $\bar{v} = \{a_1, a_2, a_3 .... a_m\}$, where a_i is the weight of the ith feature in the vector $\bar{v}$.

The framework we developed primarily operates on a term-verse matrix ($tv - matrix$), which is generated using the Verse Vector Set generator. It is a compilation of all the verse vectors into one matrix where the columns represent the terms (the vocabulary of the dataset) and the rows represent the verses. First the vocabulary is compiled from the verses, and the frequency of terms across all verses is recorded. Then the $tv - matrix$ is developed based on the verse order, in which the elements of the matrix are calculated according to one of the three term weighting methods namely: Boolean(B), Frequency(F) or TF-IDF(TF).

In traditional document similarity studies the $tf - idf$ method of term weighting is most recommended. However, Qur'an is considered as a text with several unique characteristics. Arabic linguists consider it as a profound piece of text, where each letter and word is relevant. Therefore we choose to experiment with different term-weighting measures to analyze the impact on the similarity computation. The feature selector is responsible for applying the feature selection method specified in the input configuration. The verse modeler then applies this selection to each verse before passing it to the verse vector set generator, which generates the $tv - matrix$.

We start with the most simple measure i.e. the Boolean. Using this measure the weight of the term t_i, would have the value 1 if the term is present in the verse and 0 otherwise. Equation (1) shows the term weighting using the term-frequency.

$$a_i = tf(v, t_i) \tag{1}$$

Equation (2) shows the term weighting using the term-frequency, inverse document frequency approach. In this scheme, N_v is the total number of verses in the dataset, vf_j is the verse frequency, and tf_{ij} is the number of occurences of the feature a_i in the verse vf_j.

$$a_i = tf_{ij} \times \log(N_v / vf_j) \tag{2}$$

2.4 Similarity Computation

With the verses represented as vectors, we measure the degree of similarity between the two verses as the correlation between their corresponding vectors using the Similarity computation module. The measures described below reflect the degree of closeness or separation of the target verses. Choosing an appropriate measure of similarity is crucial.

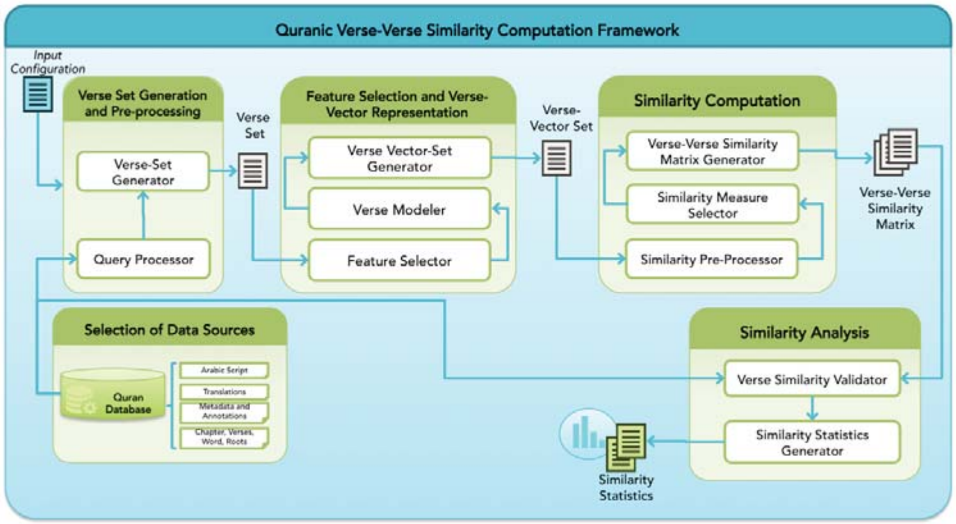


Fig. 1: Qur'anic Verse Similarity Computation framework

We believe that understanding the effects of different similarity measures when applied to the Qur'an to be of great importance in helping to choose the best one. We adopt four of the measures described in [12] and [3] and map them to the problem of computing similarity for the Qur'anic verse vectors. The output of this stage is a Verse-Verse Similarity matrix, which provides a similarity measure on a scale of 0 to 1 for each verse pair in the Qur'an.

2.4.1 Euclidean Distance

Given the two verses v_a and v_b, represented by their term vectors $\bar{v}_a$ and $\bar{v}_b$ respectively, the Euclidean distance of the two verses is defined in Equation (3).

$$D_E(\bar{v}_a, \bar{v}_b) = (\sum_{t=1}^{m} |w_{t,a} - w_{t,b})|^2)^{1/2} \tag{3}$$

where the term set is $T = \{t_1, t_2 ... t_m\}$. As mentioned earlier, we use different weighting measures. Therefore, the $w_{t,a}$ may be $tfidf(d_a, t)$ or $tf(d_a, t)$ or $tb(d_a, t)$.

2.4.2 Cosine Similarity

Cosine similarity is one of the most popular similarity measure applied to text documents. Some notable studies that report the use of this distance measure include [3] [12]. Given the two verses v_a and v_b, represented by their term vectors $\bar{v}_a$ and $\bar{v}_b$ respectively, their cosine similarity is given by Equation (4)

$$S_C(\bar{v}_a, \bar{v}_b) = \frac{\bar{v}_a \cdot \bar{v}_b}{|\bar{v}_a| \times |\bar{v}_b|} \tag{4}$$

where v_a and v_b are m-dimensional vectors over the termset $T = \{t_1, t_2 ... t_m\}$. Each dimension represents a term with its weight in the verse, which is non-negative. Therefore, the cosine similarity is non-negatively bounded between 0 and 1.

2.4.3 Jaccard Similarity

The Jaccard coefficient measures similarity as the intersection divided by the union of the entities. For the verses, the Jaccard coefficient computes the ratio between the dot product and the sum of the squared norms minus the dot product of the given verse vectors. The definition is given in Equation (5).

$$S_J(\bar{v}_a, \bar{v}_b) = \frac{\bar{v}_a \cdot \bar{v}_b}{|\bar{v}_a|^2 + |\bar{v}_b|^2 - \bar{v}_a \cdot \bar{v}_b} \tag{5}$$

The Jaccard coefficient is a similarity measure and ranges between 0 and 1. It is 1 when the $v_a = v_b$ and 0 when v_a and v_b are disjoint, where 1 means the verses are the same and 0 means the verses are completely different.

2.4.4 Pearson Correlation Coefficient

Pearson's correlation coefficient is another measure of the extent to which two vectors are related. There are different forms of this coefficient. Given the term set $T = \{t_1, t_2 ... t_m\}$, a commonly used form is given in Equation (6).

$$S_P(\bar{v}_a, \bar{v}_b) = \frac{m \sum_{t=1}^{m} w_{t,a} \times w_{t,b} - TF_a \times TF_b}{\sqrt{(m \sum_{t=1}^{m} w_{t,a}^2 - TF_a^2)(m \sum_{t=1}^{m} w_{t,b}^2 - TF_b^2)}} \tag{6}$$

where $TF_a = \sum_{t=1}^{m} w_{t,a}$ and $TF_b = \sum_{t=1}^{m} w_{t,b}$.

This is also a similarity measure. When $v_a = v_b$, the value will be 1.

2.5 Similarity Analysis

We devised three similarity classes for the purpose of analysis as shown in Table 2. We automate some validation measures for determining the results. We use the Verse-Verse Similarity matrix for computing the statistics on similarity reported in Section 3.

Table 2: Similarity Classes for Analysis

Class	Description	Similarity Range
Identical	Identical Verses	Verse Pairs with Similarity Value 1
High	Almost Identical, Near Identical	Verse Pairs with Similarity Value > 0.9
Medium-High	With identifiable similarity	Verse Pairs with Similarity Value between 0.75 and 0.9
Low	Minimal semantic similarity	Verse Pairs with Similarity less than 0.75

3. Experimentation and Results

3.1 Evaluation Measures

We devised the accuracy measure, for the sake of quantification, for the *Identical* similarity class using the evaluation measures shown in equations (7 - 9).

$$P(Precision) = \frac{TP}{TP + FP} \tag{7}$$

$$R(Recall) = \frac{TP}{TP + FN} \tag{8}$$

$$F1(F - Measure) = \frac{2 \times P \times R}{P + R} \tag{9}$$

For the sake of analysis, we limit our quantification to those verse pairs with similarity value 1. Therefore, we treat those verse pairs with similarity value 1 as true positives (TP). A TN is therefore a verse pair with similarity value that does not equal one. The reason for limiting this analysis is that establishing the precise similarity between two verses has not been done in a standardized manner and there are no existing benchmark datasets available. The only dataset available QurSim [10] employs a different approach, i.e. it classifies verse pairs according to their relatedness measure, using human subjective judgement, based on the verse pairs compiled from a famous source of Qur'anic Exegesis by Ibn Kathir. For our study, we rely on pure measures of distance, as described earlier in this section. Therefore, establishing a similarity benchmark is not possible against which evaluation may be carried out. The only pairs of verses we are able to consider as ground truth are the identical verse pairs.

3.2 Experiments

The experiments are applied to every combination of using each of the four dataset representations, each of the similarity measures and each of the term-weighting methods. We therefore had $3 \times 4 \times 4 = 48$ experiments in total. An abbreviation scheme is used to denote the experiments. A sample experiment scheme indicates the dataset used, term-weighting applied and the distance measure. E.g. Q-DIAC-C-B indicates, Boolean(B) term weighting is used, and Cosine(C) similarity measure is used for the Q-DIAC dataset, which is the Qur'an representation in Arabic with diacritics (case markings). The experimental configurations are summarized in Table 3.

3.3 Results and Analysis for Identical Verses

The results for the 48 experiments are shown in Table 3. We chose the F1 measure in order to provide a general measure of performance of the similarity configurations. We highlight the performance of each dataset below and then provide an overall comparison:

3.3.1 Q-NODIAC Representation

The Q-NODIAC representation provides the most accurate results when dealing with the Identical similarity class, with a perfect recall and almost perfect precision, and therefore the F1 measure being close to 1. The best result is obtained using C-F, E-TFIDF, E-F, and P-TFIDF combinations (indicated by 1 pair of verses classified as False Positive). All other combinations also perform almost as well. The difference is indicated by the number of False Positives (FP)s, in this case 2 pairs of verses are classified as FPs. The two FPs are the verse pairs [23:83, 27:68] and [22:62, 31:30][2]. Looking into the reasons for misclassification, we notice that the former pair includes a word that occurs twice in the verse 23:83 but occurs only once in 27:68. This is correctly captured using some of the combinations but missed with others such as the ones involving Boolean term weighting measure. The later verse pair, is a unique case, where the words are similar in both verses, however, the order is different. This is a unique case indeed, and perhaps, the only one of its kind in the Qur'an.

3.3.2 Q-DIAC Representation

The Q-DIAC performs considerably close to Q-NODIAC in terms of precision, however, the recall suffers due to the strikingly large number of verse pairs classified as False Negatives (FN)s. The verse pairs classified as FPs are the same as those in Q-NODIAC, apart from one additional verse pair i.e. [30:52, 27:81]. The reason for this verse pair being mis-classified in the difference in orthography of the words, which are not captured in the original dataset. By manual inspection we analyzed the verse pairs in the

[2]23:83 indicates Chapter 23, Verse 83 in the Qur'an

Table 3: Precision and Recall Measures

		Cosine (C)			Euclidean (E)			Jaccard (J)			Pearson (P)		
		B	**TFIDF**	**F**	**B**	**TFIDF**	**F**	**B**	**TFIDF**	**F**	**B**	**TFIDF**	**F**
Q-NODIAC	**TP**	775	775	775	775	775	775	775	775	775	775	775	775
	FP	2	2	1	2	1	1	2	2	2	2	1	2
	FN	0	0	0	0	0	0	0	0	0	0	0	0
	P	0.997	0.997	0.999	0.997	0.999	0.999	0.997	0.997	0.997	0.997	0.999	0.997
	R	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	F	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999
Q-DIAC	**TP**	737	737	737	737	737	737	737	737	737	737	737	737
	FP	3	3	2	3	2	2	3	2	2	3	2	2
	FN	38	38	38	38	38	38	38	38	38	38	38	38
	P	0.996	0.996	0.997	0.996	0.997	0.997	0.996	0.997	0.997	0.996	0.997	0.997
	R	0.951	0.951	0.951	0.951	0.951	0.951	0.951	0.951	0.951	0.951	0.951	0.951
	F	0.973	0.973	0.974	0.973	0.974	0.974	0.973	0.974	0.974	0.973	0.974	0.974
Q-ENG	**TP**	661	661	660	661	660	660	661	661	660	661	660	661
	FP	12	12	10	12	10	10	12	12	10	12	10	12
	FN	114	114	115	114	115	115	114	114	115	114	115	114
	P	0.982	0.982	0.985	0.982	0.985	0.985	0.982	0.982	0.985	0.982	0.985	0.982
	R	0.853	0.853	0.852	0.853	0.852	0.852	0.853	0.853	0.852	0.853	0.852	0.853
	F	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913
Q-ROOTS	**TP**	738	738	738	775	775	775	738	738	738	738	738	738
	FP	106	98	100	259	249	249	106	96	96	106	98	100
	FN	37	37	37	0	0	0	37	37	37	37	37	37
	P	0.874	0.883	0.881	0.750	0.757	0.757	0.874	0.885	0.885	0.874	0.883	0.881
	R	0.952	0.952	0.952	1.000	1.000	1.000	0.952	0.952	0.952	0.952	0.952	0.952
	F	0.912	**0.916**	0.915	0.857	**0.862**	**0.862**	0.912	**0.917**	**0.917**	0.912	**0.916**	0.915

FN category, and we discovered that although the verses are actually similar, the orthography of the words, which is preserved in the diacritics as per the original manuscript causes them to appear different. For these experiments we preserved this in the original dataset. However, some of these orthographical cases may be removed without loss in any phonetic or linguistic characters of the word. Doing so, we expect that the Q-DIAC results would follow closely the results of Q-NODIAC dataset.

3.3.3 Q-ENG Representation

The Q-ENG representation, amongst all those approaches which use the raw Qur'anic representation, performs with a relatively low recall of around 85%. There is very little difference in either of the methods as indicated by the F1 measure which comes out to be the same for all. This representation reports the greatest number of False Negatives (FN) as indicated in the Table 3. This is a significant indicator especially when it comes to the identical verses. A FN implies that the verse pair which was expected to be classified as identical fails to do so. This has implications towards the number of translations available. It is clear that the identical verses in different occurrences within the Qur'an are being translated differently. To verify this we took a few sample cases and verified this implication. وَلَقَدْ يَسَّرْنَا الْقُرْآنَ لِلذِّكْرِ فَهَلْ مِن مُّدَّكِرٍ This verse is repeated in 54:17 and 54:22. The translation differs slightly (by one word) in the dataset we used for our experimentation.

The precision in the translation is a subjective issue. In the verses 54:17 and 54:22 the context may cause the first initial to mean as 'and' or 'but'. However, in another translation, the two verses are translated the same.

Analyzing the FPs, we discovered that in some cases, the English translation is the same but the Arabic terms used are different. This is significant when analyzing similar verses. Arabic is a rich morphological language. In particular the language of the Qur'an is considered to be precise and slightest variation or alteration in the arrangement or morphological manifestation of the word implies something significant, which often the translation fails to capture. E.g. in Table 4, the two verses are compared using the original text. The arabic text clearly distinguishes سحار and ساحر. The two words actually provide a different connotation, which is not captured by the English translation. These cases confirm this aspect of the translations that are available for the Qur'an. This can be used as a good measure for assessing the quality of translations. Another case is that of verse 37:80 [إِنَّا كَذَٰلِكَ نَجْزِي الْمُحْسِنِينَ] and verse 37:110 [كَذَٰلِكَ نَجْزِي الْمُحْسِنِينَ]. The word إِنَّا is well known in Arabic for a particle of extreme emphasis; this word is present in the former verse, however it is absent in the later. The english translation does not capture the difference in emphasis present in the arabic speech, as indicated by the highlighted word, in the context of these two verses.

Although, for this study we only used one translation of the Qur'an, available in English, our framework can prove useful in analyzing the quality of translations. An examination of those cases in other translations revealed that there are translations that distinguish clearly between near identical verses. This indeed is a reflection of the quality of the deliberation and effort that has gone into the translation process and thus may be measured to some extent.

Table 4: Analysis of false positive case (Q-ENG)

Verse	Verse Text(English)	Verse Text (Arabic)
7:112	And bring up to thee all (our) sorcerers well versed.	يأتوك بكل سحار عليم
26:37	And bring up to thee all (our) sorcerers well versed.	يأتوك بكل ساحر عليم

3.3.4 Q-ROOTS Representation

For the Q-ROOTS dataset, while recall is high, precision is quite low compared to the other datasets. In terms of the overall F1 measure, the Q-ROOTs and the Q-ENG datasets perform close to one another. However, the recall is much higher for the Q-ROOTs dataset compared to the Q-ENG dataset. In terms of the similarity measures, the pattern is not as consistent as with the other datasets. The C, J and P similarity measures perform closely. However, the Euclidean measure has a much lower F1 measure in comparison to the others. Interestingly, the recall for Euclidean comes out as 1. However, the precision is very low compared to the others. An investigation into this revealed some interesting results. An analysis of the FN cases showed that all 37 verse pairs are actually special verses which occur at the beginning of some chapters, and are special compositions of letters which are said to have no known roots or meanings. The Q-ROOTs dataset thus does not assign these terms any weight. The Euclidean measure, however, counts them as equivalent, even though the verse vectors are empty. Nevertheless, due to the raw nature of that similarity measure it computes them as equal. This explains the rather high number of FPs using the Euclidean measure, as compared to the other three. We manually inspected the verse pairs in this category to verify this. If we discount those verses, the measures would report equivalent. A future improvement to this measure could disregard those verses, or make a special case for the other similarity measures. If this is the case, all other methods would also report a perfect recall. However, for the initial experimentation, we preserved this result, because it indeed highlights the unique nature of those verses. Another interesting observation is that the Q-ROOTs returns the highest number of verse pairs with similarity value 1. This is indicative of high semantic similarity.

3.3.5 Overall Comparative Analysis

In general, we can conclude that using the Q-NODIAC representation provides the most accurate results when dealing with the high similarity class. In terms of the similarity methods used, the P-TFIDF configuration, i.e. the Pearson method, with TFIDF as the weighting measure and the C-TFIDF, the cosine method, with TFIDF as the weighting method perform close and comparable. The analysis of the Similarity class where the verse pairs are identical is not sufficient to provide a reasonable estimate of the precision of the method that would perform the best overall. We therefore also look into some of the verse pairs that fall in the next similarity range to give us a reasonable estimate.

3.4 Results and Analysis for Verses with High Similarity

While the similarity method performance results are similar for the identical cases, this is not the case for the similarity ranges other than identical. This is shown in Figure 2. An investigation into the verse pairs retrieved with similarity values falling in the range of more than 0.9 and less than 1 reveals the numbers shown in Figure 2.

The Pearson method returns the most number of verses in that range. This is an important finding, as the results are meaningful. Verses with reasonably high similarity, which are expected to fall in this range are not falling in that range when using Cosine, Jaccard or Euclidean. We manually inspected some verses, which bear high semantic similarity. The Q-ROOTs dataset returned a large number of FPs, some of which are actually verses with a difference of only a single literal. When we looked into the similarity values of such pairs, and compared these values across the different experimental configurations, we observed that for a highly similar verse pair, the Pearson method returns the highest similarity values, whereas the similarity values returned by other methods are much lower. This pattern is reflective in Figure 2 which displays the number of verse pairs in the similarity ranges from 0.75 to 1. As an example we took the verse pair [94:5, 94:6], which differs by a single literal. For both Q-DIAC and Q-NODIAC the values returned by Cosine, Jaccard and Euclidean are lower than 0.75, whereas for the Pearson method, the highest values are above 0.95. Therefore it may be concluded that Pearson performs best. As for the weighting measures the TFIDF measure with pearson returns the highest value. However, the performance of the term weighting measure may not be generalized from these instances alone and would require further investigation.

4. Discussion and Analysis

Our study, as outlined and described in this paper, set out to quantify similarity between text segments, which is considered an important step in authorship attribution, corpus comparison, information retrieval, text mining and other fields. We attempted to provide an essential foundation by providing a means to choose an effective text similarity measuring scheme especially across cross-lingual texts of the same origin. We concluded that for the identical verses, the best results are obtained using the Q-NODIAC dataset, and all similarity measures perform comparably well. The Q-ROOTs dataset shows relatively low precision.

Challenges in measuring Text Similarity vs. Text Relatedness: It is worth mentioning that the notion of both text similarity and text relatedness are of strong importance when

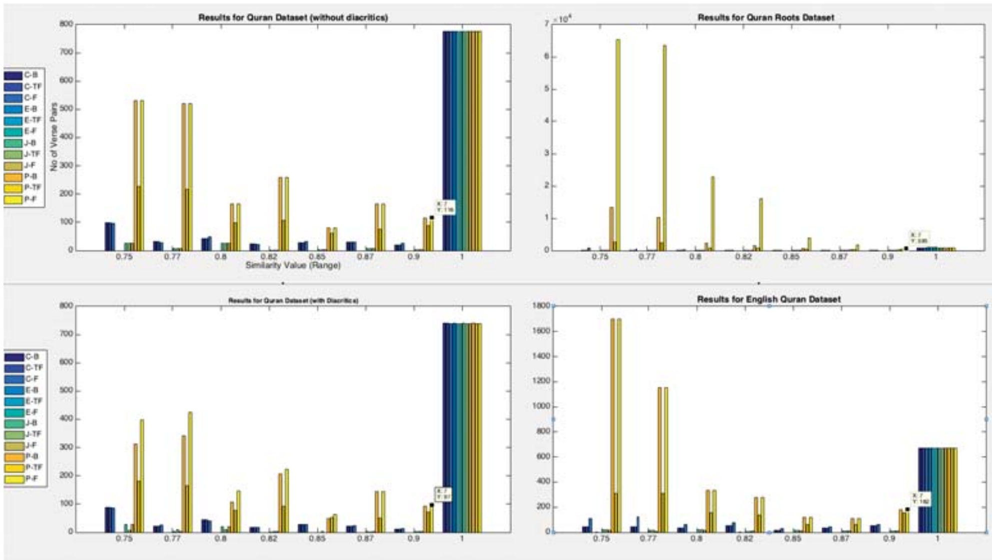


Fig. 2: Results for all representations (Similarity Range vs. No of Verse Pairs)

studied in the context of the Qur'anic scripture.The vector space model (VSM) or the bag of words approach imposes certain limitations and has several implications. The order of words is ignored which may be of value. The issue of synonymy is ignored. In addition, the issue of polysemy is not addressed because the VSM model only considers word forms. These limitations have been highlighted in previous research such as [8].

However, we feel that, the Qur'an is a text that deserves considerable attention when it comes to similarity at different levels. The results from this study will provide sound grounds to build upon and enhance the proposed similarity measures and methods with stem based methods to provide for more enhanced similarity classes. A future comparative study is planned to this end.

5. Conclusion

In this paper, we investigated the application of various similarity measures to heterogenous representations of the Qur'an. The results focused on an analysis of the identical verses, delineating upon some interesting findings. As a natural next step, we aim to obtain some highly reliable similarity measures for non-identical verses, especially in the higher similarity ranges in order to establish ground truth for verse similarity in the Qur'an. We believe our study will prove to be a useful step in this direction. We aim to involve users and experts through established crowdsourcing approaches. We hope to devise an improved, hybrid similarity measure in the future for establishing a more precise estimate of semantic similarity. We also plan to extend our work to include other translations of the Qur'an and carry out a comparative analysis, as our study revealed some interesting findings that can help assess the quality of translations, and help standardize the existing translations.

References

[1] E. Atwell, N. Habash, B. Louw, B. Abu Shawar, T. McEnery, W. Zaghouani, and M. El-Haj, "Understanding the quran: A new grand challenge for computer science and artificial intelligence," *ACM-BCS Visions of Computer Science 2010*, 2010.

[2] M. Al Qady and A. Kandil, "Automatic clustering of construction project documents based on textual similarity," *Automation in Construction*, vol. 42, pp. 36–49, 2014.

[3] A. Huang, "Similarity measures for text document clustering," in *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, 2008, pp. 49–56.

[4] G. Song, Y. Ye, X. Du, X. Huang, and S. Bie, "Short text classification: A survey," *Journal of Multimedia*, vol. 9, no. 5, pp. 635–643, 2014.

[5] D.-T. Vo and C.-Y. Ock, "Learning to classify short text from scientific documents using topic models with various types of knowledge," *Expert Systems with Applications*, vol. 42, no. 3, pp. 1684–1698, 2015.

[6] M. A. Sherif and A.-C. Ngonga Ngomo, "Semantic quran - a multilingual resource for natural-language processing," *Semantic Web*, 2009.

[7] R. S. Forsyth and S. Sharoff, "Document dissimilarity within and across languages: A benchmarking study," *Literary and Linguistic Computing*, vol. 29, no. 1, pp. 6–22, 2014.

[8] H. Froud, A. Lachkar, and S. A. Ouatik, "A comparative study of root-based and stem-based approaches for measuring the similarity between arabic words for arabic text mining applications," *arXiv preprint arXiv:1212.3634*, 2012.

[9] M. S. Khorsheed and A. O. Al-Thubaity, "Comparative evaluation of text classification techniques using a large diverse arabic dataset," *Language resources and evaluation*, vol. 47, no. 2, pp. 513–538, 2013.

[10] A.-B. M. Sharaf and E. Atwell, "Qursim: A corpus for evaluation of relatedness in short texts." in *LREC*, 2012, pp. 2295–2302.

[11] E. Atwell, C. Brierley, K. Dukes, M. Sawalha, and A.-B. Sharaf, "An artificial intelligence approach to arabic and islamic content on the internet," *Proceedings of NITS 3rd National Information Technology Symposium*, 2011.

[12] A. Strehl, J. Ghosh, and R. Mooney, "Impact of similarity measures on web-page clustering," in *Workshop on Artificial Intelligence for Web Search (AAAI 2000)*, 2000, pp. 58–64.

KNN based Word Categorization considering Feature Similarity

Taeho Jo
Department of Computer and Information Engineering, Inha University, Incheon, South Korea

Abstract—*In this research, we propose that the K Nearest Neighbor should be used for categorizing words semantically, considering the feature similarities. In the reality, the dependencies and relations among features are available; texts as features for encoding words into numerical vectors tend to have their similarities with others. In this research, we define the similarity measure considering both feature values and features and use it for modifying the K Nearest Neighbor as the approach to the word categorization. As the benefits from this research, we obtain the potential possibility of more compact representations of words and the improvement of their discriminations among even sparse vectors. Hence, the goal of this research is to implement the word categorization systems with the benefits.*

Keywords: Word Categorization, Feature Similarity

1. Introduction

The word categorization refers to the process of classifying the words into some or one of the predefined categories. As its preliminary task, we must define a list of topics as the classes, and allocates sample words to each topic. The sample words are encoded into numerical vectors as the preprocessing step, and the classification capacity is built by learning the labeled words. Afterward, the novice words are also encoded so, and they are classified into one or some of the predefined topics. The scope of this research is restricted to the classification of words by their topics; the POS tagging as a kind of word classification is set out of this research.

Let us consider some challenges with which this research tries to tackle. Previously, the dependencies among features were discovered, but it requires very complicated analysis for considering them [1]. The assumption of independencies among features for the simplicity causes requirement of many features for the robustness. Because each feature has very little coverage, we cannot avoid the sparse distribution where zero values are dominant with more than 95% in each numerical vector[3]. Therefore, this research is intended to solve the problems by considering the feature similarities as well as the feature value similarities.

Let us mention what we propose in this research as its idea. In this research, we consider the both feature similarity and feature value similarity for computing the similarity between numerical vectors. The KNN (K Nearest Neighbor) is modified into the version which accommodates the both similarity measures. We apply the modified version to implementing the word categorization system. Therefore, the goal of this research is to improve the word categorization performance, using smaller number of features by solving the above problems.

Let us mention what we expect from this research as the benefits. In the proposed research, we consider semantic similarity among features as well as feature values. By considering both, the discriminations among even sparse vectors are improved. This research provides potentially the way of reducing the dimension of numerical vectors by doing so. Therefore, this research pursues the benefits for implementing the text categorization systems.

This article is organized into the four sections. In Section 2, we survey the relevant previous works. In Section 3, we describe in detail what we propose in this research. In Section 4, we mention the remaining tasks for doing the further research.

2. Previous Works

Let us survey the previous cases of encoding texts into structured forms for using the machine learning algorithms to text mining tasks. The three main problems, huge dimensionality, sparse distribution, and poor transparency, have existed inherently in encoding them into numerical vectors. In previous works, various schemes of preprocessing texts have been proposed, in order to solve the problems. In this survey, we focus on the process of encoding texts into alternative structured forms to numerical vectors. In other words, this section is intended to explore previous works on solutions to the problems.

Let us mention the popularity of encoding texts into numerical vectors, and the proposal and the application of string kernels as the solution to the above problems. In 2002, Sebastiani presented the numerical vectors are the standard representations of texts in applying the machine learning algorithms to the text classifications [4]. In 2002, Lodhi et al. proposed the string kernel as a kernel function of raw texts in using the SVM (Support Vector Machine) to the text classification [5]. In 2004, Lesile et al. used the version of SVM which proposed by Lodhi et al. to the protein classification [6]. In 2004, Kate and Mooney used also the SVM version for classifying sentences by their meanings [7].

It was proposed that texts are encoded into tables instead of numerical vectors, as the solutions to the above problems. In 2008, Jo and Cho proposed the table matching algorithm as the approach to text classification [8]. In 2008, Jo applied also his proposed approach to the text clustering, as well as the text categorization [12]. In 2011, Jo described as the technique of automatic text classification in his patent document [10]. In 2015, Jo improved the table matching algorithm into its more stable version [11].

Previously, it was proposed that texts should be encoded into string vectors as other structured forms. In 2008, Jo modified the k means algorithm into the version which processes string vectors as the approach to the text clustering[12]. In 2010, Jo modified the two supervised learning algorithms, the KNN and the SVM, into the version as the improved approaches to the text classification [13]. In 2010, Jo proposed the unsupervised neural networks, called Neural Text Self Organizer, which receives the string vector as its input data [14]. In 2010, Jo applied the supervised neural networks, called Neural Text Categorizer, which gets a string vector as its input, as the approach to the text classification [15].

The above previous works proposed the string kernel as the kernel function of raw texts in the SVM, and tables and string vectors as representations of texts, in order to solve the problems. Because the string kernel takes very much computation time for computing their values, it was used for processing short strings or sentences rather than texts. In the previous works on encoding texts into tables, only table matching algorithm was proposed; there is no attempt to modify the machine algorithms into their table based version. In the previous works on encoding texts into string vectors, only frequency was considered for defining features of string vectors. Texts which are used as features of numerical vectors which represent words have their semantic similarities among them, so the similarities will be used for processing sparse numerical vectors, in this research.

3. Proposed Approach

This section is concerned with modifying the KNN (K Nearest Neighbor) algorithm into the version which considers the similarities among features as well as feature values, and it consists of the three sections. In Section 3.1, we describe the process of encoding words into numerical vectors. In Section III-B, we do formally the proposed scheme of computing the similarity between two numerical vectors. In Section 3.3, we mention the proposed version of KNN algorithm which considers the similarity among features as the approach to word categorization. Therefore, this article is intended to describe in detail the modified version of KNN algorithm and its application to the word categorization.

3.1 Word Encoding

This subsection is concerned with the process of encoding words into numerical vectors. Previously, texts each of which is consists of paragraphs were encoded into numerical vectors whose attributes are words. In this research, we attempt to encode words into numerical vectors whose attributes are text identifiers which include them. Encoding of words and texts into numerical vectors looks reverse to each other. In this Section, we describe in detail the process of mapping words into numerical vectors, instead of texts.

In the first step of word encoding, a word-document matrix is constructed automatically from a text collection called corpus. In the corpus, each text is indexed into a list of words. For each word, we compute and assign its weight which is called TF-IDF (Term Frequency-Inverse Document Frequency) weight [2], by equation (1),

$$w_i = TF_i(\log_2 N - \log_2 DF_i + 1) \tag{1}$$

where TF_i is the total frequency in the given text, DF_i is the total number of documents including the word, and N is the total number of documents in the corpus. The word-document matrix consists of TF-IDF weights as relations between a word and a document computed by equation (1). Note that the matrix is a very huge one which consists at least of several thousands of words and documents.

Let us consider the criterion of selecting text identifiers as features, given labeled sampled words and a text collection. We may set a portion of each text in the given sample words as a criteria for selecting features. We may use the total frequency of the sample words in each text as a selection criterion. However, in this research, we decided the total TF-IDF (Term Frequency and Inverse Document Frequency) which is computed by equation (1) as the criterion. We may combine more than two criteria with each other for selecting features.

Once some texts are selected as attributes, we need to consider the schemes of defining a value to each attribute. To each attribute, we may assign a binary value indicating whether the word present in the text which is given as the attribute, or not. We may use the relative frequency of the word in each text which is an attribute as a feature value. The weight of word to each attribute which is computed by equation (1) may be used as a feature value. Therefore, the attributes values of a numerical vector which represent a word are relationships between the word and the texts which are selected as features.

The feature selection and the feature value assignment for encoding words into numerical vectors depend strongly on the given corpus. When changing the corpus, different texts are selected by different values of the selection criterion as features. Even if same features are selected, different feature values are assigned. Only addition or deletion of texts in the given corpus may influence on the feature selection and the assignment of feature values. In order to avoid the dependency, we may consider the word net or the dictionary as alternatives to the corpus.

3.2 Feature Similarity

This subsection is concerned with the scheme of computing the similarity between numerical vectors as illustrated in Figure 1. In this research, we call the traditional similarity measures such as cosine similarity and Euclidean distance feature value similarities where consider only feature values for computing it. In this research, we consider the feature similarity as well as the feature value similarity for computing it as the similarity measure which is specialized for text mining tasks. The numerical vectors which represent texts or words tend to be strongly sparse; only feature value similarity becomes easily fragile to the tendency. Therefore, in this subsection, as the solution to the problem, we describe the proposed scheme of computing the similarity between numerical vectors.

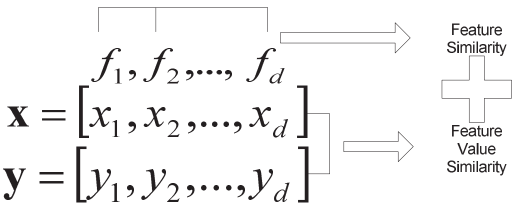


Fig. 1
THE COMBINATION OF FEATURE AND FEATURE VALUE SIMILARITY

Text identifiers are given as features for encoding words into numerical vectors. Texts are dependent on others rather than independent ones which are assumed in the traditional classifiers, especially in Naive Bayes [1]. Previously, various schemes of computing the semantic similarity between texts were developed [2]. We need to assign nonzero similarity between two numerical vectors where non-zero elements are given to different features with their high similarity. It is expected to improve the discriminations among sparse vectors by considering the similarity among features.

We may build the similarity matrix among features automatically from a corpus. From the corpus, we extract easily a list of text identifiers. We compute the similarity between two texts by equation (2),

$$s_{ij} = sim(d_i, d_j) = \frac{2 \times tf(d_i, d_j)}{tf(d_i) + tf(d_j)} \tag{2}$$

where $tf(d_i, d_j)$ is the number of words which are shared by both texts, d_i and d_j, and $tf(d_i)$ is the number of words which are included in the text, d_i. We build the similarity matrix which is consists of similarities between text identifiers given as features as follows:

$$S = \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1d} \\ s_{21} & s_{22} & \cdots & s_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ s_{d1} & s_{d2} & \cdots & s_{dd} \end{pmatrix}.$$

The rows and columns in the above matrix,S, correspond to the d text identifiers which are selected as the features.

The texts, $d_1, d_2, ..., d_d$ are given as the features, and the two words, t_1 and t_2 are encoded into the two numerical vectors as follows:

$$t_1 = [w_{11}, w_{12}, ..., w_{1d}]$$

$$t_2 = [w_{21}, w_{22}, ..., w_{2d}].$$

The features,$d_1, d_2, ..., d_d$ are defined through the process which was described in Section 3.1. We construct the d by d matrix as the similarity matrix of features by the process mentioned above. The similarity between the two vectors are computed with the assumption of availability of the feature similarities, by equation (3),

$$sim(t_1, t_2) = \frac{\sum_{i=1}^{d} \sum_{j=1}^{d} s_{ij} w_{1i} w_{2j}}{d \cdot \|t_1\| \cdot \|t_2\|} \tag{3}$$

where $\|t_1\| = \sqrt{\sum_{i=1}^{d} w_{1i}^2}$ and $\|t_2\| = \sqrt{\sum_{i=1}^{d} w_{2i}^2}$. We get the value of s_{ij} by equation (2).

The proposed scheme of computing the similarity by equation (3) has the higher complexity as payment for obtaining the more discrimination among sparse vectors. Let us assume that two d dimensional numerical vectors are given as the input for computing the similarity between them. It takes only linear complexity, $O(d)$, to compute the cosine similarity as the traditional one. However, in the proposed scheme takes the quadratic complexity, $O(d^2)$. We may reduce the complexity by computing similarities of some pairs of features, instead of all.

3.3 Proposed Version of KNN

This subsection is concerned with the version of K Nearest Neighbor which considers both the feature similarity and the feature value one. The sample words are encoded into numerical vectors whose features are texts by the scheme which was described in Section 3.1. The novice word is given as the classification target, and it is also encoded into a numerical vector. Its similarities with the sample words are computed by equation (3) for selecting nearest neighbors, in the proposed version. Therefore, in order to provide the detail algorithm, we describe the proposed KNN version, together with the traditional one.

The traditional KNN version is illustrated in Figure 2. The sample words which are labeled with the positive class or the negative class are encoded into numerical vectors. The similarities of the numerical vector which represents a novice word with those representing sample words are computed using the Euclidean distance or the cosine similarity. The k most similar sample words are selected as the k nearest neighbors and the label of the novice entity is decided by voting their labels. However, note that the traditional KNN version is very fragile in computing the similarity between very sparse numerical vectors.

The proposed KNN version is illustrated in Figure 3. Like the traditional version, a word is given as an input and it is encoded into a numerical vector. The similarities of the novice word with the sample ones are computed by equation (3) which was presented in Section 3.2. Like the traditional version, k most similar samples are selected as the nearest neighbors, and the label of the novice is decided by voting their labels. The scheme of computing the similarity between numerical vectors is the essential difference between the two versions.

We may derive some variants from the proposed KNN version. We may assign different weights to selected neighbors instead of identical ones: the highest weights to the first nearest neighbor and the lowest weight to the last

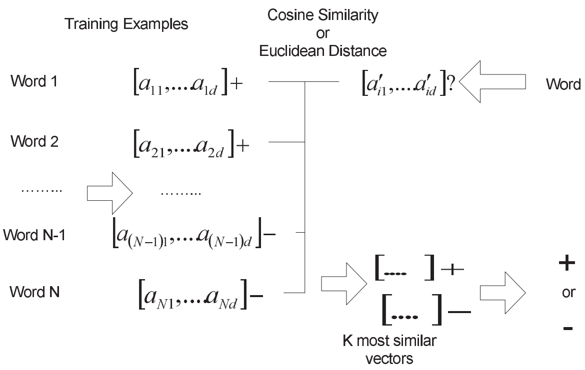


Fig. 2
THE TRADITIONAL VERSION OF KNN

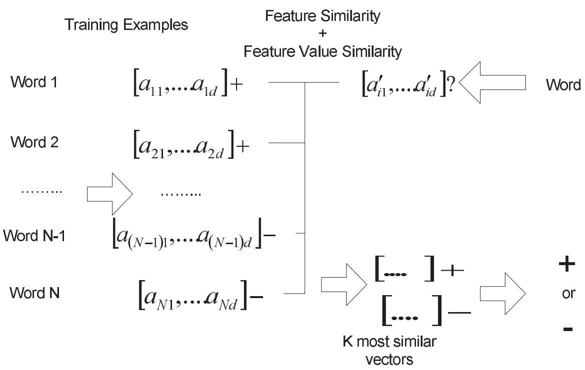


Fig. 3
THE PROPOSED VERSION OF KNN

one. Instead of a fixed number of nearest neighbors, we select any number of training examples within a hyper-sphere whose center is the given novice example as neighbors. The categorical scores are computed proportionally to similarities with training examples, instead of selecting nearest neighbors. We may also consider the variants where more than two variants are combined with each other.

Let us compare the both KNN versions with each other. In computing the similarity between two numerical vectors, the traditional version uses the Euclidean distance or cosine similarity mainly, whereas the proposed one uses the equation (3). Both versions are common in selecting k nearest neighbors and classifying a novice item by voting the labels of them. However, the proposed version is more tolerant to sparse numerical vectors in computing the similarities among them than the traditional version.

4. Conclusion

Let us mention the remaining tasks for doing the further research. We need to validate the proposed approach in specific domains such as medicine, engineering, and economics, as well as in generic domains such as ones of news articles. We may consider the computation of similarities among some main features rather than among all features for reducing the computation time. We try to modify other machine learning algorithms such as Naive Bayes, Perceptrons, and SVM (Support Vector Machine) based on both kinds of similarities. By adopting the proposed approach, we may implement the word categorization system as a real program.

References

[1] T. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

[2] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval: The Concepts and Technology behind Search*, Addison-Wesley, 2011.

[3] T. Jo, "The Implementation of Dynamic Document Organization using Text Categorization and Text Clustering" PhD Dissertation, University of Ottawa, Ottawa, Canada, 2006.

[4] F. Sebastiani, "Machine Learning in Automated Text Categorization", *ACM Computing Survey*, Vol. 34, pp. 1-47, 2002.

[5] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text Classification with String Kernels", *Journal of Machine Learning Research*, Vol. 2, pp. 419-444, 2002.

[6] C. S. Leslie, E. Eskin, A. Cohen, J. Weston, and W. S. Noble, "Mismatch String Kernels for Discriminative Protein Classification", *Bioinformatics*, Vol. 20, pp. 467-476, 2004.

[7] R. J. Kate and R. J. Mooney, "Using String Kernels for Learning Semantic Parsers", in *Proc. ICCL '06*, 2006, pp. 913-920.

[8] T. Jo and D. Cho, "Index based Approach for Text Categorization", *International Journal of Mathematics and Computers in Simulation*, Vol. 2, 2008, pp. 127-132.

[9] T. Jo, "Single Pass Algorithm for Text Clustering by Encoding Documents into Tables", *Journal of Korea Multimedia Society*, Vol. 11, 2008, pp. 1749-1757.

[10] T. Jo, "Device and Method for Categorizing Electronic Document Automatically", South Korean Patent 10-1071495, 2011.

[11] T. Jo, "Normalized Table Matching Algorithm as Approach to Text Categorization", *Soft Computing*, Vol. 19, 2015, pp. 849-849.

[12] T. Jo, "Inverted Index based Modified Version of K-Means Algorithm for Text Clustering", *Journal of Information Processing Systems*, Vol. 4, 2008, pp. 67-76.

[13] T. Jo, "Representationof Texts into String Vectors for Text Categorization", *Journal of Computing Science and Engineering*, Vol. 4, 2010, pp. 110-127.

[14] T. Jo, "NTSO (Neural Text Self Organizer): A New Neural Network for Text Clustering", *Journal of Network Technology*, Vol. 1, 2010, pp. 31-43.

[15] T. Jo, "NTC (Neural Text Categorizer): Neural Network for Text Categorization", *International Journal of Information Studies*, Vol 2, 2010, pp. 83-96.

A General Language Framework for General Intelligence

Sudharsan Iyengar, Theron Rabe
103 D Watkins Hall,
Computer Science Department
Winona State University
siyengar@winona.edu

Abstract: Natural language assimilation and utilization in communication and cognition are central to human intelligence. Natural languages *evolve* we humans assimilate these; thus we have so many different languages – in form and "grammar". One's language is the cumulative result of one's experience. Intelligent decision making is accomplished through deductive, inductive, and/or abductive reasoning processes. We propose the notion of a self-evolving general language L, a superset of all languages. We also define and develop a process of reduction, R, on L - as a way for decision-making in L. We then propose the use of a general framework based on lambda calculus for implementing this language and the reduction process.

Keywords—Artificial General Intelligence; General Language; Reduction; lambda calculus;

I. INTRODUCTION

The original goal of artificial general intelligence is to simulate intelligent behavior at or above human level [1]. One way to ascertain intelligence is to study and implement an efficient way to gather knowledge and make subsequent decisions [2]. The accuracy and speed with which decisions are made present the apparentness of intelligence in humans – and so has to be emulated in agents as well. It is also important for general intelligence there is the need to not only pre-defined problems but also an evolving scope of problems.

Humans use intelligence by reasoning over observations [2]. This reasoning falls under various forms of deductive, inductive, or abductive reasoning. If a person's repeated observations indicate that *A* must always imply *B* then *if A then B* is learned, and upon subsequent observation of *A*, *definitely B* is decided through *deductive* reasoning. Alternatively, if a person has observed that often (not always) *A* imply *B*, then *if A implies B* is learned and upon subsequent observation of A maybe B is decided through *inductive* reasoning. Finally, if a person observes that occasionally *A* imply *B*, then *B possibly* A is learned and subsequently when *B* is observed then *maybe A* is decided using *abductive* reasoning. Since inductive and abductive reasoning are inherently uncertain, they are of particular interest for solving problems suited for general intelligence, which usually involve a high degree of uncertainty. Such problems include machine learning, pattern classification, natural language processing, statistical analysis, computer vision, and data compression [3, 6].

Interestingly one trait of human intelligence is the ability to abridge statements to efficiently communicate. This ability to generate a shortened version of a complex version in one language can be construed as articulation or an abductive reasoning. An example of this can be an acronym TCP can stand for a stock symbol or for network protocol. This articulation and abductive reasoning is based on previous experience and the context of the evaluation. All decisions are based on previous experiences. Previously unknown expressions are either deemed irrelevant (and hence discarded) or considered acceptable new knowledge that is integrated into the language. Inductive and abductive reasoning requires earlier experience and decisions that help similar current decisions. For example, when a person (infers) induces that they will be hungry later because they did not eat breakfast, they must have previously experienced that not eating breakfast could cause hunger. The same can be said for a person who (feels) observes hunger and abductively attributes it to not having eaten breakfast. At the time of originally deciding that "not eating breakfast causes hunger", a minimum of two critical observations should have occurred. First, the person must have observed that they did not eat breakfast. Second, they must have later observed hunger. Any number of other intermediate observations (like "drank coffee" or "watched TV") may have occurred between these critical observations. Initially, all the relevant observations are included, considered, and processed in arriving at "not eating breakfast causes hunger". The number of observations included in such intermediate decision making can be called the critical observations' distance. Upon such repeated processing one tends to shorten the distance between observations in decision making. Thus, the act of correlating hunger to not eating breakfast is an act of shortening the distance between them (and pruning other insignificant observations).

The association of these two observations are utilized in inductive (forward or anticipatory) and abductive (backward or causal) reasoning. This process of learning, and subsequent shortening of the distance between cause and effect observations, is elementary in demonstrative intelligence.

Thus general intelligence possesses a primitive decision-making process that shortens the distance between critical observations. This process lowers the complexity of the decision-making process by removing unnecessary intermediate steps. In other words, an intelligent agent must be able to take a series of observations like $A \rightarrow B \rightarrow C$ and simplify it to decide that A may indicate C without regard to the presence of B, where applicable. We call this decision-making process reduction.

Natural languages, the primary visible forum for intelligence, are incremental, self-evolving, and self-mutating. The sounds and phrases as well as the grammar for natural languages are not set in stone – but are subject to change over time. One might infer grammar from them but there are many exceptions to these. One's vocabulary is a consequent of the environment in which one "grows". One learns French growing up France. The same person learns German (or Chinese) growing up in Germany (or China). This contextual ability to use vocabulary is also applicable to areas of education and work and social interactions. The syllables, words, sentences, and co-relation of phrases to semantics forming ones language is the cumulative result of experiences from the environment(s). Hence, the concept of general all-purpose language, and ones subset language that is currently accumulated.

Section II defines and describes the properties of the reduction process. Section III defines and describes the general language. Section IV presents the limitations of lambda calculus with respect to a general language and presents our approach to modifying lambda calculus so as to implement the features described in this paper.

II. REDUCTION - PROCESSING PHRASES IN A LANGUAGE

Reduction is manifested at multiple layers of abstraction within intelligent thought. People use language as a medium for abstracting their thought process. An observation abstracted in a language is a phrase. A phrase is either a symbol or a sequence of symbols within a language. For example, "rain" is a phrase composed of a single symbol that represents the observation of "water falling from the sky". Note that this process is independent of the natural language in question. (In Hindi – it could be "Baarish" – a two phrase with two syllables for rain). Even though the phrase is associated with semantics of the observation, we do have a co-relation between "rain" and "water falling from the sky". Reduction permits simplification of observations it permits simplification through abstractions. When complex phrases are interpreted by an intelligent agent reduction can be applied to shorten the relation between its sub-phrases and observations. This helps simplify the task of reasoning over their semantics as well.

For example, let us take the sentence: "Enough humidity has gathered in the air as to generate clouds of an unmaintainable density" which could be interpreted to the phrase "It is raining".

This sentence has multiple sub-phrases (observations) viz. enough humidity, gathered in the air, generate clouds, and unmaintainable density. Upon reasoning, the phrase becomes simpler but interprets the same. By utilizing a 'shortened' version of the original phrase, one is able to simplify the semantic interpretation of the original phrase. In other words, the reduced version is faster to interpret.

With respect to language, reduction is the translation of phrases to semantically-equivalent (or -approximately equivalent), but syntactically-minimal previously learned abstract phrases.

We now present the properties of such a reduction process. Correlation between phrases and semantics, when indicated, are presumed. The establishment and verification of the semantics to phrases are beyond the scope of this paper.

Definition 1: A language L = {T, N, G, S}, where T is a set of terminal phrases, N is a set of non-terminal phrases, G is set of production rules or its grammar, and S is its semantics.

The reduction process described in this paper does not take into account the semantics of the phrases, and is thus a mechanical process applied on the phrases alone.

Definition 2: Given a language L*, a phrase P in* L *is a sequence of symbols of the form* $\{s_1, s_2, .., s_n\}$ *such that* $0<i<n$, $s_i \in \{T \cup N\}$ *. Thus, all members of the power set* ($T \cup N$) *satisfy the definition of a phrase.*

Per this definition, any sequence of symbols are valid phrases that can be generated by using terminal and non-terminal phrases from the language. In practice only a subset of these are actually encountered and used by any agent.

Definition 3: Given a language L, *its grammar G is a set of production rules, each of the form A→B, where A and B are phrases in* L.

Definition 4: Given a language L, its semantics S = {(t, b)}; t ∈ T, and b is an observation.

An observation is some mechanical or logical effect on an L interpreter. For example, the terminal *apple* can be associated with the observation of an object apple (its image and associated properties, etc).

It is important to note that a phrase contains terminal and non-terminal symbols, but the semantics of the phrase is expressed by way of terminals only.

Definition 5: Given a phrase P, the set of symbols used in P is denoted {P} and its *size* is |{P}|

Definition 6: Given a language L, *the evaluation of a phrase* $P = \{s_1, s_2, .., s_n\}$, *denoted* P(), *is a function such that:*

$$P() = \{b \mid \forall s_i\ s \in T \text{ and } (t, b) \in S\} \cup \{P'() \mid \forall s_i \in N : (s_i \rightarrow P') \in G\},\ 0<i<n.$$

Where *P*' is some partial evaluation of *P*.

For a phrase – the evaluation is a set of observations obtained through the process where - the terminals provide the observations from S; and the phrase or non-terminals provide observations from their evaluation based on G. Thus the evaluation of a phrase yields a set of observations.

Definition 7: The effort of an evaluation, denoted E(P()), is given as follows:

$E(P()) = \sum E(s_i);\ 0<i<n,\ s_i \in \{P\}$ such that $E(s_i()) = 1$ for $s_i \in T$; $E(s_i())$ otherwise.

An evaluation function correlates a phrase to its abstracted observations, thus causing a series of mechanical or logical effects on an interpreter. The evaluation of a phrase is dependent on the symbols of the phrase as each symbol has to be evaluated. Terminals need no further reduction as they carry semantics. The non-terminals or phrases, recursively need further reduction.

Definition 8: Given a language L, *the reduction of a phrase P with respect to* L, *denoted* R(*P,* L), *is a function such that:*

R(*P,* L) = *p, where p is a phrase in* L *, and* $R(P,L) = R(p, L);\ P() = p();\ E(p()) \le E(P())$

First, that the reduction of phrase *P* is equivalent to the reduction of its reduction, *p*. That is, the reduction function is final. Second, that the evaluation of the phrase *P* will be equivalent to the evaluation of its reduction, *p*. In other words, reduction does not change the semantics of a phrase. Third, the complexity of evaluating the reduced phase is less than or equal to that of the original.

An input string is reduced in formal languages by iteratively applying the rewrite rules specified in the language's formal grammar, on an input string, until it cannot be further reduced. Since natural languages have no exact formal grammar, their reduction is more difficult to achieve. Reduction of a natural language depends on an accumulated familiarity with the phrases that constitute the language. The correlations and equivalences amongst these accumulated phrases behave as the language's grammar. Because reduction of a natural language depends on phrases having been learned and subsequently used in a meaningful way, natural language reduction appears indicative of intelligence.

Thus, to replicate this act of intelligence using artificial systems, the reduction process must be achievable in a language that is being prescribed through free use of previously unknown phrases that could become part of the language. Thus our proposal for a framework for a general language as opposed to a specific natural language. Since general intelligence processes must be applicable in broad domains we define a general language next.

III. General Language

We note that the intelligent behavior is dependent on what is known, understood, and utilized. Contrast this with an artificial system that can process phrases in the French language. This system is demonstratively limited in what it can accomplish because it is programmed as such, and it does not accommodate and/or learn other phrases. Humans on the other hand possess the ability to behave on what is assimilated, but additionally also accept and ingest new information, and thus evolve or grow. In fact, this is modus-operandi of human behavior. (Ironically, we consider this intelligent behavior and not the ability to process teraflops in milliseconds.) Importantly, note the language of a person is but that which has been assimilated and unrestricted, in contrast to what might be prescribed to be English, French, or the sign-language.

For the purposes of developing and implementing an intelligent machine we describe the notion of an unrestricted general language. This general language must satisfy the following three criteria:

- *General language must accept all possible phrases*
- *General language must be Turing-complete*
- *General language must be interpretable in-order*

Primarily, all potential phrases must be acceptable in the general language. This requirement implies that a general language has no predefined syntax rules. This is important as the order of the phrases is immaterial as long as the sentence is interpretable. Arguably, capability of interpretation without strict limitations on the order of the phrases, captures elementary intelligence. An example of this would be interpreting poetry as opposed to prose. Additionally, the general language must accept new previously un-encountered phrases - as legitimate phrases. The interpretation of such phrases is subject to the intent of observations associated with the phrase and other considerations.

Secondly, the general language must have Turing-complete semantics, so as to enable inference of a type 0 grammar [7]. Given this feature, we can automate the grammar application of this language, giving us the possibility of developing an AGI system.

Thirdly, we note that intelligent behavior generally interprets observations as they are input - without the need for a pre-requisite forward (anticipatory) reference. As such, the general language must accommodate interpretation without a requirement of forward reference. This requirement is further explained.

Since, this general language lacks definite syntax rules, it must be able to accommodate an infinite alphabet – though at any moment its alphabet is finite. An infinite set of symbols cannot be enumerated, as required for a formal grammar, but the set of contextually pertinent symbols can be. Consequently, during forward interpretation when a new symbol is encountered, the interpretation process must treat that symbol as a valid member of the language's alphabet in order to accept possible phrases with the new symbol.

***Remedy 1**: Represent infinite alphabet through its encountered subset.*

This simplification permits an interpreter to reason a partial formal grammar over an alphabet. Note that as a consequence, the interpreter must possess the ability to maintain a dynamic alphabet and grammar rules. As a general language interpreter is used, it will encounter an increasingly large set of phrases. As such, it must maintain a repository of phrases encountered so far, and utilize this repository in its future interpretations. Thus the interpreter must be able to maintain and use a dynamic set of terminals (and their associated observations).

Definition 9: *A set of encountered phrases $\{p_0..p_n\}$, represents an interpreter's history **P**.*

Due to the general language's need to be interpreted in-order, a function defined within phrase p_i must be expressed in terms relative to phrases $p_{0..(i-1)}$. In other words, the semantics of some future phrase is determined by its relation to past encountered phrases. Therefore, ***P*** represents a learned subset of the general language. This makes ***P*** an evolving construct analogous to a human's understanding and use of natural language. For example, a person might equate the phrase "rain" to "water that falls from the sky", but "water that falls from the sky" is just another phrase that can only be interpreted in terms of other learned phrases.

Definition 10: $\forall p_i \in \mathbf{P}\ (p_i() = f(p_0..p_{i-1}))$; *where p_k is an evaluation of P_k, $0<k<i$, and* f is some computable function.

Thus evaluation of any phrase by the interpreter is based on the ability to evaluate all previously encountered phrases – or the phrase is a new phrase in the interpreter's history.

Since a general language interpretation machine must be Turing-complete, it must support a means of defining and applying functions that support arbitrary recursion and abstraction. [4]

A machine that correctly interprets a general language, regardless of the semantics of that general language, will learn both the phrases and the grammar that constitute a subset of the general language. Since all languages are subsets of the general language, a general language interpreter can learn natural languages by interpreting input that causes it to construct a ***P*** that is approximate to some desired natural language in both phrase content and grammar. Because reduction is a computable function so with the formal grammar approximation of a natural language we have a mechanism to interpret this approximate natural language.

If semantics are defined for a general language approximation of a natural language, then reduction of this language is an approximation of intelligence use of this language. The speed and accuracy of these decisions improves as the language evolves within the system.

The interpreter starts with an empty set of terminals, non-terminals, empty grammar, and empty semantics. Through encounters the interpreter accepts newer symbols, phrases, and semantics. Over time, through a process of interpretations and threshold values the system is taught and evolves with a set of grammar as well for further interpretations.

Given formal semantics for a general language, an abstract machine can be designed for evaluation of general language strings. A machine that evaluates general language has an inherent ability to learn, due to general language's requirement of an extensible alphabet and grammar. Furthermore, since the interpretation machine must be Turing-complete, it has the ability to derive and perform computable function over its learned alphabet. Provided with the correct input string, an abstract machine that evaluates general language can learn both the phrases that constitute a natural language, as well as the functions that correlate those phrases within its language. Thus, a general language interpreter can be made capable of improving its "intelligence" with respect to any language (domain oriented information), and therefore, trained for different domains.

IV. LAMBDA CALCULUS AND ITS LIMITATIONS

To address the semantics for the general language, and exemplify the ambiguities that arise in doing so, we start with a Turing-complete language, and progressively remove all syntax rules. We use λ-calculus [5] as the starting language.

To exemplify the ambiguities that arise from removing syntax rules from λ-calculus, we will examine three syntactically invalid λ-expressions:

1. $\lambda xyz.a$
2. $\lambda\lambda x.F.a$
3. $\lambda\lambda x.xy.a$

Expressions (1), (2), and (3) each define a function whose body is composed of the symbol a and whose abstraction declaration contains syntax errors. Thus, in order for λ-calculus to meet the requirements of the general language, its semantics must be altered in such a way that each of these expressions is syntactically valid and unambiguously outputs the symbol *a*.

Expression (1)'s abstraction declaration contains three symbols (x, y, z) where only one is allowed by λ-calculus' formal grammar. To make this syntax valid, we suggest modifying λ-calculus such that a function with multiple symbols between λ and '.' is semantically equivalent to its fully curried version.

***Remedy 2**: $\lambda S.a = \lambda s_1.\lambda s_2. \ldots \lambda s_{n-1}.\lambda s_n.a$ for any sequence S of symbols $s_1..s_n$*

With this modification, Expression (1) becomes syntactically valid. And given any three inputs, Expression 1 retains unambiguous output of symbol *a*.

Expression (2) contains two consecutive λ symbols, so it can be referenced in parts. Call part "$\lambda x.F$ " the inner function, and everything else the outer function. Let F to be some oracle function that returns either symbol a or symbols xy. The output of F becomes the output of the inner function, which by way of Remedy 2 becomes the abstractions used by the outer function. Should F return symbol a, the outer function no longer outputs symbol a, and instead behaves as the identity function. Although the behavior of Expression (2) may arbitrarily change, it remains unambiguous in either definition it is dynamically given. We suggest the acceptance of semi-decidable function definitions by means of evaluating all definitions. Since definition is a prerequisite of application, any definition must be evaluated before its function can be applied. Because a function could potentially be applied immediately after definition, the expression containing its definition must be evaluated in-order.

***Remedy 3**:* $\lambda\lambda x.F.a \rightarrow \lambda\ (\lambda x.F.a)$

Expression (3) also appears to have an inner and outer function. Ambiguously, the inner function may consist of either $\lambda x.xy$ or $\lambda x.x$, depending on which function (inner or outer) owns symbol y. Should the inner function be provided another function for input x, that function x may be applied to one of two input sources, and in one of two orders. A function abstracted by x may be applied to y, or to whatever expression follows that which provided x. Additionally, that application may occur either before or after y has been provided with an expression to abstract. Depending on which of these evaluation pattern is taken affects Expression 3's ability to output symbol a. To correct this ambiguity, we suggest marking both the start and end of both function definitions and function inputs with dedicated symbols.

***Remedy 4**:* $\lambda x.y\ z \rightarrow (\lambda x.y)\ [z]$

By using these symbols purposefully and without restriction we can preserve the general language's first requirement (lack of syntax rules) and prevent ambiguity. This language is implemented as the EESK languages as described below.

V. CURRENT WORK & THE LANGUAGE - EESK

The High-level programming language Eesk is implemented and available on github. This language is based on lambda calculus and attempts to be a general language on the lines described in this paper. The Eesk system behaves as a lambda calculus interpreter that has, for the most part, remedied the ambiguities related to the double-lambda problem described above. With a few exceptions, this language meets all the three criteria of the general language.

The Eesk runtime environment has shown equivalent to an abstract machine that performs reduction on arbitrary learned languages for all halting inputs that have been tested. We intend to continue developing this system to use as a framework for further investigating the use of general language reduction as an approach to improving both the speed and accuracy of artificial general intelligence.

As with any correct implementation of the general language, Eesk's syntax is arbitrary. Valid Eesk is defined as any sequence of symbols. Conceptually, any symbol is either of the terminal or non-terminal type. Operators may be treated as terminal symbols. Operators that may be applied to an operand of one type may equally be applied to any operand of the other type. Thus, the language is weakly and dynamically typed. Since the typing is implicit, automatic, and prone to change, it does not necessarily concern an Eesk programmer.

Similar to other homoiconic functional languages like Scheme and Racket [8,9], Eesk is lexically scoped and full funarg [10] capable. The availability of symbols to their sub- and super-scope can be explicitly decided using "public" and "private" modifiers. Declaration of new symbols is done implicitly upon first encounter, defaulting to accessibility for all sub-scopes, but not the super-scope.

Due to general language's third requirement, Eesk may be parsed by a means as simple as LL(1) [11]. Each symbol encountered by such a naive left-to-right parser could be translated directly into machine code without respect to what symbols come next. The current implementation however, uses a recursive descent approach instead. Each descent may be implicitly escaped by encountering the end of a symbol stream. This solution permits much of the computational expense associated with determining scope to be handled at compile time.

To accommodate the remedies prescribed in this paper, Eesk employs a runtime architecture composed of three stacks, separating it from the list-processing approaches taken by philosophically similar languages [8, 9, 14]. The first of these stacks is used to store intermediate computed symbols, and the second to store function arguments. The Eesk calling convention causes these first two stacks to exchange responsibilities. This stack rotation method allows Eesk functions to both accept and produce syntactically arbitrary Eesk expressions without causing stack corruption. Furthermore, stack rotation permits the elements belonging to many sequential dynamic data structures to be accessed in constant time.

Eesk's third stack maintains control information for the calling convention, and its presence is opaque to an Eesk programmer. The third stack can be modeled using only the first two stacks, but in doing so, the runtime environment loses constant-time lookup of symbols in the super-scope.

Through the remedies provided in this paper, Eesk is a reflective language in which syntax is a first class citizen, and reduction of syntax is the primary mode of evaluation. Eesk expressions can be dynamically generated and evaluated by means of reduction. Beyond the primitive operators suggested for a pure reduction system, Eesk delivers additional predefined (but overridable) operator symbols that permit pattern matching between expressions, similar to use of (quote …) and (match …) in some languages [8,9] of LISP [14] heritage. Also, through intentional placement of function application operators, an Eesk programmer can explicitly denote whether a function is evaluated eagerly or lazily [12]. Additional features provided by the Eesk language framework include first class citizenship of continuations [13] and a foreign function interface.

The EESK language was implemented by an undergraduate senior student – Theron Rabe.

VI. CONCLUSION

We have defined complementary tools of reduction and general language that characterize general intelligence in language processing. The process of reductions is aimed at simplifying the complexity of decision-making over uncertain problem domains. The beneficial and problematic implications of implementing such a framework is discussed. The use of λ-calculus, and suggestions for modifying its syntactic structure to make it suitable for use as the general language, are presented as well. We are calling on the need for the formulation of formal semantics of the general language as an approach to general intelligence.

VII. REFERENCES

1. Allen Newell and Herbert A. Simon. 1976. Computer science as empirical inquiry: symbols and search. Commun. ACM 19, 3 (March 1976), 113-126. DOI=10.1145/360018.360022 http://doi.acm.org/10.1145/360018.360022

2. Sudharsan, Iyengar. Cognitive Primitives for Automated Learning, Frontiers in Artificial Intelligence and Applications, Vol. 171, AGI 2008, pp. 409-413.

3. Duda, Richard O., David G. Stork. Pattern Classification (Pt.1). (09 November 2000)

4. Turing, A. M. Computability and λ-Definability. The Journal of Symbolic Logic. Vol. 2, No. 4 (Dec., 1937), pp. 153-163

5. Church, Alonzo. An Unsolvable Problem of Elementary Number Theory

6. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning series) (24 August 2012)

7. Chomsky, Noam. On Certain Formal Properties of Grammars. (1959)

8. Gerald Jay Sussman and Guy L. Steele, Jr.. 1998. Scheme: A Interpreter for Extended Lambda Calculus. Higher Order Symbol. Comput. 11, 4 (December 1998), 405-439. DOI=10.1023/A:1010035624696 http://dx.doi.org/10.1023/A:1010035624696

9. Matthew Flatt. 2012. Creating languages in Racket. Commun. ACM 55, 1 (January 2012), 48-56. DOI=10.1145/2063176.2063195 http://doi.acm.org/10.1145/2063176.2063195

10. Joel Moses. 1970. The function of FUNCTION in LISP or why the FUNARG problem should be called the environment

problem. SIGSAM Bull. 15 (July 1970), 13-27. DOI=10.1145/1093410.1093411 http://doi.acm.org/10.1145/1093410.1093411

11. D. J. Rosenkrantz and R. E. Stearns. 1969. Properties of deterministic top down grammars. In Proceedings of the first annual ACM symposium on Theory of computing (STOC '69). ACM, New York, NY, USA, 165-180. DOI=10.1145/800169.805431 http://doi.acm.org/10.1145/800169.805431

12. Paul Hudak. 1989. Conception, evolution, and application of functional programming languages. ACM Comput. Surv. 21, 3 (September 1989), 359-411. DOI=10.1145/72551.72554 http://doi.acm.org/10.1145/72551.72554

13. Reynolds, J. C. (1993). The discoveries of continuations. Lisp and symbolic computation, 6(3-4), 233-247.

14. McCarthy, John. Recursive functions of symbolic expressions.

Example Based Machine Translation Using Fuzzy Logic from English to Hindi

Manish Rana
Research Scholar, Post Graduate Department of Computer Science & Engineering,
Sant Gadge Baba Amravati University,
Amravati, India
manishrana@live.com

Mohammad Atique
Associate Professor,
Post Graduate Department of Computer Science,
Sant Gadge Baba Amravati University,
Amravati, India
mohd.atique@gmail.com

Abstract: *Example based machine translation is one of the approaches in machine translation. The concept uses the corpus of two languages and then translates the input text to desired target text by proper matching. The different languages have different language structure of the subject-object-verb (SOV) alignment. The matching is then arranged to give proper meaning in target text language and to form proper structure. This paper, describes the Example Based Machine Translation using Natural Language Processing demo. The proposed EBMT framework can be used for automatic translation of text by reusing the examples of previous translations through the use of Fuzzy which is proposed work. This framework comprises of three phases, matching, alignment and recombination. Same type of machine translation is possible through use of soft computing tool (Fuzzy Logic).*

Keyword Terms: Machine Translation; Machine Learning; Natural Language Processing; Fuzzy Logic; Fuzzification; Fuzzy Rules; Visualization; Visualize Data; subsuming; etc.

I. Introduction

Machine Learning or Machine translation is a key to future for Artificial Intelligence world. It is the first step toward the growth of AI machines. These machines will be communicating to use in our language (Natural Language Processing). The question is “Is it possible?” The answer is “yes”, but “when”, “How much Time “required making these type of machine.

Machine learning techniques are often used for financial analysis and decision-making tasks such as accurate forecasting, classification of risk, estimating probabilities of default, and data mining. However, implementing and comparing different machine learning techniques to choose the best approach can be challenging. Machine learning is synonymous with **Non-parametric** modeling techniques. The term non-parametric is not meant to imply that such models completely lack parameters but that the number and nature of the parameters are flexible and determined from data.

In this example, several supervised machine learning techniques available in MATLAB are highlighted. One may apply one or more of the techniques and compare them to determine the most suitable ones for different data sets.

II. Related Work

Author describes here the translation of written texts or documents, not the interpretation of spoken utterances. The transfer of spoken to written language [7], and the synthesis of spoken language from written texts are topics in their own right which can be treated separately.

Machine translation of Indian Languages has been pursued mostly on the linguistic side. Hand crafted rules were mainly used for translation, [8][9]. Rule based approaches were combined with EBMT system to build hybrid systems and perform Interlingua based machine translation [10][11]. Input in the source language is converted into UNL, the Universal Networking Language and then converted back from UNL to the target language.

Recently, used linguistic rules are used for ordering the output from a generalized example based machine translation [12].While, in general in the machine translation literature, hybrid approaches have been proposed for EBMT primarily using statistical information most of which have shown improvement in performance over the pure EBMT system [13].

Automatically derived a hierarchical TM from a parallel corpus, comprising a set of transducers encoding a simple grammar [14]. Used example-based re-scoring method to validate SMT translation candidates and proposed an example based decoding for statistical machine translation which outperformed the beam search based decoder Showed improvement in alignment in EBMT using statistical dictionaries and calculating alignment scores bi-directionally combined the sub- sentential alignments obtained from the EBMT systems with word and phrase alignments from SMT to make ‘Example based Statistical Machine Translation‘ and ‘Statistical Example based Machine Translation‘ [14].

III. MACHINE TRANSALTION PROCESS

1) Segmenting documents into words, sentences and formatting information

The basic elements of translation programs are words and rules for combining them to form sentences, paragraphs and complete texts. Every document to be translated first needs to be decomposed into words, numbers and punctuation marks. Since the layout of the translation in most cases should look just like the original, this information must also be recognized so it can be inserted into the translation at the proper places.

2) Reduction of word forms to their canonical form and dictionary lookup

Every translation program needs a dictionary. Here all information is stored which is necessary for the analysis of sentences and their translation, e.g. part of speech, gender, or semantic classification.

3) Recognizing sentential structures

In the beginning many researchers believed that could obtain reasonable translations by having a program translate word by word. It became clear very quickly that this was an illusion, because firstly, languages differ very much in word order, and secondly, many words can have more than one meaning of which only one is valid in a given sentence. The results were completely unintelligible sequences of alternate word translations which nobody could use.

4. Assigning translations to single words

Each word and many word groups are associated with one or more translations in the dictionary. When after grammatical analysis the contexts of the words are known, the appropriate translations can be selected.

5. Generating the structure of target sentences

Starting from the structure of the source sentence and the word translations selected, the structure of the target sentence is built up. It can be quite different from the original. Thus

e.g. Sentence : "*India has won the match by six wickets*"
Tokens : *"India" "has" "won" "the" "match" "by" "six" "wickets"*

Becomes

भारत छह विकेट से मैच जीत लिया है
Bhārata chaha vikēṭa sē maica jīta liyā ha

because the word vikēṭa in Hindi is not transitive, and therefore an additional verb - jīta - is required as a kind of intermediary.

6. Generating word forms
During the generation of the correct word order for the target sentence, translation programs usually work with canonical forms or word stems. Only after the structure established, forms such as vikēṭa and maica of the previous example become Bhārata chaha vikēṭa sē maica jīta liyā ha "भारत छह विकेट से मैच जीत लिया है".

7. Adding layout information

The layout information which was taken out in the first step must now be added to the translations such that in the end there is a new text which almost looks like the original. One note may be in order here: some formatting information such as bold face must be available even during the translation process, since the corresponding translations should appear in bold as well.

Machine translation, sometimes referred to by the abbreviation MT (not to be confused with translation, machine-aided human translation (MAHT) or interactive translation is a sub-field of computational linguistics that investigates the use of software to translate text or speech from one natural language to another.

A. Tokenization

Tokenization is a primary step of Example based machine translation. In this phase, the input sentence is decomposed into tokens. These tokens are give n to POS stagger function to tag the tokens with their respective type.
e.g. Sentence : "*India has won the match by six wickets*"
Tokens : *"India" "has" "won" "the" "match" "by" "six" "wickets"*
भारत छह विकेट से मैच जीत लिया है

B. POS Tagger

A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some language and assigns parts of speech to each word (and other token), such as noun, verb, adjective, etc., although generally computational applications use more fine-grained POS tags like 'noun-plural'.
Matching Phase

Searching the source side of the parallel corpus for 'close' matches and their translations.. In matching phase, each token which is tagged with POS tag is searched in the dictionary of words and if match is found, then that word is passed to next phase**.**

C. Word-based Matching

Perhaps the "classical" similarity measure, suggested by Nagao (1984) and used in many early EBMT systems, is the use of a thesaurus or similar means of identifying word similarity on the basis of meaning or usage. Here, matches are permitted when words in the input string are replaced by near synonyms (as measured by relative distance in a hierarchically structured vocabulary, or by collocation scores such as mutual information) in the example sentences. This measure is particularly effective in choosing between competing examples, as in Nagao's examples [6], where, given (14a, b) as models, we choose the correct translation of *eat* in (15a) as *taberu* 'eat (food)', in (15b) as *okasu* 'erode', on the basis of the relative distance from *he* to *man* and *acid*, and from *potatoes* to *vegetables* and *metal*.

D. *Indexing*

In order to facilitate the search for sentence substrings, we need to create an inverted index into the source-language corpus. To do this we loop through all the words of the corpus, adding the current location (as defined by sentence index in corpus and word index in sentence) into a hash table keyed by the appropriate word. In order to save time in future runs we save this to an index file.

E. *Chunk searching and subsuming*

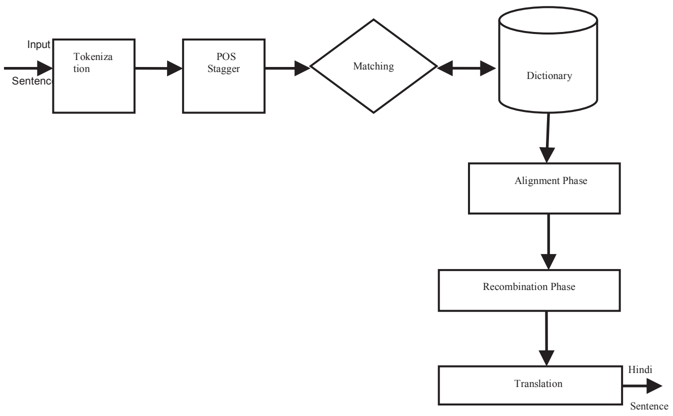


Figure: 1 Design of Proposed System for EMBT

A. *Alignment Phase*

Determining the sub sentential translation links in those retrieved examples.
The alignment algorithm proceeds as follows:
Stem the words of specified [1] source sentence look up those words in a translation dictionary Stem the words of the specified target sentence try to match the target words with the source words—wherever they match, mark the correspondence table. Prune the table to remove unlikely word correspondences. Take only as much target text as is necessary in order to cover all the remaining (unpruned) correspondences for the source language chunk. The pruning algorithm relies on the fact that *single* words are not often violently displaced from their original position

B. *Recombination Phase*

Recombining relevant [2] parts of the target translation links to derive the translation. Having matched and retrieved a set of examples, with associated translations, the next step is to extract from the translations the appropriate fragments ("alignment" or "adaptation"), and to combine these so as to produce a grammatical target output ("recombination"). This is arguably the most difficult step in the EBMT process [3]. difficulty can be gauged by imagining a source-language monolingual trying to use a TM system to compose a target

Keep two lists of chunks: current and completed.
Looping through all words in the target sentence:

See whether locations for the current word extend any chunks on the current list. If they do, extend the chunk.
Throw away any chunks that are 1-word. These are rejected.
Move to the completed list those chunks that were unable to continue. Start a new current chunk for each location
At the end, dump everything into completed.
Then, to prune, run every chunk against every other:
If a chunk properly subsumes another, remove the smaller one
If two chunks are equal and we have too many of them, remove one.

IV. Proposed Approach

A. *Translation*

In this phase, after matching and recombination, the matched words are mapped with the Hindi Corpus by searching. If it finds match, the Hindi word is substituted.
EBMT Implementation

Example Based Machine Translation is based on the idea to reuse the previously done translations as examples. Following are three examples are given. EBMT system tries to translate the given input English Text into Hindi by using these previous translations.

B. *Graphical User Interface (GUI)*

The Graphical User Interface is prepared for the project Example Based Machine Translation by using MATLAB GUI Editor utility.

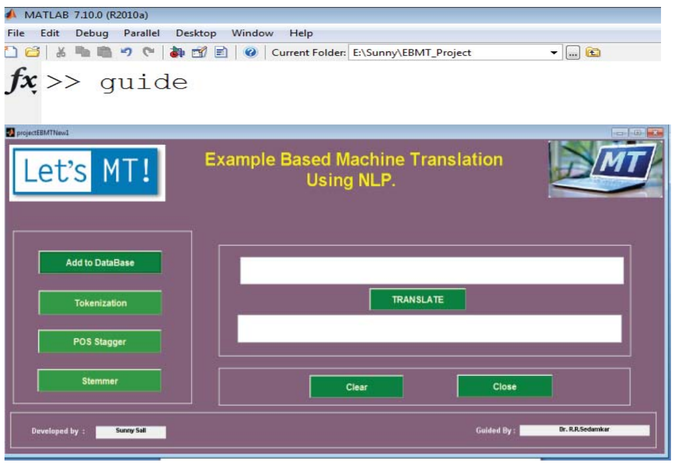


Figure2. Screen shot for User Interface

1. Add to Database

This option is used to add new word [5] and its corresponding Hindi word into corpus or dictionary. New English word along with its type i.e Noun, Pronoun, Verb, Adjective and Adverb can be added by using this form.

Figure 3. Add to dictionary

i	मै	noun
we	हम	noun
you	तुम	noun
you	तुम	noun
he	वो	pronoun
she	वो	pronoun
it	वह	pronoun
am	हू	verb
is	है	verb
are	है	verb
was	था	verb
were	थे	verb
have	है	verb
has	है	verb
has	है	verb
had	थी	verb
table	मेज	noun
cloths	कपडे	noun
india	भारत	noun
river	नदी	noun
water	पानी	noun
country	देश	noun
big	बडा	adjective
good	अच्छा	adjective
bad	बुरा	adjective

fresh	ताजा	adverb
this	ये	pronoun
that	वो	pronoun
door	दरवाजा	noun
medicine	दवा	noun
like	पसंद	verb
cricket	क्रिकेट	noun
small	छोटा	adjective
small	छोटा	adjective
broad	चौडा	adjective
game	खेल	noun
house	घर	noun
home	घर	noun
go	जाना	verb
come	आना	verb
walk	चलना	verb
walking	चल रहा	verb
very	बहुत	adjective
Sunny	सन्नी	noun
chating	वार्तालाप	verb
slowly	धीरे	adverb
great	महान	adjective
God	भगवान	noun
Ram	राम	noun
god	भगवान	noun

Figure : 4.Dictionary

2. Tokenization

Tokenization is a primary step of Example based machine translation. In this phase, the input sentence is decomposed into tokens. These tokens are give n to POS stagger function to tag the tokens with their respective type.

```
Input String :'India is my country. It is a great country!'

ring =

is my country. It is a great country!


 are 2 sentences in this paragraph !

 The Tokens are......
---------------------------------------------------------------

token =


    'India'    ' is'    ' my'    ' country.'           []

    'It'       ' is'    ' a'     ' great'     ' country!'
```

Figure: 5 Result of tokenization

3. POS Tagger

A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some language and assigns parts of speech to each word (and other token), such as noun, verb, adjective, etc., although generally computational applications use more fine-grained POS tags like 'noun-plural'.

```
inpPString =

Boy is very good

------------------------------------------------------------

 boy :  noun

 is :   verb

 very :     adjective

 good :     adjective
------------------------------------------------------------
```

Figure: 6 POS Tagger result

4. Stemmer :

This option is very useful to find stem i.e root word of any word :

e.g. for “running” stem is **run**

for “Indian” stem is **India**,

for “beautiful” stem is **beauty**.

```
Indian useful beutiful stemmer

----------------------------------------------------

 Stem = india
stem =

use


 Stem = use
 Stem = beuty
 Stem = stem
```

Figure: 7. Result of Stemmer

5. Translation

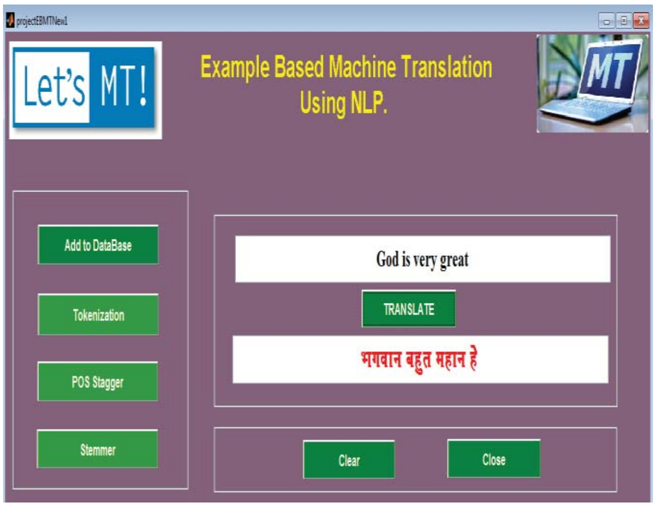


Figure: 8 Translations

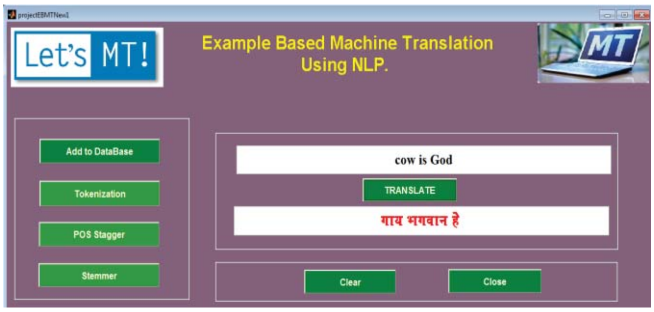
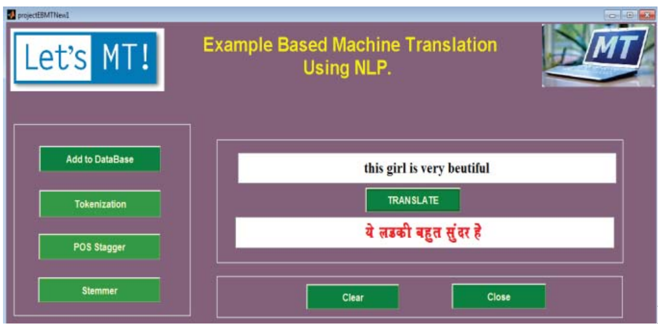


Figure: 9 Translations



F

Figure: 11 Translations

Example 1
English : India won the match.
Hindi : भारत ने मैच जीता

Example 2
English : India is the best
Hindi : भारत सबसे अच्छा है

Example 3
English : Sachin plays well
Hindi : सचिन अच्छी तरह से खेलते हैं

Input
English : Sachin is the best
Translation (Output)
Hindi : सचिन सबसे अच्छा है

Table 1 illustrates the comparison of three machine translation techniques, Rule-Based Machine Translation (RBMT), Statistical Machine Translation (SMT) and Example-Based Machine Translation (EBMT) on the basis of various parameters such as Consistency, predictable quality, Quality of out of domain translation, Use of grammar, robustness, Fluency and performance.

Table: 1 Comparison of various Machine Translation schemes

Parameter	**RBMT**	**SMT**	**EBMT**
Consistency	High	Low	Medium
Predictable Quality	Good	Similar	Very well
Out of Domain Quality	Medium	Low	High
Use of Grammar	Yes	No	No
Robust	Yes	No	Yes
Fluency	Less	Medium	High
Performance	Good	Medium	Good

Some translation results are also presented of some existing machine translation tools and there system in Table 2.

Table: 2 Comparison of translation of text from various Machine Translation tools

English Sentence	**Hindi Translation by existing MT tools**	**Hindi translation by our EBMT translation**
India is great	भारत है महान	भारत महान है
I am a boy	मैं हूँ लड़का	मैं लड़का हूँ
She is beautiful	वह है सुंदर	वह सुंदर है
He was great	वह है महान	वह महान है
Where do you live ?	आप हैं कहाँ रहते?	आप कहाँ रहते हैं?
She reads book	वह किताब है पढ़ता	वह किताब पढ़ता है
Milk is white	दूध है सफेद	दूध सफेद है

V. Results

Based on this model expected result is as following:
Input: Source Language, **ENGLISH** (SL).
Output: Target Language, **HINDI** (TL).
The result obtained is with minimal human interface.

Table: 3 Performance evaluation of EBMT

Corpus Size (No. of Sentences)	Unigram Precision	Unigram Recall	F-measure	BLEU	NIST	mWER	SSER
50	0.71	0.79	0.74	0.71	2.6	81.11	94.21
100	0.74	0.80	0.76	0.73	3.2	78.44	93.96
200	0.79	0.85	0.81	0.75	3.9	77.24	93.12
300	0.84	0.88	0.85	0.81	4.5	74.02	92.32
500	0.85	0.92	0.88	0.83	5.0	70.00	89.44
1000	0.90	0.94	0.91	0.91	6.6	65.22	81.77

VI. Conclusion & Future Scope

This research work proposes a new system, which is scalable, transparent and efficient. The entire system will convert the source language text into target language text using natural language processing. It will use the machine translation technique which is better than the existing tools available in the market.

The algorithm is such that, there is dictionary / corpus / vocabulary of **English** and **Hindi**. The parsing will be proper. The mapping technique will also be used. All the Literals will be separated using partitioning and stemming techniques. The root word will be identified using artificial intelligence and bilingual translation.

We pursue the study of example based machine translation using natural language processing.

VII. Reference

[1] Ali, V. Singh, “Potentials of Fuzzy Logic : An Approach To Handle Imprecise Data,” *American Medical Informatics Association*, vol 2, no. 4, pp. 358- 361, 2010.

[2] E. Binaghi, I. Gallo, C. Ghiselli, L. Levrini, K. Biondi, “An Integrated Fuzzy Logic And Web-Based Framework For Active Protocol Support,” *International Journal of Medical Informatics*, vol. 77, pp. 256-271, 2008.

[3] A. Ciabattoni, T. Vetterlein, K. K. Adlassnig, “A Formal Logical Framework for Cadiag-2,” *Studies in Health Technology & Informatics*, vol. 150, pp. 648- 652, 2009.

[4] H. Owaied, M. M. Qasem, “Developing Rule-Case- Based Shell Expert System,” *Proc.of Int. MultiConf. of Engineers & Scientists*, vol. 1,pp.234-240, 2010.

[5] M. G. Tsipouras, C. Voglis, D. I. Fotiadis, “ A Framework for Fuzzy Expert System Creation -Application to Cardiovascular Diseases,” *IEEE Transactions Biomedical Engg*., vol. 54, no. 11, pp. 2089-2105, 2007.

[6] X. He and L. Deng, “Speech-centric information processing: An optimization-oriented approach,” Proc. IEEE, vol. 101, no. 5, pp. 1116–1135, May 2013.Zadeh LA: Fuzzy sets. Information and control 8: 338-353, 1965 A system architecture for medical informatics. Fuzzy sets and systems 1994 ; 66 : 195-205.

[7] C. Liu, Y. Hu, L.-R. Dai, and H. Jiang, “Trust region-based optimization for maximum mutual information estimation of hmms in speech recognition,” IEEE Trans.Audio,Speech,Lang. Process., vol. 19, no. 8, pp. 2474–2485, Nov. 2011.

[8] F. Castro, A. Vellido, A. Nebot, and F. Mugica, ‘‘Applying data mining techniquestoe-learningproblems,’’inEvolutionofTeachingandLearning Paradigms in Intelligent Environment. Berlin, Germany: Springer-Verlag, 2007, pp. 183–221.

[9] Hongshen Chen, Jun Xie, Fandong Meng, Wenbin Jiang Qun Liu “*A Dependency Edge- based Transfer Model for Statistical Machine Translation*” Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, August 23-29 2014, pages 1103–1113

[10] J.Baker, L. Deng,J.Glass,S. Khudanpur,C.-H.Lee,N.Morgan,and D. O’Shgughnessy, “Research developments and directions in speech recognitionandunderstanding,parti,”IEEESignalProcess.Mag.,vol. 26, no. 3, pp. 75–80, 2009.

[11] H.Zen,M.J.F.Gales,Y.Nankaku,andK.Tokuda,“Productofexperts forstatisticalparametricspeechsynthesis,”IEEEAudio,Speech,Lang. Process., vol. 20, no. 3, pp. 794–805, Mar. 2012.

[12] L. Deng, “Switching dynamic system models for speech articulation and acoustics,” in Mathematical Foundations of Speech and LanguageProcessing. New York, NY, USA: Springer-Verlag, 2003, pp. 115–134

[13] G.Dahl,D.Yu,L.Deng,andA.Acero,“Context-dependentpre-trained deep neural networks for large-vocabulary speech recognition,”IEEE Trans. Audio, Speech, Lang. Process., vol. 20, no. 1, pp. 30–42, Jan. 2012.

[14] D.GolovinandA.Krause,“Adaptivesubmodularity:Anewapproach to active learning and stochastic optimization,” in Proc. Int. Conf. Learn. Theory, 2010.

Author Detail:

Manish Rana, PhD scholar Department of Computer Science, Sant Gadge Baba Amravati University, Amravati. He has more than seven years Teaching Experience. His area of interest includes Artificial Intelligence, Machine translation and soft computing. He has published 5 International Journal and 3 Papers in national Conference.

Dr. Mohammad Atique , is presently working as Associate Professor, P.G. Department of Computer Science, Sant Gadge Baba Amravati University, Amravati. He has around 37 publications to his credit in International/National Journal and conferences. His area of interest includes Artificial Intelligence, Machine translation and soft computing.

SESSION

XV TECHNICAL SESSION ON APPLICATIONS OF ADVANCED AI TECHNIQUES TO INFORMATION MANAGEMENT FOR SOLVING COMPANY-RELATED PROBLEMS

Chair(s)

Dr. David de la Fuente
Dr. Jose A. Olivas

Application of the FTOPSIS ranking method to an industrial facility location problem

J. Puente[1], **I. Fernandez**[1], **J. Lozano**[1], **B. Ponte**[1], **P. Priore**[1], **D. de la Fuente**[1]
[1]Business Administration Department, University of Oviedo, Gijón, Spain

Abstract - *Industrial location problems belong to the scope of multicriteria decision making (MCDM) given the high number of attributes to be considered in its planning and analysis. This work deals with a problem with four potential sites and five influential subjective criteria. A panel of experts related to the problem makes a fuzzy valuation of these attributes - allowing to capture the uncertainty inherent in the subjectivity of such assessments-. Similarly, using the AHP method, experts provide with compared importance ratings from pairs of these attributes, facilitating the calculation of their global weights of importance. Finally, the implementation of the FTOPSIS method, produces the ranking of the potential sites for the location problem.*

Keywords: facility location, AHP , FTOPSIS.

1 Introduction

Facility location decision is included within the decisions of a strategic nature in the fields of Operations Research and Business Organization. From the initial works by Weber [1] multiple contributions have been made to solve industrial facility location problems as it is shown in different literature reviews on the subject [2]–[4]. A joint optimization of some of the influential dimensions in the location decision (costs, revenues, service, distance, times of cycle or response, resiliency, etc. including labor costs, proximity to markets and customers, availability of suppliers, and even quality of life issues) is sought in these works.

Facility location problems may arise at different levels according to: the type of facilities (factories, warehouses, distribution centers (DCs) and retail outlets) and the hierarchy between them; the type and quantity of products involved and their corresponding flows; the criteria evaluated in each potential location (as well as their criticality and degree of objectivity) and the methodology used in the optimization of the criteria considered most relevant in each case [5].

Various reviews of the literature in this area [6]–[11] identified a large number of potential criteria with influence in the assessment of sites to be considered. Some of the most recurrently cited are: political risk, government barriers, other facilities, business climate, proximity to customers or/and suppliers, connectivity to intermodal transport means, investment cost, climate condition, labor force quality and quantity, to mention some.

Different multicriteria decision making (MCDM) tools can help in the task of evaluate a group of industrial facility locations alternatives based on a finite number of relevant criteria in their assessment - given by a decision-maker or a group of decision makers- [12]. A brief description of the methods most frequently used in the field of MDCM (Weighted sum model - WSM-, Weighted product model - WPM-, Compromise programming - CP-, Analytical hierarchy process - AHP-, Elimination and Choice Expressing Reality - ELECTRA - method, Technique for Order of Preference by Similarity to Ideal Solution - TOPSIS-, Preference ranking method for enrichment evaluation organization - PROMÉTHÉE - & Serbian Multicriteria Optimization and Compromise Solution - VIKOR-) can be found in [13].

In our case we have chosen the TOPSIS methodology due to its capacity to provide a well-structured analytical framework for the classification of alternatives, its intuitive condition when being applied and its property of extending its concept to the fuzzy sphere (FTOPSIS) –allowing to manage the intrinsic uncertainty inherent in the group decision maker judgements through the fuzzy sets theory [14]-.

In order to identify the critical attributes in the facility location problem and determine its value and importance, the opinion of experts in the field are usually gathered. To this end, the Delphi methodology proves to be very effective in obtaining, by means of structured communication, the required consensus among the group members [15].

2 FTOPSIS Method

TOPSIS ranks different alternative solutions to a problem according to their similarity to an ideal solution (ranking method). Thus, the best alternative should be as close as possible to the positive ideal solution (which minimizes the criteria of cost and maximizes the benefit) and as far as possible to the negative ideal solution (vice versa). However, the TOPSIS method does not allow, in practice, to manage the uncertainty and vagueness associated with the importance estimations and weights expressed by the decision-maker regarding the criteria to be evaluated. This problem can be overcome by allowing the valuation being formulated in linguistics terms (via fuzzy variables) using the theory of Fuzzy Sets. The fuzzy extension of the FTOPSIS methodology will allow to establish the ranking of alternative solutions to a problem incorporating the uncertainty inherent in the subjective valuations granted to the determining criteria.

The steps followed when the Fuzzy TOPSIS methodology is applied are outlined below:

Step 1. Determination of the fuzzy decision matrix.

Let Ci (i = 1.. n) be the criteria to evaluate and Aj (j = 1.. m) the facility alternatives whose ranking is pursued. Let x ̃_ij be the fuzzy evaluation (agreed upon by the group decision maker) on the "j" criterion related to the "i" site - a triangular fuzzy number (TFN). The decision fuzzy matrix will have the following structure:

$$[\widetilde{D_x}] = \begin{array}{c} \\ A_1 \\ A_2 \\ \cdots \\ A_i \\ \cdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccccc} C_1 & C_2 & \cdots & C_j & \cdots & C_n \end{array} \\ \begin{pmatrix} \tilde{x}_{11} & \tilde{x}_{12} & \dots & \dots & \dots & \tilde{x}_{1n} \\ \tilde{x}_{21} & \tilde{x}_{22} & \dots & \dots & \dots & \tilde{x}_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ & & \dots & \tilde{x}_{ij} & \dots & \\ & & \dots & \dots & \dots & \\ \tilde{x}_{m1} & \tilde{x}_{m2} & \dots & \dots & \dots & \tilde{x}_{mn} \end{pmatrix} \end{array} \quad (1)$$

Step 2. Normalization and weighting of the decision matrix.

In order to homogenize the evaluation supplied for all the criteria, their value must be linearly normalized in the [0,1] interval:

$$\tilde{x}_{ij}^* = \left(x_{ij_1}^*, x_{ij_2}^*, x_{ij_3}^*\right) \quad \wedge x_{ij_k}^* \in [0,1] \qquad (2)$$

On the other hand it is necessary to determine the importance of each criterion: W_j (the AHP method has been applied to the comparative valuations obtained from a panel of expert decision makers). Applying these importance weights of the criteria to the normalized fuzzy values, a normalized and weighted fuzzy decision matrix is obtained: $\tilde{v}_{ij} = W_j * \tilde{x}_{ij}^*$

Step 3. Calculation of the proximity coefficients for each alternative

Considering benefit- type criteria, the so called fuzzy positive ideal solution (FPIS) takes for each criterion the maximum fuzzy value of this criterion in all alternatives and the so called fuzzy negative ideal solution (FNIS), the minimum fuzzy value. In general, the real numbers "1" and "0" tend to be chosen - in their fuzzy representations-, to express both ideal values:

$$\tilde{v}_j^+ = (1,1,1);\ \tilde{v}_j^- = (0,0,0);\ \forall \mathrm{j} = 1,2,\dots,\mathrm{n}. \qquad (3)$$

Distances of each alternative to the ideal positive and negative solutions are calculated as:

$$d_i^+ = \sum_{j=1}^n d\left(\tilde{v}_{ij}, \tilde{v}_j^+\right) \quad \mathrm{i} = 1,2,\dots,\mathrm{m};\ \mathrm{j} = ,2,\dots,\mathrm{n}. \qquad (4)$$

$$d_i^- = \sum_{j=1}^n d\left(\tilde{v}_{ij}, \tilde{v}_j^-\right) \quad \mathrm{i} = 1,2,\dots,\mathrm{m};\ \mathrm{j} = ,2,\dots,\mathrm{n}. \qquad (5)$$

where $d\left(\tilde{v}_{ij}, \tilde{v}_j^+\right)$ is a measure of the distance between both fuzzy numbers.

The closeness coefficient of each alternative 'i' is calculated as:

$$CC_i = \frac{d_i^-}{d_i^- + d_i^+} \quad \forall i = 1,2,\dots m, \qquad (6)$$

and to establish the ranking of alternatives, it is enough to sort these according to decreasing values of their closeness coefficient.

3 Implementation of FTOPSIS in a facility location problem

By way of example, the previous methodology is applied below to an industrial facility location problem, in which 4 feasible sites had been a priori identified (denoted as A, B, C and D).

In this example, the appropriate questionnaires have been given to a panel of experts following the Delphi methodology. After two rounds of analysis with feedback, five criteria were identified as the most relevant and influential in the problem (denoted as C1, C2, C3, C4 and C5). Likewise, the importance weights of the criteria were obtained by the AHP methodology through the pairs of criteria importance comparison and finally the linguistic assessment assigned to each of these five criteria was averaged at each potential location

Using the linguistic labels shown in table 1, the team of experts fuzzily assessed the five criteria (C1, C2, C3, C4 and C5) in the four different possible sites (A, B, C and D). After averaging such values, the fuzzy decision matrix was obtained as illustrated in table 2.

After the corresponding normalization process (in order to obtain the ratings within the range 0-1), the normalized fuzzy matrix was obtained (see table 3).

Subsequently, the AHP methodology was used to obtain the global importance weights of each of the five criteria, (a clear and detailed explanation can be found in [13]. With this methodology the expert panel consensus was obtained from the pair-compared criteria matrix given in table 4

From table 4, the paired comparisons matrix, its corresponding normalized matrix and the estimated weight of importance of each criterion –obtained as the average of each row from the normalized matrix- are calculated (see table 5):

To assess the reliability of these estimates, the consistency ratio (CR) of the problem was calculated (as a ratio between the consistency index and the random index – related to the 5 criteria)-. A sufficiently significant consistency ratio (< 0,05) was obtained CR=0,048.

Afterwards, the results shown in table 6 were determined by pondering the values from table 3, according to the weights (W*) obtained in the last step,

Table 1: linguistic labels applied to the criteria evaluations

Crisp Value	0	1	2	3	4	5	6	7	8	9	10
TFN	(0; 0; 1)	(0; 1; 2)	(1; 2; 3)	(2; 3; 4)	(3; 4; 5)	(4; 5; 6)	(5; 6; 7)	(6; 7; 8)	(7; 8; 9)	(8; 9; 10)	(9; 10; 10)

Table 2: Fuzzy matrix of criteria valuations

	A			B			C			D		
C1	5,88	6,75	7,84	7,14	8,14	9,02	4,89	5,25	6,15	6,50	7,46	8,36
C2	7,43	8,06	8,86	3,26	4,06	4,92	6,68	7,68	8,55	4,23	5,14	6,05
C3	5,27	6,27	7,25	6,96	7,96	8,84	8,55	9,23	9,87	6,20	7,20	8,10
C4	6,07	7,07	8,06	5,92	6,92	7,92	4,32	5,25	6,16	3,25	4,14	5,00
C5	3,85	4,82	5,81	6,24	7,05	7,86	4,50	5,02	5,62	3,60	4,50	5,50

Table 3: Normalized fuzzy matrix of criteria assessments

	A			B			C			D		
C1	0,65	0,75	0,87	0,79	0,90	1,00	0,54	0,58	0,68	0,72	0,83	0,93
C2	0,84	0,91	1,00	0,37	0,46	0,56	0,75	0,87	0,97	0,48	0,58	0,68
C3	0,53	0,64	0,73	0,71	0,81	0,90	0,87	0,94	1,00	0,63	0,73	0,82
C4	0,75	0,88	1,00	0,73	0,86	0,98	0,54	0,65	0,76	0,40	0,51	0,62
C5	0,49	0,61	0,74	0,79	0,90	1,00	0,57	0,64	0,72	0,46	0,57	0,70

Table 4. Compared Importance of the criteria

	More important than								Equal	Less important than								
	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	
C1												X						C2
C1									X									C3
C1												X						C4
C1							X											C5
C2						X												C3
C2									X									C4
C2	X																	C5
C3									X									C4
C3						X												C5
C4				X														C5

Table 5: Paired comparisons matrix, normalized matrix and the estimated weight of importance of each criterion (W*)

	C1	C2	C3	C4	C5
C1	1	0,25	1	0,25	3
C2	4	1	4	1	9
C3	1	0,25	1	1	4
C4	4	1	1	1	6
C5	0,33	0,11	0,25	0,17	1
Sum	10,33	2,61	7,25	3,42	23,00

	C1	C2	C3	C4	C5	W*
C1	0,10	0,10	0,14	0,07	0,13	0,11
C2	0,39	0,38	0,55	0,29	0,39	0,40
C3	0,10	0,10	0,14	0,29	0,17	0,16
C4	0,39	0,38	0,14	0,29	0,26	0,29
C5	0,03	0,04	0,03	0,05	0,04	0,04

Table 6: Fuzzy normalized and weighted matrix of the criteria evaluation

	A			B			C			D		
C1	0,07	0,08	0,09	0,08	0,10	0,11	0,06	0,06	0,07	0,08	0,09	0,10
C2	0,34	0,36	0,40	0,15	0,18	0,22	0,30	0,35	0,39	0,19	0,23	0,27
C3	0,09	0,10	0,12	0,11	0,13	0,14	0,14	0,15	0,16	0,10	0,12	0,13
C4	0,22	0,26	0,29	0,21	0,25	0,29	0,16	0,19	0,22	0,12	0,15	0,18
C5	0,02	0,02	0,03	0,03	0,04	0,04	0,02	0,03	0,03	0,02	0,02	0,03

Then, given that all the considered criteria are of benefit type (the higher the score, the higher the satisfaction) and the TFN in the matrix are in the range [0,1], the fuzzy ideals reference points (positive "v +" and negative "v-", respectively) are defined: v^{+}= [(1,1,1), (1,1,1), (1,1,1), (1,1,1), (1,1,1)]

v^{-}= [(0,0,0), (0,0,0), (0,0,0), (0,0,0), (0,0,0)]

Finally, the distances of each of the criterion with respect to such fuzzy ideals reference points are calculated and the closeness coefficient is estimated for each location.

By way of example, the following equations represent the calculation of both distances for the site "A" as well as the closeness coefficient.

$$d_1^+ = \sqrt{\frac{1}{3}[(0.07-1)^2+(0.08-1)^2+(0.09-1)^2]} + \sqrt{\frac{1}{3}[(0.34-1)^2+(0.36-1)^2+(0.40-1)^2]} + \cdots + \sqrt{\frac{1}{3}[(0.02-1)^2+(0.02-1)^2+(0.03-1)^2]} = 4.171$$

$$d_1^- = \sqrt{\frac{1}{3}[(0.07-0)^2+(0.08-0)^2+(0.09-0)^2]} + \sqrt{\frac{1}{3}[(0.34-0)^2+(0.36-0)^2+(0.40-0)^2]} + \cdots + \sqrt{\frac{1}{3}[(0.02-0)^2+(0.02-0)^2+(0.03-0)^2]} = 0.835$$

$$CC_1 = \frac{d_1^-}{d_1^- + d_1^+} = 0.167$$

Results for all locations analyzed as well as their final ranking are illustrated in the Table 7.

Table 7: Distances, closeness coefficients and ranking of sites

Sites	d_i^+	d_i^-	CC_i	Ranking
A	4,171	0,835	0,167	1
B	4,306	0,701	0,140	3
C	4,226	0,779	0,156	2
D	4,392	0,616	0,123	4

Thus, in view of the results the ranking of the sites proved to be "A", followed by "C", then "B" and last "D".

4 Conclusions

In this work a method of easy application is developed to solve any industrial facility location problem subject to the assessment of subjective criteria that are influential in the choice of an optimal location. A specific example illustrates its application. By using the Delphi method, two rounds of consultations to a panel of experts provided with the relevant information to identify the critical criteria for the location problem and to establish the criteria ratings assigned to each potential location and the compared importance between each two of these criteria.

The utilization of the AHP methodology allows the criteria importance weights vector to be calculated from the comparative importance provided by the experts. This vector permits to distinguish the decisive criteria in each problem. On the other hand, estimations assigned by experts to each attribute in each location are analyzed and summarized by fuzzy triangular numbers. This process allows to manage the uncertainty inherent in the subjectivity of such assessments.

Finally, the application of the FTOPSIS methodology results in a metric of each site that measures the proximity of the joint evaluation of all criteria to the ideal optimum solution. This metric facilitates to establish the ranking of all potential sites and, out of it, choose the optimum.

5 References

[1] Weber, Alfred, Theory of the Location of Industries. Universit of Chicago Press, 1909.

[2] R. Z. Farahani, N. Asgari, N. Heidari, M. Hosseininia, y M. Goh, «Covering problems in facility location: A review», Comput. Ind. Eng., vol. 62, n.o 1, pp. 368-407, feb. 2012.

[3] R. Z. Farahani, M. SteadieSeifi, y N. Asgari, «Multiple criteria facility location problems: A survey», Appl. Math. Model., vol. 34, n.o 7, pp. 1689-1709, jul. 2010.

[4] M. T. Melo, S. Nickel, y F. Saldanha-da-Gama, «Facility location and supply chain management – A review», Eur. J. Oper. Res., vol. 196, n.o 2, pp. 401-412, jul. 2009.

[5] G.-S. LIANG y M.-J. J. WANG, «A fuzzy multi-criteria decision-making method for facility site selection», Int. J. Prod. Res., vol. 29, n.o 11, pp. 2313-2330, nov. 1991.

[6] A. Awasthi, S. S. Chauhan, y S. K. Goyal, «A multi-criteria decision making approach for location planning for urban distribution centers under uncertainty», Math. Comput. Model., vol. 53, n.o 1-2, pp. 98-109, ene. 2011.

[7] J. Heizer y B. Render, Principles of Operations Management, 8 edition. Upper Saddle River, N.J: Prentice Hall, 2010.

[8] W. J. Stevenson, Operations Management. McGraw-Hill/Irwin, 2005.

[9] B. J. Finch, OperationsNow.com: processes, value, and profitability. New York; Montréal: McGraw-Hill, 2003.

[10] C. Kahraman, D. Ruan, y I. Doğan, «Fuzzy group decision-making for facility location selection», Inf. Sci., vol. 157, pp. 135-153, dic. 2003.

[11] C.-T. Chen, «A fuzzy approach to select the location of the distribution center», Fuzzy Sets Syst., vol. 118, n.o 1, pp. 65-73, feb. 2001.

[12] F. A. Lootsma, Multi-Criteria Decision Analysis via Ratio and Difference Judgement, 1999 edition. Dordrecht, Netherlands ; Boston: Springer, 1999.

[13] G. Kabir, R. Sadiq, y S. Tesfamariam, «A review of multi-criteria decision-making methods for infrastructure management», Struct. Infrastruct. Eng., vol. 10, n.o 9, pp. 1176-1210, sep. 2014.

[14] S.-Y. Chou, Y.-H. Chang, y C.-Y. Shen, «A fuzzy simple additive weighting system under group decision-making for facility location selection with objective/subjective attributes», Eur. J. Oper. Res., vol. 189, n.o 1, pp. 132-145, ago. 2008.

[15] J. Pill, «The Delphi method: Substance, context, a critique and an annotated bibliography», Socioecon. Plann. Sci., vol. 5, n.o 1, pp. 57-71, feb. 1971.

Heat Exchanger Network (HENET): A parallel GA/SA algorithm

A. Gómez[1], D. Toimil[1], R. Rosillo[1], J. Parreño[1], N. García[1], D. de la Fuente[1]
[1]Business Administration Department, University of Oviedo, Gijón, Spain

Abstract - *Heat exchanger network (HEN) synthesis has been a well-studied subject over the latest decades. Many studies and methodologies were proposed to make possible the energy recovery. Based on simulated annealing and genetic algorithm, this paper presents an efficient simultaneous synthesis method that provides the optimal networks in a two-level procedure. Genetic algorithm is used by an evolutionary algorithm to manage HEN topology and simulated annealing is used to manage heat load distribution among exchangers. This two-level method is applied to solve one benchmark that includes 28 instances with different dimensions, from 3 to 39 streams. For every instance, this benchmark includes the best result obtained with a state-of-the-art algorithm from the literature, which has been useful in this work to compare our proposed hybrid algorithm with the best existing approaches. The results of this study show that, in some situations, the hybrid approach is able to derive networks that are more economical than those from the known solutions in the literature.*

Keywords: heat exchanger network synthesis; genetic algorithm; simulated annealing; hybridization; optimization; benchmark.

1 Introduction

The problem of synthesizing optimal network configurations has received considerable attention in the literature in the last few decades, so as for its significant impact on energy and cost saving in industry. The objective is to design a heat exchanger network that minimizes total annualized cost (TAC) as the sum of annualized investment cost and annual operating cost with the given sets of streams and utilities.

The complexity of the HENs has a combinatorial nature. For a fixed number of streams, there are a wide range of possibilities of combinations among exchangers. However, the number of possible HEN configurations that contains the minimum utilities consumption is smaller than the entire number of configurations. This restriction ensures finding a HEN with the minimum utilities to a presented minimum temperature of approach (ΔTmin).

For addressing the HEN synthesis problem most of methods can be grouped into three different lines, which are thermodynamic based approaches, mathematical programming methods and metaheuristic optimization methods.

On one hand, thermodynamic approaches based on the pinch analysis by Linnhoff and Flower [1] and Linnhoff and Hindmarsh [2] are most commonly used. Pinch analysis method is flexible and provides an overview of the problem. It creates the problem into a sub-problems based on the concept of pinch and with various targets, which are then solved sequentially. The targets include minimum approach temperatures, ΔTmin. They will illustrate for the cumulative cost of the heat exchanger network, and certain way they can define the optimal level of ΔT min or can be used as an instrument of the optimization progresses. A review on this method was collected by Shenoy [3].

On the other hand, mathematical programming methods could solve the problem with and without decomposition. With decomposition, commonly called sequential approach, the reduction of computational complexity is found. This method usually have mixed integer linear programming (MILP) or non-linear programming (NLP) formulation. The most well-known works are the transshipment model Papoulias and Grossmann [4], the explanation of Biegler [5], and superstructure model Floudas [6].

Without decomposition, frequently called simultaneous approaches, near global optimal solutions are found by mixed integer non-linear programming (MINLP) formulations. Non-convex terms as the LMTD of heat exchangers, the energy balances for mixers and splitters and the non-linear area cost function make the solution of these models much more difficult. For this reason, some simplifications in the problem must be done in order to reduce its complexity. For instance, the stage-wise network superstructure proposed by Yee [7] make the assumption of isothermal mixing for streams. Chen approximation of LMTD [8] term is usually used to avoid numerical difficulties, when the approach temperatures of both sides of the exchanger are equal. Other well-known approximation was made by Paterson [9]. Ciric et al [10] collected a review of mathematical programming method.

Finally the third line, metaheuristic optimization methods, such as Simulated Annealing (SA), have been applied by Athier et al [11], Tabu Search (TS) by Lin and miller [12] and Genetic Algorithm (GA) by Lewin et al. [13], another contribution of their work is to introduce the concept of 'HEN level' for structure representation which was then used in a two-level synthesis method of HENs combining harmony search (HS) and sequential quadratic programming (SQP) [14]. Differential evolution (DE) algorithm for synthesis of HENs have been proposed by Yerramsetty and Murty [15] and also utilized the structure representation similar to 'HEN level'. A particle swarm optimization (PSO) method and a GA/PSO algorithm have been presented by Silva et al. [16] and Huo et al. [17] respectively. These techniques are able to solve complex problems without being limited by non-linearities, non-convexities and discontinuities of the models. All of them are robust and can find near optimal solution by means of searching space within a reasonable time but they still have the difficulty in converging to the precise global optimal solution in the feasible region.

This paper proposes a benchmark that has 28 problems with the same characteristics. To test this benchmark, some tests with hybrid algorithm are conducted. In the proposed hybrid algorithm, stochastic methods are combined in a two-level approach to take the advantage of each method and compensate deficiencies of individual methods. This hybrid algorithm uses the GA and SA algorithm; GA is used to create for network structural, while the fitness of each structure is calculated by SA.

The remaining parts of this paper are organized as follow. In section 2, the mathematical formulation of HENS is presented. Section 3 describes the proposed two-level synthesis method in detail. Section 4 briefly describes available benchmark instances and also results obtained followed by conclusions in Section 5.

2 Mathematical Formulation

The HENs problem was first rigorously defined by Masso and Rudd [18] and its objective is to find a sequence of combining exchangers in pairs of streams, getting that the network either optimal in relation to the global cost.

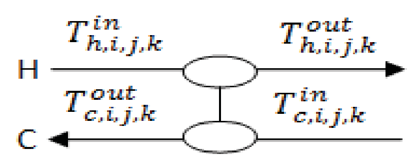
Section 2 presents the mathematical formulation solved by the two-level simultaneous synthesis method in this paper.

2.1 HEN Structure Representation

A structure representation based on superstructure proposed by Yee [7] is presented, it has stages where only one exchange is allowed between a specific hot stream and a specific cold stream, however this stage wise superstructure allows a stream to split into several substreams (or branches) at each stage to exchange heat with other streams of the opposite kind. The utilities streams are placed at the ends of the sequence of stages.

A stage-superstructure with branch involving two hot and two cold streams along with cold and hot utilities is shown in Fig. 1.

For single heat exchanger with counter-current flow patterns, the feasibility of heat exchange temperature difference is rigorously required as shown in Fig. 2.



$$T^{in}_{h,i,j,k} > T^{out}_{c,i,j,k}$$

$$T^{out}_{h,i,j,k} > T^{in}_{c,i,j,k}$$

Figure 2: Modification of HEN representation

3 Method

A two-level method is proposed to optimize the binary and continuous variables, one is based on Genetic Algorithm (GA) [19] and other is settle on Simulated Annealing (SA) [20]. Both of them are in principle random methods generally used to solve large scale combinatorial optimization problems.

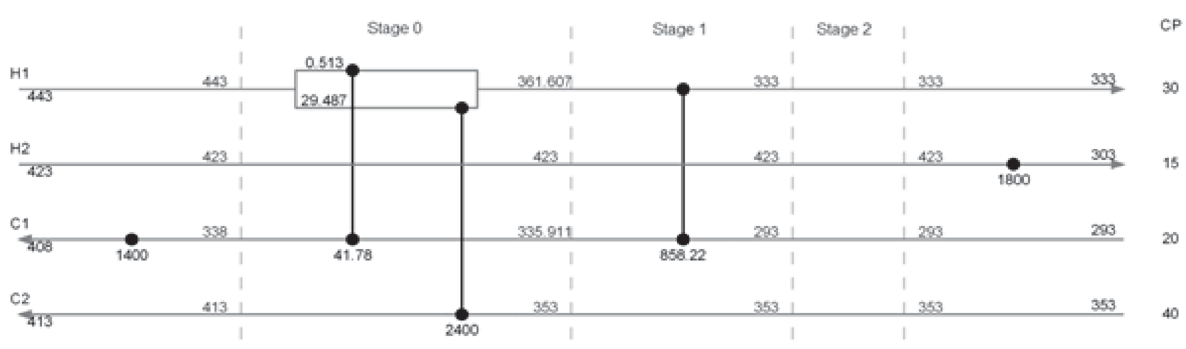


Figure 1: Stage-wise superstructure

GA is applied for binary variables optimization to search optimal network structure since it has been proven to be a powerful discrete variables optimization algorithm of combinatorial problem and SA algorithm is applied to the continuous variables optimization as well and designed to converge to the optimal heat distribution of each candidate structure with low computational effort [21, 22]. Fig. 3. Illustrates the basic concept of the two-level method.

SA can find good quality solutions in a neighborhood, but most it will get trapped in local minimum and takes longer to scope, while GA rapidly discovers the search space, but has difficulty in finding the exact minimum. For this reason a parallel GA/SA hybrid has been adopted in the present work where in the upper level, a series of candidate structures will be generated by specified strategy by GA and then sent to the lower level SA for solving minimum TAC until converging to an optimal HEN solution.

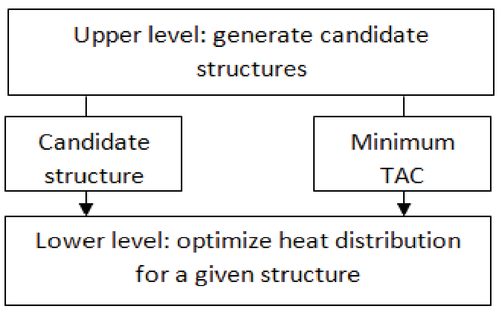


Figure 3: Basic concept of the two-level method

3.1 Genetic algorithm

Genetic algorithm is settling on the natural selection and genetic mutation in biological world. The genetic algorithm consists of three main operators: selection, crossover, and mutation. The individual with a better value of fitness (lower value of the objective function) has a greater chance to be selected to produce its offspring by crossover, or to return directly to the next generation. By using a crossover operation, two selected parents are combined to form their offspring. A mutation operation will introduce new genes into the population to avoid the evolution converging into a local optimum.

3.2 Simulated Annealing algorithm

Simulated annealing is organized according to the Monte Carlo simulation technique developed by Metropolis et al. and the theory of Markov [23] chains provides mathematical properties about its asymptotic convergence. The simulated annealing algorithm was firstly introduced to solve large combinatorial optimization problems by Kirkpatrick et al. [20] who drew an analogy between the annealing of a solid and the optimization of a complex system. For it accepts and rejects 'moves' generated randomly on the basis of a probability related to an 'annealing temperature', SA can accept uphill moves and consequently escape from a local optimum. Obviously, the accepted proportion of uphill moves increases with the annealing temperature T. Until a specified stop criterion is satisfied, the annealing temperature is periodically reduced according to the annealing schedule. The higher the temperature, the larger the possibility of having accepted random moves. Therefore, the ability of this algorithm to escape from the region of poor local optima can be controlled by adjusting the annealing schedule.

3.3 Structure of the two-level method

The global optimization procedure can be summed up as follows. In the upper level, the candidate structure combined by 0 or 1 binary variables are evaluated based on the minimum TACs solved in the lower level. The current structure will be gradually improved by the genetic and simulated annealing mechanisms and finally converge to an optimal structure. The overall algorithm is illustrated in Fig. 4.

The optimization is started with a randomly initial population of structures that is produced by the GA. The structure search should be performed in a sufficiently feasible space to guarantee that the optimal structure is involved. A topological structure will be generated randomly as the first current structure, where zi,j,k = 1 denotes that a heat exchanger is matched between ith hot stream and jth cold stream at the kth stage. Each exchanger has an associated value, qi,j,k, which represents the exchanged heat load in this exchange. In the beginning, this value is set to 0 and the whole heat load is carried out by the utilities. SA is the responsible of the optimization of these heat loads as long as the total annual cost is concerned. Notice that SA is not allowed to change z structure; SA only handles the values of heat loads related to each exchange for calculating TAC.

After the optimal heat distribution for a structure has been obtained by SA, GA handles it within a population in order to obtain the best structure for applying operators by roulette-wheel procedure and obtain offspring.

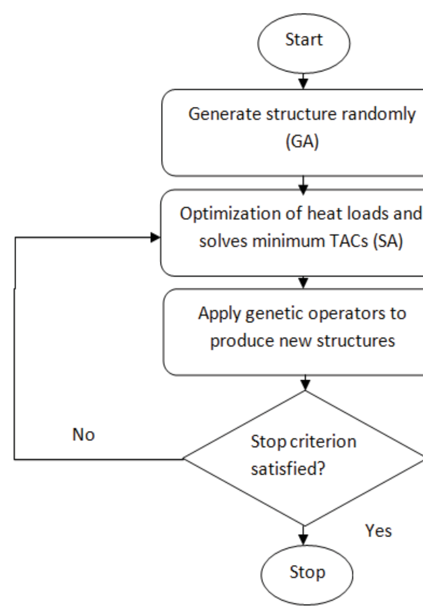


Figure 4: Flow diagram describing the structure of the two-level method

GA presents two probability model establish Monte Carlo [23] sampling respectively to create new structures (offspring), this is add or delete heat exchangers as random moves for a given structure. These two moves are equally probable to be performed when a candidate structure is generated. Genetics operations are taken into account in accordance with TAC. The better the TAC, the bigger the likelihood to be chosen to create the offspring. Roulette-wheel is the responsible of choosing the adequate structure that the offspring will have, once the parents are selected; only one genetic operation is applied. This operation is chosen randomly. There are two different possible operators, one-point crossover or mutation.

(i) One-point crossover to search the bigger solution space possible, combination of two structures selected by Roulette-Wheel is repeated to generate new offspring. Two parents, p1, p2 are combined by simple crossover to create two offspring, s1, s2, a half of the structure will go to one offspring and the other half to the other.

$$S1 = 0.5*p1 + 0.5*p2 \quad (28)$$

$$S2 = 0.5* p2 + 0.5*p2 \quad (29)$$

(ii) In addition to maintain the diversity of the population, the main purpose of this operator is to help prevent information loss in the evolution progress. Mutation of the parents is made as follow. Randomly an exchange of each parent is selected, and then these values are switched as follows.

$1 \rightarrow 0$

$0 \rightarrow 1$

Once the new structure is obtained, GA adds it to the population. Then, roulette-wheel updates the likelihood to choose parents. So, one child could be chosen in the next steps. The iteration ends when the second half of the structures is completely formed by offspring.

4 Cases and discussion

In order to verify the performance of our algorithm, the computation is conducted on a benchmark. The benchmark is composed of instances taken from the literature. The instances are organized according to the number of streams; they are sorted in ascending order.

4.1 Results

In this section, the instances of the benchmark are solved to test the performance of the presented two-level method. The results are compared to the best result obtained with a state-of-the-art algorithm from the literature. These results are summarized in Table 1.

The algorithm proposed is able to achieve five best results as done by others in the literature for HTN2, HTN5, HTN9, HTN21 and HTN22 problems.

Table 1: Summary results.

Method	Annual cost($/year)	Method	Annual cost($/year)	Method	Annual cost($/year)
HNT1		HNT11		HNT21	
Bjoerk2002	**76350**	Silva2010	**1624768**	Brandt2011	6110902
Huang2012	76327				
Huang2013	76327				
This work	76742.8	This work	1885667.4	This work	**5520273.5**
HNT2		HNT12		HNT22	
Bjoerk2002	52429	Bjoerk2002	**61295**	Agarwal2008	43728
				Huang2013	43359
This work	**47901.2**	This work	64891.4	This work	**36479**
HNT3		HNT13		HNT23	
Isafiade2008	97211	Isafiade2008	1150460	Petterson2008	**43331**
Ponce-Ortega2010	**97079**	Ponce-Ortega2010	1121175	Escobar2013	44081.4
		Huang2013	**1115868**		
This work	184798.7	This work	2152571.5	This work	43949.7
HNT4		HNT14		HNT24	
Bjoerk2002	**411746**	Bjoerk2002	96001	Wei2004	**43048**
		Huang2012	95643		
		Huang2013	**94742**		
		Huang2014	**94742**		
This work	434750.4	This work	126718.8	This work	44267.9
HNT5		HNT15		HNT25	
Escobar2013	470732.1	Petterson2008	**80962**	Khorasany2009	**5662366**
		Zamora1998	83400	Huang2012	5737274
				Huang2014	5733679
				Yerramsetty2008	5666756

Method	Annual cost($/year)	Method	Annual cost($/year)	Method	Annual cost($/year)
This work	**280451.4**	This work	1835556.8	This work	5673129.7
HNT6		HNT16		HNT26	
Isafiade2008	**311300**	Bjoerk2002	139083	Bjork2005	1530063.55
		Huang2012	128169	Escobar2013	**1524678.3**
		Huang2013	**123398**		
This work	570378.3	This work	288465.5	This work	1852969.9
HNT7		HNT17		HNT27	
Khorasany2009	**11895**	Isafiade2008	595100	Brandt2011	**6110902**
		Fieg2009	571698		
		Wei2004	571585		
		Toffolo2009	**570900**		
This work	14324.7	This work	593102.7	This work	71211663.5
HNT8		HNT18		HNT28	
Pettersson2008	84066	Khorasany2009	572476	Li2014	1805971
Zamora1998	87328	Huang2012	571657	Escobar2013	**1591070.1**
Yerramsetty2008	85972	Huang2013	**570362**	Huang2014	1937377
Toffolo2009	**82363**	Huang2014	612362		
Pariyani2006	85307				
This work	108245	This work	590016.4	This work	2620949.3
HNT9		HNT19			
Ponce-Ortega2010	385346	Isafiade2008	**168700**		
This work	**376176.5**	This work	174307.7		
HNT10		HNT20			
Ravagnani2005	117069.34	Chen2007	109765		
Chen2007	**109765**	Wei2004	**99524**		
		Huang2013	105403		
This work	154578	This work	109263.6		

5 Conclusions

A hybrid methodology for design and optimization of heat exchanger networks is presented. The HENs problem is solved by a two-level procedure, first GA is used to construct a HEN structure and then SA is employed to find optimum exchanger heat. Throughout the evolutionary process by the GA the structures of the individuals alter continuously. This is due to the genetic operations of structure crossover and mutation, respectively. In the lower level, the heat distribution of each candidate structure is optimized for minimum TAC by simulated annealing algorithm. The synthesis performance of this two-level method has been demonstrated using a benchmark. The assessed results indicate that the proposed algorithm is competitive with other forms of optimization algorithms. Combinations of heuristic based optimization methods for the efficient synthesis of HEN seem therefore very promising.

6 References

[1] Bodo Linnhoff and John R Flower. Synthesis of heat exchanger networks: I. systematic generation of energy optimal networks. AIChE Journal, 24(4):633–642, 1978.

[2] Bodo Linnhoff and Eric Hindmarsh. The pinch design method for heat exchanger networks. Chemical Engineering Science, 38(5):745–763, 1983.

[3] Uday V Shenoy. Heat exchanger network synthesis: The pinch technology-based approach.1995.

[4] Soterios A Papoulias and Ignacio E Grossmann. A structural optimization approach in process synthesis-ii: Heat recovery networks. Computers & Chemical Engineering, 7(6):707–721, 1983.

[5] Lorenz T Biegler, Ignacio E Grossmann, Arthur W Westerberg, and Zdravko Kravanja. Systematic methods of chemical process design, volume 796. Prentice Hall PTR Upper Saddle River, NJ, 1997.

[6] Christodoulos A Floudas, Amy R Ciric, and Ignacio E Grossmann. Automatic synthesis of optimum heat exchanger network configurations. AIChE Journal, 32(2):276–290, 1986.

[7] Terrence F Yee, Ignacio E Grossmann, and Zdravko Kravanja. Simultaneous optimization models for heat integration - i. area and energy targeting and modeling of multi-stream exchangers. Computers and chemical engineering, 14(10):1151–1164, 1990.

[8] JJJ Chen. Comments on improvements on a replacement for the logarithmic mean. Chemical Engineering Science, 42(10):2488–2489, 1987.

[9] WR Paterson. A replacement for the logarithmic mean. Chemical Engineering Science, 39(11):1635–1636, 1984.

[10] AR Ciric and CA Floudas. Heat exchanger network synthesis without decomposition. Computers & Chemical Engineering, 15(6):385–396, 1991.

[11] G Athier, P Floquet, L Pibouleau, and S Domenech. Optimization of heat exchanger networks by coupled simulated annealing and nlp procedures. Computers & chemical engineering, 20:S13–S18, 1996.

[12] B Lin and DC Miller. Solving heat exchanger network synthesis problems with tabu search. Computers & chemical engineering, 28(8):1451–1464, 2004.

[13] Daniel R Lewin, Hao Wang, and Ofir Shalev. A generalized method for hen synthesis using stochastic optimization–i. general framework and mer optimal synthesis. Computers & chemical engineering, 22(10):1503–1513, 1998.

[14] R.M. Khorasany, M. Fesanghary, A novel approach for synthesis of cost optimal heat exchanger networks, Comput. Chem. Eng. 33 (2009) 1363 e 1370.

[15] K.M. Yerramsetty, C.V.S. Murty, Synthesis of cost-optimal heat exchanger networks using differential evolution, Comput. Chem. Eng. 32 (2008) 1861 e 1876.

[16] A.P. Silva, M.A.S.S. Ravagnani, E.C. Biscaia, J.A. Caballero, Optimal heat exchanger network synthesis using particle swarm optimization, Optim. Eng. 11 (2010) 459 e 470.

[17] Z. Huo, L. Zhao, H. Yin, J. Ye, Simultaneous synthesis of structural-constrained heat exchanger networks with and without stream splits, Can. J. Chem. Eng. 91 (2013) 830 e 842.

[18] A.H. Masso, D.F. Rudd, The synthesis of system designs. II. Heuristic structuring, AIChE J. 15 (1969) 10 e 17.

[19] John H Holland. Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press, 1975.

[20] Scott Kirkpatrick, C Daniel Gelatt, Mario P Vecchi, et al. Optimization by simulated annealing. science, 220(4598):671–680, 1983.

[21] F. Schoen, Stochastic techniques for global optimization: a survey of recent advances, J. Glob. Optim. 1 (1991) 207 e 228.

[22] L. €Ozdamar, New simualted annealing algorithms for constrained optimization, Asia-Pac. J. Oper. Res. 27 (2010) 347 e 367.

[23] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller,Equation of state calculations by fast computing machines, J. Chem. Phys. 21 S. Kirkpatrick, (1953) 1087 e 1092.

[24] Biegler, L. T., Grossmann, I. E. & Westerberg, A. W. (1997). Systematic methods of chemical process design, p. 552). Prentice Hall.

A Semantic Repository Approach to Improve Health Information Management

Jose Gracia-Maldonado[1], **Francisco P. Romero**[2] **and Jose A. Olivas**2
[1]Indra-Software Labs, Indra-UCLM Research Center, Ciudad Real, Spain
[2]Dept. Information Technologies and Systems, University of Castilla La Mancha, Ciudad Real, Spain

Abstract— *Nowadays healthcare organization have to daily manage large amounts of information, an optimum use of this is increasingly difficult. This is especially significant when it comes to large and international contexts where the concepts and terminology handled in many cases, are not completely unified. The use of ontologies is widely presented as a solution to this problem because they allow the formal modeling of knowledge within a given domain. However, their implementation in real systems is not yet meaningful because they are not easy to use by non experts and it requires a great experience and training for proper understanding. In this paper we present ASMOR, a system to facilitate the task of organization, retrieval of key information in a healthcare context. The proposed prototype allows the user to monitor in a visually and fully automatic way the evolution of contents stored in a semantic repository.*

Keywords: Ontology, Healthcare Information Management, Semantic Repositories

1. Introduction

Ontologies have proven to be successful in handling a machine processable representation of information and have been used to model domain contexts in several real world applications [1], [2]. They allow capturing domain knowledge such as products, services, markets, etc. in an explicit and formal way such that it can be shared among human and computer systems [3]. An ontology can take the simple form of a taxonomy (i.e., knowledge encoded in a minimal hierarchical structure) or as a vocabulary with standardized machine interpretable terminology supplemented with natural language definitions. Ontology is often specified in a declarative form by using semantic markup languages such as RDF and OWL [4].

The notion of ontology is becoming very useful in various fields such as intelligent information extraction and retrieval, cooperative information systems, electronic commerce, and knowledge management [5]. In nowadays, it is important to deal with the concept of Information Filtering, through which a subset of relevant documents is quickly and effectively selected for further detailed analysis. Information Filtering has emerged with the aim of dynamically adapting the distribution of information where both evolving user's interests and new incoming information are taken into account. Although information retrieval and information filtering share the goal of selecting documents whose content matches a particular information need, they are different in some aspects. Information retrieval systems are designed for supporting users with short term interests (query), and retrieval refers to information selection from relatively static information collections or databases. In contrast, filtering refers to information selection from a dynamic stream of incoming data on the basis of user's long term interests (profile). As a result of their common goal, the techniques applied to information retrieval have been also applied to information filtering task [6].

Although ontologies are used in a wide range of applications and have been instrumental in many interoperability projects, they have so far had only limited success [7]. Early semantic search engines tried to use ontology concepts and structures as controlled search vocabularies, but this was unpractical both functionally and from a usability perspective. Another bottleneck in developing ontology-based systems stems from the fact that the conceptual formalism that supports by common ontologies may not be sufficient to represent uncertainty that is commonly found in many application domains [8]. Moreover, most of ontologies are application-specific and its construction can be a long and costly task. The use of conceptual structures to provide users access useful information requires that there is something bridging the gap between the conceptual and the real world.

In this paper, the tool **ASMOR** (**A**lert **M**anagement System for **O**ntologies **R**epositories) is presented. This tool is a prototype that allows the user to perform a dynamic management of the contents stored in a complex Information System in an easy and intuitive way. The presented work differs from previous approaches that exploit ontologies and SPARQL Queries in a transparent way to the user. The main features of this tool are: (i) Useful and usable in daily use, (ii) Intelligent inbox management, (iii) Visual Interaction (alerts and results graphs) (iv) Flexibility and modularity (it can be easily applied to several domain fields).

The rest of the paper is organized as follows. Section 2 describe the main components of the tool, meanwhile, in section 3, the high-level architectural design of ASMOR and its key functionalities are presented. Section 4 describes some demonstration scenarios, and finally, some conclusions and future works are pointed out in section 5.

2. Components

The use of a publish/subscribe (pub/sub) system can be considered relevant to carry out an active management of semantic repositories. A pub/sub receives information from data sources and notifies users if the messages match the subscriptions criteria. Typically, the publish/subscribe system contains three roles: (i) event providers (ii) event consumers, who subscribe events that they interested, and (iii) the event broker, who is responsible for routing the events from the publishers to subscribers [9]. In our context the event providers are the sources of health information about patients, the event consumers are the professionals, and ASMOR is a tool that allows the user to do an automatic, intuitive and visual event broker with the aim of monitoring of the evolution of the contents stored in the repository.

Two main components are the basis of ASMOR tool: the alerts and the ontological model.

2.1 Alerts

An alert is a time saving tool that makes the user aware that a new information of interest has been made available, eliminating the need for constant monitoring of the information repository. In this context, alerts are useful to help the user to keep up-to-date and stay aware of changes in your patients or on new patients in a certain conditions. Automated alerts have recently emerged as a major instrument to influence clinician behavior. In the hospital setting, randomized trials have shown the efficacy of alerts [10].

The alerts systems is feed using the knowledge stored in a semantic repository. An alert means that the underlying data is changing, thus, ASMOR is not only a tool to evaluate the existing data set to find alerting case, it is a tool to monitorize the changes in the contents of the data repository over the time. This process is carried out by defining a set of alerts that will be reported to the user according to their priority.

There are a few different components that make up an alert definition, they include:

- *Priority*: The priority is used for ordering notifications of different types for the same user.
- *Criteria*: The filtering criteria are focused on: entities, entities relationship, and entities attributes.
- *Graphical Representation*: ASMOR offers a user-friendly interface to assist users in the definition of their criteria in a graphical and intuitive way.
- *Query*: ASMOR translates the specified criteria to a SPARQL query. This process is transparent to the user.
- *Recommendations/actions*: It is possible to define actions to carried out when the notification is generated.

Alert descriptions are usually defined by the Semantic Repository Administrator, but the user also have the capacity to create their own alert definitions, that includes building their own criteria, prescribing their own recommendations, and assigning actions.

2.2 Ontological Model

Due to the diversity of such information sources and the increasing amounts of data produced nowadays, it is necessary to build a semantic model that describes the interchanged knowledge in the information system. The model is useful for the system to discover contents and users correlation.

Although there is not a universal consensus on the definition of ontology, it is generally accepted that ontology is an explicit specification of a conceptualization [11]. A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose. Every knowledge base, knowledge-based system, or knowledge-level agent is committed to some conceptualization, explicitly or implicitly.

Ontology provides a number of potential benefits in representing and processing knowledge, including the separation of domain knowledge from application knowledge, sharing of common knowledge of subjects among human and computers, and the reuse of domain knowledge for a variety of applications [12]. There are several ontologies found in the literature created for it use in a particular knowledge domain.

In this work, the proof on concept of the proposed tool has been made using the healthcare information expressed on the CCR (Continuity of Health Record) standard [13]. The Continuity of Care Record standard contains information such as patient demographics, insurance and health care provider information, medication lists, allergies and recent medical procedures [14].

The proposed model has been developed in Web Ontology Language (OWL) and is an updated version of the one presented in [15]. Several studies has been demonstrated the utility of this model for semantic interoperability [16]. An synthetic of the proposed ontological model can be seen in Figure 1.

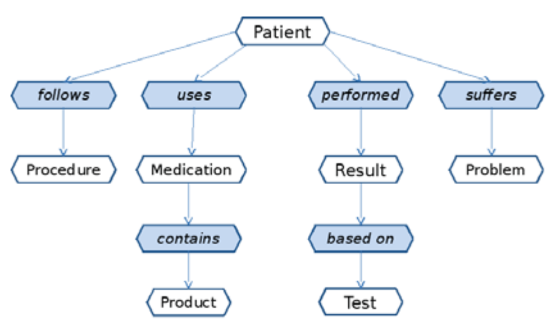


Fig. 1: Synthetic view of the Ontological Model

The modeling process has been carried out using Protege [1] and the repository has been implemented using Sesame OpenRDF [2].

[1] http://protege.stanford.edu/
[2] http://www.openrdf.org/index.jsp

3. Architecture

A global overview of the context and environment of ASMOR tool can be seen in Figure 2. The identified flows of information in the system are the following:

1) Healthcare data is generated at many points in a healthcare organization. Much of the data generated is patient oriented. This data is integrated in a CCR format which is the format in which health information is stored and exchanged across systems.
2) Information from health centers is translated to a RDF format and stored in the semantic repository.
3) The more relevant alerts to the healthcare organization are defined by the administrator.
4) Notifications will be collected automatically in the user's inbox. They were generated by the system when the criteria were verified.

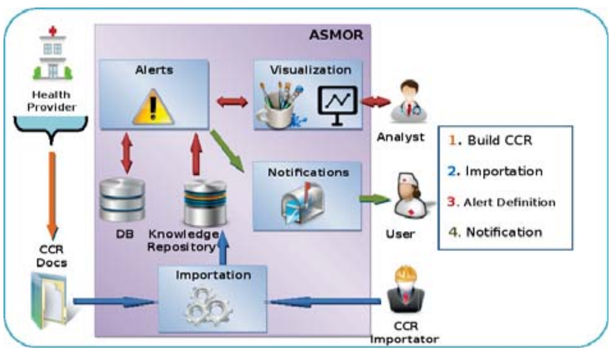


Fig. 2: Global View of ASMOR tool

The monitoring process is carried out by means the user-defined set of alerts. These alerts are reported to the user's inbox according theirs contents and their priority. For each alert, the user can define, in a completely visual way, a set of criteria that, when one is verified, it will generate and automatic end user notification. After this notification the users can interact visually with the content that triggers the alert.

ASMOR has been built according to a modular architecture that comprises two key modules acquisition and exploitation as is detailed in the the following subsections (Figure 3).

3.1 Acquisition

The ontological repository is able to integrate and merge information from different sources in a heterogenous format. Therefore, the content provided by the information sources is transformed to RDF triplets in order to be stored in the semantic repository.

Fig. 3: Asmor Architecture

This procedure is carried out in a three-step process(Figure 4):

1) Knowledge instances are extracted from the documents in a specific format.
2) RDF representation of these knowledge instances.
3) Storing the information in the ontological repository.

Fig. 4: Acquisition Process

The transformation rules that are used in the CCR to RDF/OWL algorithm are the following:

- Complete elements are transformed into OWL classes.
- Simple elements are transformed into data properties.
- The complex relationships between elements are transformed into class-class relationships, that is, in OWL object properties (see Table 1).

Table 1: OWL object properties.

Property	Source	Target
Bases_on	Result	Test
Contains	Medication	Product
Follows	Patient	Procedure
Performed	Patient	Result
Uses	Patient	Medication
Suffers	Patient	Problem

In compliance with the transformation rules explained above, the second part of the process takes care of transforming CCR to RDF. For this purpose, it is necessary to transform XML data to RDF/OWL by creating instances of the necessary OWL classes, RDFS datatypes, OWL datatype and object properties.

3.2 Exploitation

The knowledge exploitation has made by three modules: alert, evaluation and notification.

3.2.1 Alert Module

In terms of volume, a Healthcare Knowledge Base (KBs) could contains millions of entities and facts about patients, problems and procedures . In this case, such information is stored in RDF format and queried with the SPARQL language. Large KBs are difficult to use as their schema (the ontological model) and its underlying semantics are rarely understood by end users.

ASMOR proposes a graphical interface to construct a SPARQL query used to monitoring the content evolution in a RDF repository. The "Alert Module" allows the user to define alerts in a completely visual way. The visualization component, based on JIT (Javascript InfoVisualization Toolkit [3]) framework, has been used for this purpose. The browser panel displays the ontology as a graph. As an ontology can be large, a search box with auto completion feature is available for finding classes and properties that will be used in the query. When a class or property is selected, the graph is centered around this concept to see all the related concepts. The visual construction of a query is composed of three main steps (similar to the proposed in [17]):

1) Selecting classes and/or properties from the semantic model browser into the query panel.
2) Linking the classes and properties defined in the previous step.
3) Adding query operators by selecting the components of the query on which they must be applied.

An example of alert in a medical ontological repository can be seen in Figure 5.

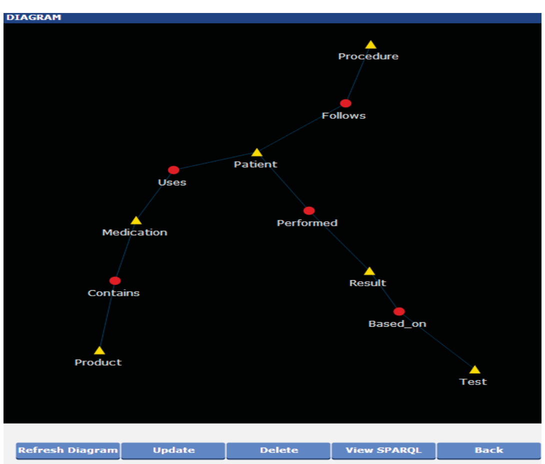


Fig. 5: Example of alert definition

[3]http://philogb.github.io/jit/

Each alert is internally translated to a SPARQL query as Figure 6 shown.

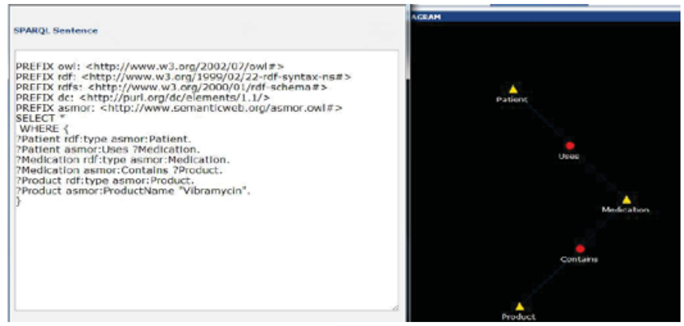


Fig. 6: SPARQL Query

3.2.2 Evaluation Module

The evaluation module consists of those features that allow ASMOR system evaluating the active alert criteria in a given period, in a completely automatic way. The module provides the necessary algorithms for the analysis and evaluation of every alert defined in the system. The tool provides a mechanism to automatically run the analysis algorithms in the background. The management of the results generated by the SPARQL queries is also part of it.

The tool evaluates only active alert criteria that are also associated with active alerts in the system. These alert criteria are executed automatically and evaluated on the semantic repository. If a criterion is satisfied based on the stored information, an action to be performed is determined. The event dispatcher is the component responsible for collecting definitions and forwarding alert to the users. The architecture of the event dispatcher in this tool is centralized.

3.2.3 Notification Module

The funcionality of the notification module consists of managing and visualizing the different "inboxes" which receive alert notifications. An example is shown in Figure 7, the notification "inbox" can be observed with two notifications that are pending to be examined.

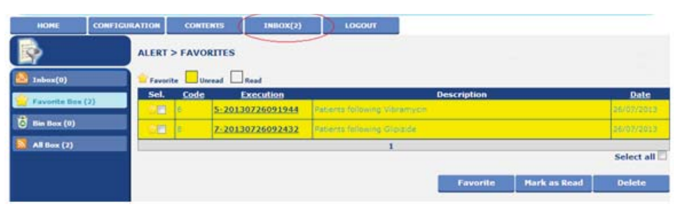


Fig. 7: Example of User Notification

One the notification is read, the tool display a graph with the results that have been activated according to the alert criteria. The user can visualize and interact with these results (Figure 8).

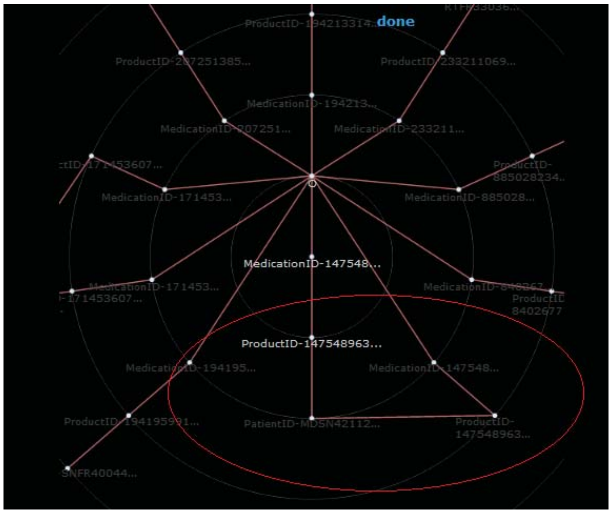


Fig. 8: Example of Result

4. Demonstration Scenarios

Three scenarios has been defined to demonstrate the feasibility of the proposed tool (Figure 9). This demonstration will include groups of cases with the aim of showing all the capabilities of the system in three different contexts:

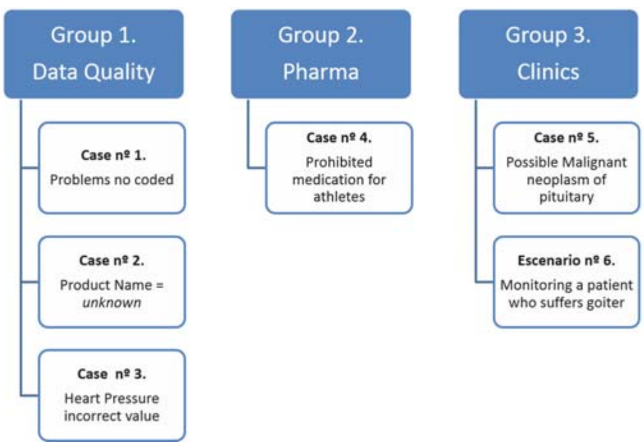


Fig. 9: Considered Cases

- *Data Quality*: This scenario illustrates the use of the tool for detecting erroneous or inconsistent data. The type of user involved in this scenario is the *Health Documentalist*. This user is responsible for ensuring the quality of data managed by the e-Health systems. In this scenario, the user can monitorize the quality of the data that would feed the semantic repository.
- *Pharma*: This scenario illustrates the use of the tool for detecting medications that containing products considered illegal or whose use must be under control. The role involved in this scenario is the *Pharmacist*, i.e., the responsible for ensuring that medication to be supplied to patient is under current legislation.
- *Clinics*: This scenario illustrates the use of the tool to define associated alerts to a number of symptoms that have high probability of being evidence of serious illness. The user involved in this scenario is the *Doctor*. This user is responsible for analyze the possibility of having a serious illness from severe detected symptoms. For example, those patients whose medical history concur problems such *Hypercalcemia* and *Cyst and pseudocyst of pancreas* has a high risk of the Wermer syndrome (also known MEM Type 1). These cases would have to request a Nuclear Magnetic Resonance (NMR) to rule out a pituitary tumor.

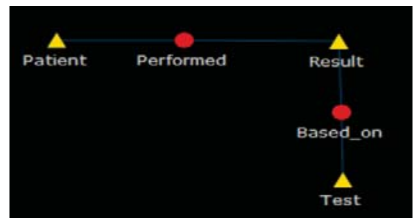


Fig. 10: Example of Alert

The goal of the demo is to show how ASMOR can be used to manage healthcare information in a very efficient manner. In absence of real world data for evaluating the tools, real anonimyzed data has been used during the testing process.

During the demonstration, the focus will be on showing the details of the process involved, starting from alert definition to the visualization of the notification results. During alert definition, the graphical user interface will assist users to issue or refine the alert query by adding, editing, or deleting semantic links among entities. Upon retrieval of the notification related data, users will view the graphical representation of the results. Also, they could be able to organized them in a number of different ways.

5. Conclusions and Future Work

A prototype of semantic repository management tool based on a graphical way for query management and results visualization is presented. The proposed tool, ASMOR, is an example of advanced decision support system based on semantic models. The basic idea is to allow the user to define alerts by identifying entities and values which are detected together and to derive a notification if they occur in the repository.

The system core is characterized by the fact that the graphically obtained SPARQL query is submitted to a ontological repository that stores information from heterogeneous data sources.

The prototype has been developed based on open source standards and following a modular design and architecture, therefore, it has given him the flexibility to be perfectly used for any ontology. This tool can be used as an alert and notification system in other areas of expertise such as e-learning if they have a related ontological model like the proposed in [18].

The prototype has been tested in different scenarios within the field of healthcare in collaboration with medical professionals. After these tests, it can be concluded that all the system features were accepted. Therefore, it has demonstrated the feasibility of building a tool that allows users to monitor visual, intuitive and automatic evolution of those contents represented in an ontological model for proper management of them on a active way.

ASMOR features are being extended to support a machine learning module to automatically define complex intelligent rules for content management.

References

[1] L. Han and G. Chen, "A fuzzy clustering method of construction of ontology-based user profiles," *Advances in Engineering Software*, vol. 40, no. 7, pp. 535–540, 2009.

[2] M. Golemati, A. Katifori, C. Vassilakis, G. Lepouras, and C. Halatsis, "Creating an ontology for the user profile: Method and applications," in *In Proceedings of the First International Conference on Research Challenges in Information Science (RCIS)*, 2007.

[3] R. Lau, "Fuzzy Domain Ontology Discovery for Business Knowledge Management," *IEEE Intelligent Informatics Bulletin*, vol. 8, no. 1, pp. 29–41, 2007.

[4] D. L. McGuinness and F. van Harmelen, "Owl web ontology language overview," W3C, Tech. Rep. REC-owl-features-20040210, 2004.

[5] A. Sieg, B. Mobasher, and R. Burke, "Web search personalization with ontological user profiles," in *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. New York, NY, USA: ACM, 2007, pp. 525–534.

[6] R. Kassab and J.-C. Lamirel, "Towards a synthetic analysis of user's information need for more effective personalized filtering services," in *SAC '07: Proceedings of the 2007 ACM symposium on Applied computing*. New York, NY, USA: ACM, 2007, pp. 852–859.

[7] G. Solskinnsbakk and J. A. Gulla, "Combining ontological profiles with context in information retrieval," *Data Knowl. Eng.*, vol. 69, no. 3, pp. 251–260, 2010.

[8] Q. T. Tho, S. C. Hui, and T. H. Cao, "FOGA: A Fuzzy Ontology Generation Framework for Scholarly Semantic Web," in *Proceedings of the 2004 Knowledge Discovery and Ontologies Workshop (KDO 2004)*.

[9] F. Tian, B. Reinwald, H. Pirahesh, T. Mayr, and J. Myllymaki, "Implementing a scalable xml publish/subscribe system using relational database systems," in *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '04. New York, NY, USA: ACM, 2004, pp. 479–490.

[10] "The use of patient and physician reminders for preventive services: results from a national study of physician organizations," *Preventive Medicine*, vol. 39, no. 5, pp. 1000 – 1006.

[11] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquis.*, vol. 5, no. 2, pp. 199–220, 1993.

[12] K. Dramé, G. Diallo, F. Delva, J. F. Dartigues, E. Mouillet, R. Salamon, and F. Mougin, "Reuse of termino-ontological resources and text corpora for building a multilingual domain ontology: An application to alzheimer's disease," *Journal of Biomedical Informatics*, vol. 48, no. 0, pp. 171 – 182, 2014.

[13] M. Botsivaly, B. Spyropoulos, K. Koutsourakis, and K. Mertika, "Enhancing continuity in care: An implemantation of the ASTM E2369-05 standard specification for continuity of care record in a homecare application," in *AMIA 2006, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 11-15, 2006*. AMIA, 2006.

[14] P. Plastiras, D. O'Sullivan, and P. Weller, "An ontology-driven information model for interoperability of personal and electronic health records," in *Proceedings of eTELEMED, The Sixth International Conference on eHealth, Telemedicine, and Social Medicine*, March 2014, pp. 130–133.

[15] J. Puustjärvi and L. Puustjärvi, "Personal health ontology: towards the interoperation of e-health tools," *IJEH*, vol. 6, no. 1, pp. 62–75, 2011.

[16] ——, "Using ontology-based registry and SPARQL engine in searching patient's clinical documents," in *HEALTHINF 2014 - Proceedings of the International Conference on Health Informatics, ESEO, Angers, Loire Valley, France, 3-6 March, 2014*, 2014, pp. 151–158.

[17] G. Fokou, S. Jean, A. Hadjali, and M. Baron, "Qars: A user-friendly graphical tool for semantic query design and relaxation," in *Proceedings of the 18th International Conference on Extending Database Technology, EDBT 2015, Brussels, Belgium, March 23-27, 2015.*, G. Alonso, F. Geerts, L. Popa, P. Barceló, J. Teubner, M. Ugarte, J. V. den Bussche, and J. Paredaens, Eds. OpenProceedings.org, 2015, pp. 553–556.

[18] C. Vidal-Castro and M. F. Satler, "Modelo ontológico para la secuenciación de objetos de aprendizaje," *IEEE-RITA*, vol. 5, no. 2, pp. 63–67, 2010.

A decision support system for risk analysis and diagnosis of hereditary cancer

César Calatrava*, Mauro J. Oruezabal†, Jose A. Olivas*, Francisco P. Romero* and Jesus Serrano-Guerrero*
*High School of Computer Engineering
University of Castilla La Mancha, Ciudad Real, Spain
†University Hospital Rey Juan Carlos, Madrid, Spain

Abstract**—**The complexity associated with the process of the capture of medical decisions resides in the managing of considerable information deriving sometimes in a series of clinical mistakes. Due to this, the clinical decision support systems are receiving a major importance nowadays in the context of medicine. The family identification with a high risk of suffering hereditary cancer is an arduous labor. The aim of the present work is the improvement of the process of identification of suspicious families that can suffer cancer of genetic character with the development of a decision support system for risk analysis and diagnosis of hereditary cancer.

Keywords**—**Clinical decision support system, cancer, genogram, genetic factors, probandus

I. Introduction

In recent years, the presence of the computer science has increased considerably in the medical area, up to the point that we can find from tools in charge of the documentary medical management up to clinical decision support system for the decision-making, facilitating of notable form the work of the sanitary personnel.

The main discipline responsible for this unity is Knowledge Engineering [1], whose main purpose is the acquisition, representation and conceptualization of knowledge. All this is orientated to emulate various intelligent human capabilities.

Inside the field of the Engineering of the Knowledge Decision Support Systems (DSS) stand out [2] [3], which try to simulate the knowledge of a person expert in a domain of specific knowledge. These systems arise in order to overcome the limitations of other IT systems popularized previously or to give response and coverage to situations not solved up to the date.

The complexity associated with the process of capture of medical decisions resides in the managing of considerable information deriving sometimes in a series of clinical mistakes, it is shown in some reports [4], which states that these might be the eighth reason of death in the industrialized countries. Due to this, the decision support systems are receiving a major importance nowadays in the context of medicine, receiving the name of Clinical Decision Support Systems (CDSS) [5].

We can define a CDSS as a knowledge system designed to help health professionals to take preventive clinical decisions or diagnostic based on a series of data of the patients. The implication of the patients in the capture of decisions is increasing in the last years, for what the design of the CDSS is orientated increasingly to the integration of his preferences.

The health, nowadays, is a topic of great scope in the society, and the cancer one of his more clear exponents, due to the scanty percentage of patients who manage to overcome successfully the disease. The term cancer [6] includes a numerous group of diseases characterized by the proliferation of abnormal cells, which divide and grow without control being able to invade other fabrics of the body.

A high percentage of the patients that suffers some type of cancer does it in sporadic form, namely, there does not exist any genetic risk [7] of suffering the disease. Nevertheless, there exists a small percentage of patients (5%-10%) that present a tumor hereditary syndrome and that is due to genetic factors that the individual carries from his birth. Accordingly, the fact of inheriting a genetic susceptibility to developing the disease does not want to say that it will end up by developing it, but the risk to suffering it is significantly superior to the rest of the population.

The family identification with a high risk of suffering hereditary cancer is an arduous labor. From the clinical point of view, the identification of a series of characteristics that must put in alert the medical personnel should be important. Some of these characteristics can be the high incident of cancer in the family, the appearance of the cancer at an early age, the occurrence of the same type of cancer, among others.

The aim of the present work is the improvement in the process of identification of suspicious families that can suffer cancer of genetic character, first, with the automatic construction of the genogram [8] (genealogical clinical tree) of the patient based on his health record, and secondly with the evaluation of this genogram, later, to recommend a series of actions to continue or to give advices according to the diagnosis obtained. This will be achieved by the development of a decision support system for risk analysis and diagnosis of hereditary cancer.

The rest of the document has been organized in the following form: section II gathers the previous investigation carried out in the area of interest, paying special attention to the genogram and the standard *Continuity of Care Record* (CCR) as a model of health record, section III describes in detail the parts that shape the system as well as its functioning, in the next section, part IV, the design of an experiment on the system functioning is shown. Finally section V contains the final conclusions and future works of the system exposed in the present work.

II. Related Work

In the process of identification of families suspicious of suffering hereditary cancer there arises the need to compile a series of information both of the patient and of the members of his family to carry out the above mentioned process. This need is covered with a genealogical clinical tree called genogram. A genogram gathers graphically (in the shape of a genealogical tree) the basic information of, at least, three generations of a family. In this case, for the system that exposes in the present document the necessary information of each of the familiar members, included the *probandus*[1] it is the following one: age, sex, presence of tumor or not, type of tumor, in case of existing, age of diagnosis, age of death and reasons of the death.

Diverse tools exist for the construction of genograms, among which we find *GenoPro* [2], this genealogical tool offers us a practical solution at the moment of creating familiar trees and genograms. In addition, it allows the storage of additional information as images and other resources.

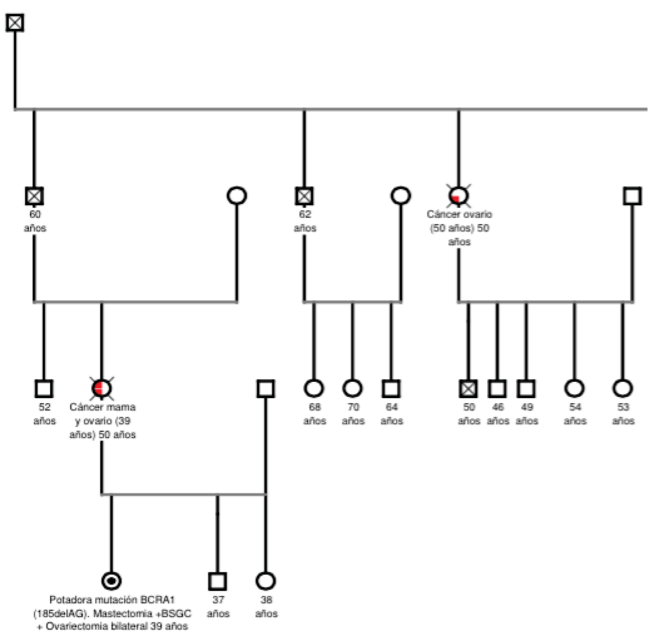


Figure 1. Example of genogram in *GenoPro*

However, GenoPro, which allows us the construction of genogramas to carry out the identification and evaluation of the suspicious families of genetic cancer is not the tool most adapted for this, principally due to the fact that it does not allow the automatic generation of the genograms based on the health record of the patient. On the contrary, it only allows the manual construction and this can lead in some occasions to mistakes. As it was commented previously in the introduction, one of the principal innovations that introduces this system is the automatic construction of the genogram based on the health record of the patient.

[1]an individual affected with a disorder who is the first subject in a study (as of a genetic character in a family lineage)

[2]http://www.genopro.com/es/, (Last visit - April 22, 2015)

Therefore, it is important to have a robust and flexible model of health record to facilitate the task of construction. The model of health record chosen has been the standard *Continuity of Care Record (CCR)* [9], which is a specification of the medical records developed by *AST International* and a set of institutions and companies related to the sanitary sector.

The principal aim of this standard is the improvement in the record of the medical information of the patients. Instead of sheltering a complete record of the patient, the CCR contains the most relevant and important information of the patient allowing this way the exchange between the sanitary personnel. The CCR is designed so that a sanitary expert can create it easily across one electronic health record.

In order to cope with the model of a health record needed by the system, it has been necessary to do an extension of the standard CCR, on the basis of the foregoing, the following objects have been added to the model CCR: HealthStatusActor, Tumor and FamilyTumorHistory.

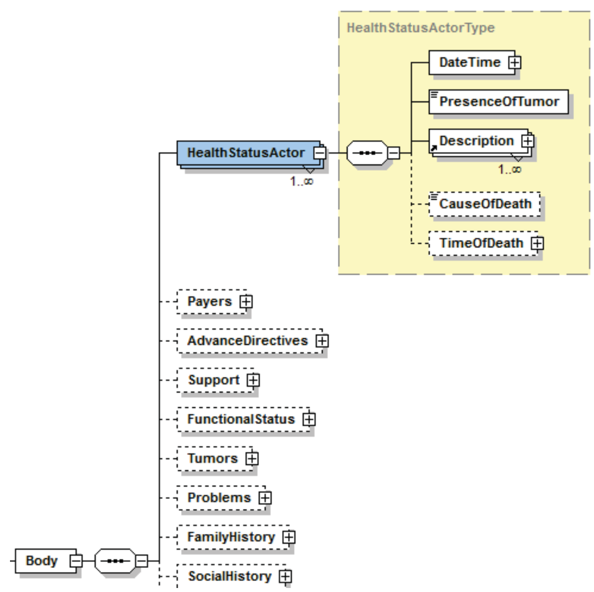


Figure 2. Extension of the *HealthStatusActor* object in CCR

- HealthStatusActor: this object has been added to the model to be able to face the requirement of the expert of indicating the age of the actor (*probandus* or family) and the presence or not of a tumor. For *probandus* will be referenced from *Body* and for the rest of relatives from *FamilyMember*.
- Tumor: the motive of his incorporation has been being able to gather in a precise and correct form the information of the tumor necessary for the correct construction of the genogram. This object is also going to contain two fields for the tumors with typical phenotype and the benign tumors with predisposition to turn in malignant.
- FamilyTumorHistory: object added to *FamilyHistory* to handle the familiar precedents affected by cancer that existed in the family and the healthy ones.

III. Proposal

The principal aim of the system is the improvement in the process of identification of the families that present a high risk of suffering hereditary cancer, as well as of their characteristics or those of a concrete relative for later to value and to recommend the actions to continue according to the evaluation carried out.

As for the structure of the system, we differentiate three parts, on the one hand, the module for the automatic generation of the genogram of a patient based on his health record, and for other one, the part that deals with the analysis and evaluation of this genogram to check the risk the *probandus* or some concrete relative for syndrome of predisposition to the cancer (SPC). Finally, but not less important, a manager of clinical histories for the management of the same ones completes the structure of the CDSS.

Next, each of these modules will be exposed in a detailed form.

With the module of generation of genograms what is looked is to construct a tool capable of generating the complete genogram of a patient of automatic form based on his clinical history. The principal advantage of this tool is that on having treated with health histories that follow a medical standard as the CCR, it allows the possibility of generating genograms of diverse nature, not only orientated to the oncology area as in this case. Other advantages are the considerable profit of time opposite to the manual construction as well as the decrease of the risk to committing mistakes in its creation.

```
▼<Code>
   <Value>394856008</Value>
   <CodingSystem>SNOMEDCT</CodingSystem>
   <Version>2014</Version>
 </Code>
</ActorRole>
▼<HealthStatusActor>
   <PresenceOfTumor>No</PresenceOfTumor>
```

Figure 3. Fragment of health record in *CCR*

Departing from a health history any one, figure 3, and after a process of transformation which we can divide in two phases, the first, in charge of the extraction of the elements to representing of the clinical record, and the second, which will carry out the graphical representation of these elements, we will obtain in this way the complete genogram of a patient similar to that of the figure 4. The principal advantage that presents representation is that every type of cancer has an own symbol associated. For this reason, the doctor only with a brief glimpse will be able to identify the types of cancer that suffers a family. For example, the bilateral breast cancer is represented with a circle of red color.

As shown in figure 4, of the healthy nodes their current age is registered and of the nodes affected by some type of cancer the age of the diagnosis and the current age is shown. The cases that have deceased are marked by a cross and their age of death is reflected in the genogram, if the reason of the death was due to some tumor will be reflected in the genograma, in opposite case it is not.

On the other hand, the module in charge of the analysis and evaluation of the genogram is formed by two submodules, one will take the previous evaluation of the genogram as a principal function with the aim to generate a preliminary report that contains a series of useful indicators for the doctor. The another submodule has most of the weight of the CDSS, it will evaluate and analyze the genogram of exhaustive form to determine if the family presents a high risk of suffering hereditary cancer.

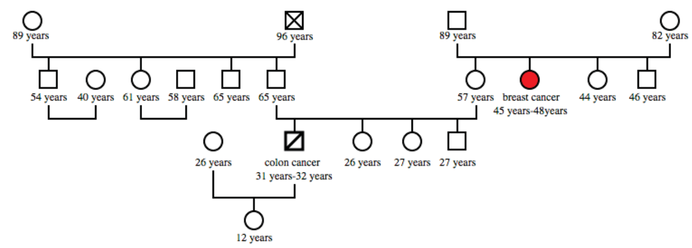


Figure 4. Genogram model obtained after the generation

The purpose of realizing a previous analysis of the genogram is orientated to the use of the system on the part of doctors of primary care, that is to say, doctors not expert in the field of the oncology and who do not present the sufficient knowledge on this speciality. Therefore, this analysis comes to cover the above mentioned lack of knowledge with the generation of a report, with this first analysis the following indicators will be obtained:

- Classification of the genogram in informative or not informative: the genograms can be classified in informative or not informative according to the characteristics that they present. A genogram is informative for the following cases:
  - Hereditary Breast Ovarian Cancer Syndrome (HBOC): existence of at least two women over 65 years in the branch where there are the cases that the syndrome defines.
  - Other syndromes: Existence of at least three persons (man or woman) over 65 years,

  the persons who are exposed above can have or not cancer. The rest of genograms that do not fulfill these criteria will be classified like not informative.

- Associate cases and his relation with syndromes of predisposition to the cancer: correspond to the nodes affected by cancer and that due to the criteria that they present can be classified in someone of the syndromes of predisposition to the cancer as the syndrome of Li-Fraumeni or Peutz-Jeghers.

- Sporadic cases: identification of sporadic cases in the genogram, nodes affected by cancer, but the origin of this cancer does not have any relation with genetic factors, with the syndromes of predisposition to the cancer. His identification is very important in order that the doctor does not confuse them with nodes affected by hereditary cancer.

- Index case and cases not indexed: search and identification inside the genogram of the case index and the cases not indexed. In a genogram the index node is that one that either presents the strangest tumor, for example, a triple negative breast cancer only affects 10% of the population, or it is the youngest node affected by some type of tumor. The rest of nodes of the genogram correspond to cases not indexed.
- Phenocopies: they correspond to the cases of sporadic cancer in a family with a syndrome of hereditary cancer. Seemingly (for the phenotype) they are similar cases affected by syndrome of genetic cancer, but do not have the mutation (different genotype). Therefore, they receive the name of phenocopy the cases that have a similar phenotype, but a different genotype.

In the figure 5, we can see the knowledge map of this module.

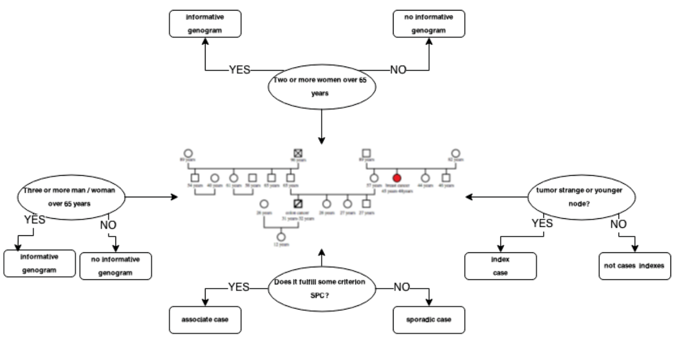


Figure 5. Knowledge map

On the other hand, the submodule in charge of the exhaustive analysis of the genograma is much more complex than the previous one that we have seen, this is due to the dimension of the evaluation that it has to make. It will realize a meticulous analysis of the genograma to verify the characteristics of each one of the nodes that shape the genogram.

It will be contemplated the diagnosis of three types of cancer such as breast cancer, ovary and colon, in addition to various syndromes related to the genetic tumors. On having finished the evaluation, we will obtain a report with the analysis realized on the genogram similar to that of the figure 7 of the next section.

Next, the criteria extracted in the meetings with the expert from one of the tumors included in the system, the cancer of breast and ovary, are exposed:

- *High-risk families*[3]
  - A case of cancer of breast to minor or equal age of 40 years.
  - Cancer of breast and cancer of ovary in the same patient, to any age.
  - Two or more cases of cancer of breast, one of them diagnosed to a minor or equal age of 50 years, or bilateral.
  - A case of cancer of breast diagnosed to a minor or equal age of 50 years or bilateral, and a cancer of ovary in a relative of the first or second degree.
  - Three cases of cancer of breast or ovary (at least one of ovary), in relatives of the first or second degree.
  - Two cases of cancer of ovary in relatives of the first[4] or second[5] degree.
  - A case of cancer of breast in male, and another case of cancer of breast (male or woman) or ovary in a relative of the first or second degree.
- *Families of moderate risk*[6]:
  - Two cancers of breast in relatives of the first degree, diagnosed between 51 and 60 years.
  - A cancer of breast in a relative of the first degree and other one in a relative of the second degree, if the sum of the ages to the diagnosis is minor or equal 118 years.

To conclude, with the development of the module of clinical management one has sought to implement a tool capable of automating to the maximum everything the process of management of the medical reports of the patients. Those clinic records manager takes as basic functions the storage and recovery of the records, beside having a small clinical seeker.

The clinical histories are loaded in a documentary warehouse in order to facilitate the access and the recovery of the same ones at any time determined. The user of the only system must worry about the selection of the history that it wants to load, so the clinic records manager will take charge storing it in his corresponding place. It is important to indicate that the above mentioned manager offers us the possibility of creating collections, the collections allow to group different document that share a series of similar characteristics. For example, if in the system we have diverse of clinical histories corresponding to a series of relatives we might create a collection for this family and hereby be able to guard his medical records of form more organized under the same "group".

Normally, as soon as the clinical histories are stored his modification or update will not be necessary. Therefore, the activity of recovery is focused on the most part the consultation of the clinical histories. The doctor will be able to accede to them for the consultation of any aspect of his interest.

Finally, the clinical seeker will be of great usefulness at the moment of realizing searches on one or several clinical histories with the aim to obtain a series of statistics or useful indicators. This seeker will cover different cases of consultation since it can be this, so the doctor will get to know the total number of affected women by cancer of breast minors 30-year-old, between others.

[3]The families of high risk are candidatas to: 1) consultation of genetic advice; 2) analysis of the genes BRCA1 and BRCA2, and 3) measures of follow-up.

[4]Any individual who is separated by a meiosis from one from the members of his family (it is to say, father / mother, brother / sister, son / daughter)

[5]Any individual who is separated by two meiosis from one of the members from his family; relative with whom an individual shares the fourth part of his genes (it is to say, grandparents, grandsons, uncle, aunt, nephew, niece)

[6]The families of moderate risk can benefit from a consultation of genetic advice, and in advisable measures of follow-up of the organs reveille beyond applied in the general population

IV. Experiment

Once the functioning of CDSS has been described accurately and detailed. We will carry out the design of an experiment where will develop each of the involved steps in the process, from the generation of the genogram up for his evaluation and analysis.

As it was mentioned before in section III, the responsible module for the automatic generation of the genogram will take the health record of the patient as entry, stored in the clinical repository, for further processing. A genogram similar to that of the figure 4 will be obtained. This can be found in the previous section.

The next step in the process of analysis is the evaluation of the genogram for the generation of a preliminary report it contains a set of indicators as indicated previously. Next, figure 6, we show the result of this assessment on a particular genogram.

```
Análisis previo del genograma 23as54ju9032sd2890

                REPORTE PRELIMINAR

- CLASIFICACIÓN DEL GENOGRAMA: Genograma no informativo.

- CASOS ASOCIADOS: Nodo I.1, Nodo II.1, Nodo II.4 y Nodo III.2

- CASOS ESPORÁDICOS: No existen evidencias de ello debido a la edad de aparición del
cáncer en el Nodo I.1.

- CASO ÍNDICE: Nodo III.3 debido a criterio nodo más joven en desarrollar cáncer.

- CASOS NO ÍNDICE: Resto nodos que conforman el genograma.

- FENOCOPIAS: No existen evidencias de ello.
```

Figure 6. Preliminary report after the evaluation of the genogram

With this report the doctor would know quickly and easily the general characteristics of the genogram, likewise, it can use to corroborate or refute his diagnosis on the analyzed genogram.

In this case, we are facing a not informative genogram. It possesses four associate cases. Sporadic cases do not exist, i.e., cases that do not guard relation with any syndrome of predisposition to the cancer. The index node is the *probandus* of the genogram.

```
Análisis final del genograma 23as54ju9032sd2890

DIAGNÓSTICO FINAL: Familia afectada por el Síndrome de Cáncer Mama y Ovario Hereditario
(CMOH)

El probando (III.3), es una mujer de 46 años que tuvo un cáncer de mama a los 35.
Han aparecido otros tres casos de cáncer de mama en mujeres jóvenes (35, 40 y 42 años),
dos de ellos en la misma mujer (II.1). Hay además un cáncer de ovario en una mujer de 40
años.
La enfermedad podría venir por la línea paterna de la primera generación ya que el
individuo I.1 tuvo un cáncer de próstata a los 62 años.
En el CMOH es frecuente observar cáncer de próstata a veces con una edad de aparición más
joven que en los casos esporádicos.
```

Figure 7. Final report after exhaustive evaluation of the genogram

In the figure 7, we can observe the obtained result after the accomplishment of the exhaustive analysis of the genogram. This report is composed of two distinct parts like are diagnosis and analysis or explanation of the same. In the shown example, the family is affected by the Hereditary Breast Ovarian Cancer Syndrome (HBOC) and this diagnosis is argued in the lower part through a series of reasons and taken facts in the analysis that has been realized on the genogram. After this, the doctor will decide the measures to take in relation to family (follow-up more meticulous of the nodes with risk, genetic testing for *probandus*).

In order to provide to the system of different cases of test for his improvement and refinement, it focused mainly on the automatic construction of the genogram and further evaluation, has been carried out the development of an automatic generator of health record based on a Bayes classifier, i.e., a probabilistic classifier based on Baye's theorem. To calculate the probability that a node has of getting cancer or no. Moreover, has been applied 1:

$$\frac{p(type\_cancer) * p(g\_age\_p)}{p(type\_cancer) + p(g\_age\_0) + ... + p(g\_age\_n)} \quad (1)$$

where $p(type\_cancer)$ is the probability which has a node of suffering a certain type of cancer or not, $p(g\_age\_p)$ the probability of suffering this cancer in function to the group of age to which it belongs and $p(g\_age\_n)$ it is the probability that the different groups of age have.

From an objective population, a series of clinical data and the probabilities of incident and mortality for different types of cancer, family health records will be generated to realize an exhaustive training of the module in charge of the automatic construction of the genogram.

Likewise, an analysis has been carried out on the obtained genograms by applying *eye tracking* techniques [11].

The issues, to be evaluated, would be different. On the one hand, we have the comparative evaluation between the expert knowledge acquired of the analysis of the genogram and the royal procedure applied. This question, which identify the rest of the document as (RQ1), it would help additional at the moment of to guarantee or refine the procedure followed by the expert not only at the time of capture steps, if not at the moment of acquiring each and every of the stages that the expert follows for the resolution of the problem.

Furthermore, the evaluation of the presentation of the genogram (RQ2) will be very interesting, with the aim of showing a genogram an expert and/or the system in the most appropriate way, which would result in improved system efficiency and performance.

The experiment involves exposing a genograms battery for evaluation of all of them by the expert. As result there will be obtained the confirmation or refutation of a possible diagnosis of cancer for each of the patients under study.

In the first question RQ1, the analysis of the results would come given by a quantitative value of comparison between the original proceeding raised by the acquisition of knowledge and the actual applied procedure by the expert in the analysis of the genogram, by obtaining the recorded data by the *eye tracking* devices.

For the question RQ2, the analysis would come given by a study of the order of the steps followed by the participant of the study with regard to a genogram, of such a form, that these steps gathered by means of the technologies of eye tracking, possibly leading to a pattern as the order and the observed zone of the sample, in order to improve the presentation of genograms. To finish, the results will rest on a subjective valuation it brings over of the perception of the participant of the study with relation to the presentation of the genograms.

V. Conclusions and Future Work

In this article "A decision support system for risk analysis and diagnosis of hereditary cancer" has been presented. The proposed system pretend improve the identification process of suspicious families that suffer hereditary cancer, the process is arduous and is very important from the clinical point of view the identification of a series of characteristic aspects that present these families his for immediate diagnostic.

The main contribution of the system is the identification and diagnosis of several types of cancer such as breast cancer, ovary cancer and colon, in addition to different syndromes related Li-Fraumeni, Peutz Jeghers, among others. Finally, the automatic generation of the genogram is a great innovation, this fact will lead to a reduction of errors which could be committed in the manual construction of the genogram as well as a considerable decrease in the time of confection of the same.

In the future, we plan to realize various experiments with the aim of improving of incremental form the functioning of the system, thus, an increasingly reliable and robust system will be built. Future research will focus on system expansion in the diagnosis part, that is to say, it will have to evaluate and analyze more types of hereditary cancer and syndromes of predisposition.

References

[1] Montes, C., Gómez, A., Juristo, N. and Pazos, J., *Ingeniería del Conocimiento*. Ceura, 1997.

[2] Scott, M., *Management Decision Systems: Computer-Based Support for Decision Making*. Harvard University Press, 1971.

[3] Keen, P., *Decision Support Systems: A research perspective*. *Massachusetts Institute of Technology*, March 1980.

[4] Corrigan, J., Kohn, L. and Donaldson, M., *To err is human: Building a safer health system*. Committee on Quality of Health Care in America, Institute of Medicine, 2000.

[5] Greenes, R., *Clinical Decision Support: The Road Ahead*. Academic Press, 2011.

[6] Benajmin, R. and Robins, M., *Cancer Biology*, 3rd ed. Harlow, England: Pearson Education Limited, 2006.

[7] Gascón, A., Cruz, J., de Lara, A., Osorio, A., Perea, J., Robledo, M., Bando, H., Benítez. J. and Urioste, M., *Planteamientos básicos del cáncer hereditario; principales síndromes*. ADEMAS Comunicación Gráfica, s.l., 2011.

[8] Kennedy, V., *Genograms*. MAI Review, 2010.

[9] ASTM INTERNATIONAL, *Standar Specifitacion for Continuity of Care Record*. http://www.astm.org/Standards/E2369.htm, (Last visit - April 22, 2015).

[10] Goldstein, M. and Woff, D., *Bayes Linear Statistics: Theory and Methods*. Wiley & Sons Ltd, 2007.

[11] Hassan, Y. and Herrero, V., *Eye-Tracking en Interacción Persona-Ordenador*, 2007.

SESSION

POSTER PAPERS

Chair(s)

TBA

Plural Object Recognition using Image Similarity and Word-Concept Association

[1]Hirokazu Watabe, [2]Misako Imono, [3]Eriko Yoshimura, and [4]Seiji Tsuchiya
Dept. of Intelligent Information Engineering and Science,
Doshisha University, Kyotanabe, Kyoto, 610-0394, Japan
[1]hwatabe@mail.doshisha.ac.jp, [2]mimono@mail.doshisha.ac.jp,
[3] sk109716@mail.doshisha.ac.jp, [4]stsuchiy@mail.doshisha.ac.jp

Abstract *- Object-recognition capabilities will be indispensable for any intelligent robot that will serve as a human partner. In this research, we combined existing image-processing and word-concept association techniques to perform object recognition based human style common-sense judgments.*

Keywords: object recognition, image similarity, degree of association, Concept-Base

1 Object recognition based on common-sense judgments

Our goal in this research was object recognition based on comprehensive judgments that mimic the human object-recognition mechanism using an image database and a word-concept association system. The procedural flow is as follows:

(1) Image input (background image + foreground image)
(2) Object extraction by background subtraction
(3) Candidate acquisition via size judgment
(4) Filter by image similarity
(5) Filter by degree of association
(6) Output object names

2 Candidate acquisition via size judgment

We used an image database to acquire candidates using size judgments. Then, we obtained the area of the region cut away by object extraction, normalized that, and compared the result with each object in the image database in terms of size to acquire object candidates.

The image database stores multiple sets of images of objects with their names. For a single object, it will store multiple images, and the current database of 50 objects includes a total of 200 images, each with size information (Figure 1).

3 Filter by image similarity

After adjusting direction of principal axes and center of gravities of two object domains I_1 and I_2, for color and binary images, the correlation values S_1 and S_2 are calculated each and the image similarity S is derived by equation 1. Because there are two ways of the direction of principal axes, the bigger one is adopted. In addition, It is used L*a*b* color system for the correlation value calculation of a color image.

$$S\ (I_1, I_2) = w_1 S_1 + w_2 S_2 \qquad (1)$$

We then filtered candidates using image similarity S between the input image and images in the image database.

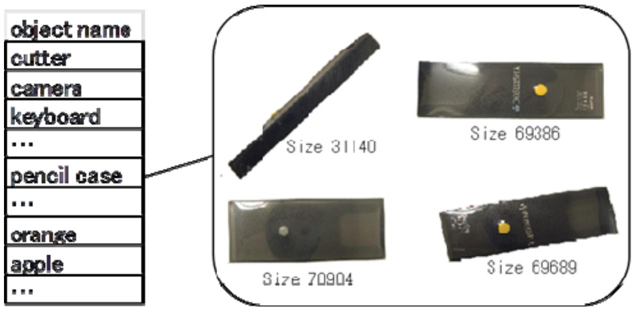


Fig.1: The image database

4 Word Concept Association System

4.1 Concept-Base

A concept-base (CB) is a knowledge base comprised of terms (concepts) mechanically constructed from multiple sources such as Japanese-language dictionaries and newspapers, along with terms (attributes) that express their semantic features. Concepts have been given attributes along with a weighting that expresses their importance. Approximately 90,000 concept notations have been compiled in the CB, with an average of 30 attributes for one concept. A certain concept A has a pair of sets of attribute a_i and weighting w_i, as appear below.

$$A = \{(a_1, w_1), (a_2, w_2), \cdots, (a_m, w_m)\}$$

Any primary attribute a_i is composed of the terms contained in the set of concept notations in its CB. Therefore, to ensure that a primary attribute matches a certain concept notation, that primary attribute can be further extracted. This is called a secondary attribute. In a CB, a concept is defined by a chained set of attributes to the n-th order.

4.2 Degree of Association

The Degree of Association(DoA)[(1)] is a value ranging from 0.0 to 1.0 that quantifies the strength of the relationship between words and words registered to the CB. There are three types of DoA [(2)]:

1) DoA concerned with meaning of words (DoA-M)
2) DoA concerned with coincidental word information (DoA-C)
3) DoA concerned with above both (DoA-MC)

Using the degree of association with combinations of two object names in the image database, the candidates are filtered.

5 Evaluation

5.1 Object extraction

We observed considerable residual noise when performing input image object extraction by background subtraction. We regard this as being a result of using a simple difference method to discern the input image from the background. Because this degradation and noise will affect subsequent image-similarity calculations and other processes, it will be necessary to consider better extraction techniques using other image-processing technologies.

5.2 Filter by size judgments

Size filtering eliminated a mean of 24 candidate objects, and did not erroneously eliminate the correct answer as a candidate. Based on this, we regard our current size filtering technique to be appropriate.

5.3 Filter by image similarity

The image similarity is calculated by the correlation value of the color image and that of the binary image with weightings w_1, w_2. (Equation 1)

We inspected these weightings w_1, w_2 using the image which came out of 40 objects same as an object stored away within image database. As shown in the figure 2, at the time of weightings (w_1,w_2)=(0.4,0.6), the precision became approximately 60%. When only the binary image is used, the precision became 42%, and when only the color image is used, the precision became 31%. It may be said that compositely using both binary image and color image is effective.

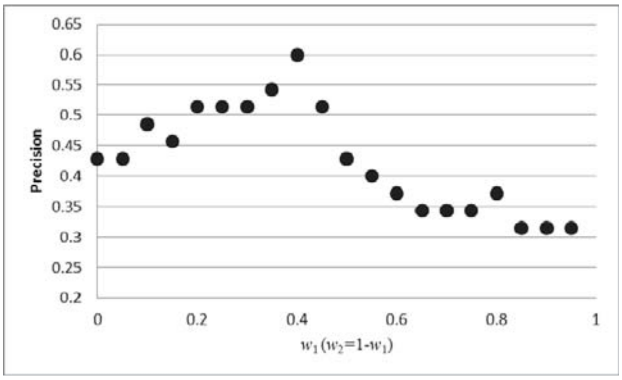


Fig.2: Image similarity

5.4 Filter by DoA

To filter by DoA, it is used the image similarity (S) and DoA (R) by combining them to obtain an assessed value (Equation 2).

$$E = W_S S + W_R R \tag{2}$$

Using 80 images (160 objects) including each two objects which may not be objects in the image database shown like figure 3, the experiment is performed with changing weightings W_S and W_R . As the degree of association, three types of those, DoA-M, DoA-C and DoA-MC are examined.

Figure 4 shows the results of this experiment. By this result, in the case using DoA concerned with coincidental word information (DoA-C) and weightings (W_S,W_R)=(0.6,0.4), the precision is the best 45%.

Therefore, it may be said that compositely using both image similarity and the word concept association is effective.

Fig.3: Part of input images

6 Conclusion

In this study, using an image similarity by the technique of the conventional image processing and the association between the plural number object, it is aimed at the realization of the object recognition based on the general judgment such as the human object recognition mechanism. The precision of the object recognition became approximately 45% by using the value of the degree of association. It was approximately 40% when the degree of association was not used, and approximately 5% of precision improvement was achieved. It is thought that it was able to show the usefulness of using a word concept association system as one technique of the object recognition.

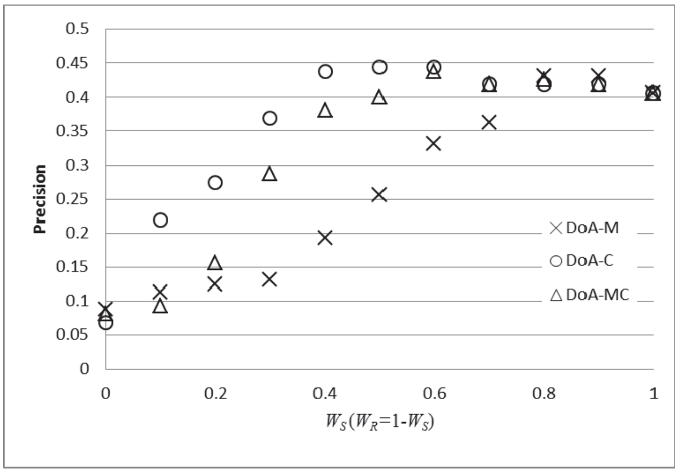


Fig.4: Experimental results

Acknowledgements

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

References

(1) H. Watabe and T. Kawaoka: "The Degree of Association between Concepts using the Chain of Concepts", *Proc. of SMC2001*, pp.877-881, 2001.

(2) H. Watabe, N. Okumura and T. Kawaoka: "The Method of Measuring the Degree of Association between Concepts using Attributes of the Concepts and Coincidence Information", *Journal of Natural Language Processing*, Vol.13, No.1, pp.53-74, 2006.

Robust stability criteria for Takagi-Sugeno fuzzy systems with sampled-data

H.Y. Jung[1], Yajuan Liu[2], and S.M. Lee[2]
[1] Department of Information Communication Engineering, Yeungnam University, Gyungsan, Republic of Korea
[2]School of Electronics Engineering, Daegu University, Gyungsan, Republic of Korea

Abstract— *This paper presents the robust stability of Takagi-Sugeno(T-S) fuzzy systems with sampled-data. By constructing an augmented modified Lyapunov-Krasovskii functional, which includes three triple integrals, some less conservative results are obtained compared with the existing results. Numerical examples are presented to demonstrate the improvement of the proposed method.*

Keywords: Takagi-Sugeno fuzzy systems, time-varying delay, Lyapunov-Krasovskii functional, Linear Matrix Inequality

1. Introduction

The main advantage of T-S fuzzy model is that it can combine the flexibility of fuzzy logic theory and rigorous mathematical theory of linear system into a unified framework to approximate complex nonlinear systems [1-3]. On the other hand, sampled-data often appears in many dynamical systems such as biological systems, neural networks, networked control systems and so on. For a digital stabilization, sampled-data control design method can reduce the amount of stabilization information and increases the efficiency of bandwidth usage. Hence, the stability of T-S fuzzy systems with sampled-data has been studied by many researchers [3]. By construction of a modified augmented Lyapunov-Krasovskii functional, which includes three triple integrals, an improved stability criterion for guaranteeing the asymptotically stable is derived by using Wirtinger-based integral inequality [5], reciprocally convex approach [4], new delay-partitioning method. It should be pointed out that different with delay-partitioning method used in [3], we only divide the time interval into some subintervals, but not considered the cases that time-varying delay belongs to different subinterval, respectively. Finally, Numerical example is given to demonstrate the effectiveness of the proposed method.

2. Problem statement and preliminaries

Consider the following nonlinear system which can be modeled as TS fuzzy model with sampled-data:
Rule i: If $\theta_1(t)$ is M_{i1} and $\ldots$ and If $\theta_n(t)$ is M_{in}, then

$$\dot{x}(t) = A_i x(t) + A_{di} x(t_k), i = 1, 2, \ldots, r. \quad (1)$$

where $\theta_1(t), \ldots, \theta_n(t)$ are the premise variables, M_{ij} is a fuzzy set, $i = 1, 2, \ldots, r, j = 1, 2 \ldots, n$. r is the index number of fuzzy rules, and $x(t) \in \mathcal{R}^n$ denotes the state of the system. A_i and A_{di} are the known system matrices and sampled-state matrices with appropriate dimensions, respectively.

In this paper, the sampled signal is assumed to be generated by using a zero-order-hold (ZOH) function with a sequence times $0 \leq t_0 < \cdots < t_k \cdots < \lim\limits_{k\to\infty} t_k = +\infty$.

$$t_{k+1} - t_k \leq h.$$

Assume that $h(t)$ is a time-varying delay satisfying

$$0 \leq h(t) \leq h, \dot{h}(t) \leq \mu, \quad (2)$$

where h, μ are known constants.

Using singleton fuzzifier, product inference, and center-average defuzzifier, the global dynamics of the delayed T-S system (1) is described by the convex sum form

$$\dot{x}(t) \quad = \quad \sum_{i=1}^{r} p_i(\theta(t))[A_i x(t) + A_{di} x(t - h(t))] \quad (3)$$

where $p_i(\theta(t))$ denotes the normalized membership function satisfying

$$p_i(\theta(t)) = \frac{w_i(\theta(t))}{\sum_{i=1}^{r} w_i(\theta(t))}, \ w_i(\theta(t)) = \prod_{j=1}^{n} M_{ij}(\theta_j(t)), \quad (4)$$

where $M_{ij}(\theta_i(t))$ is the grade of membership of $\theta_i(t)$ in M_{ij}. For the sake of simplicity, let us define

$$\bar{A} = \sum_{i=1}^{r} h_i(\theta(t)) A_i, \ \bar{A}_d = \sum_{i=1}^{r} h_i(\theta(t)) A_{di}. \quad (5)$$

Now, the system (3) can be rewritten as

$$\dot{x}(t) \quad = \quad \bar{A}x(t) + \bar{A}_d(x(t - h(t)). \quad (6)$$

3. Main results

Let us consider the following Lyapunov-Krasoskii functional candidate

$$V(t) = \sum_{i=1}^{5} V_i, \quad (7)$$

where

$$V_1 = \begin{bmatrix} x(t) \\ \int_{t-\alpha h}^{t} x(s)ds \\ \int_{t-h}^{t-\alpha h} x(s)ds \end{bmatrix}^T \mathcal{P} \begin{bmatrix} x(t) \\ \int_{t-\alpha h}^{t} x(s)ds \\ \int_{t-h}^{t-\alpha h} x(s)ds \end{bmatrix},$$

$$V_2 = \int_{t-h(t)}^{t} x^T(s)Q_1x(s)ds,$$

$$V_3 = \int_{t-\alpha h}^{t} x^T(s)Q_2x(s)ds + \int_{t-h}^{t} x^T(s)Q_3x(s)ds,$$

$$V_4 = \alpha h \int_{-\alpha h}^{0}\int_{t+\alpha}^{t} \dot{x}^T(s)R_1\dot{x}(s)dsd\alpha,$$

$$V_5 = (1-\alpha)h \int_{-h}^{-\alpha h}\int_{t+\alpha}^{t} \dot{x}^T(s)R_2\dot{x}(s)dsd\alpha d.$$

By using above Lyapunov funtional, we can derive a stability criterion for delayed T-S fuzzy systems, and $e_i \in \mathcal{R}^{10n\times n}(i = 1, 2, \dots, 10)$ are defined as block entry matrices(for example $e_4 = [0\ \ 0\ \ 0\ \ I\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$). The other notations are defined as:

$$\begin{aligned}
&\Xi_1^1 = \Pi_1^1\mathcal{P}[e_8\ \ e_1-e_2\ \ e_2-e_4]^T + (*),\\
&\Xi_1^2 = \Pi_1^2\mathcal{P}[e_8\ \ e_1-e_2\ \ e_2-e_4]^T + (*),\\
&\Xi_2 = e_1Q_1e_1^T - (1-\mu)e_3Q_1e_3^T,\\
&\Xi_3 = e_1Q_2e_1^T - e_2Q_2e_2^T + e_1Q_3e_1^T - e_4Q_3e_4^T,\\
&\Xi_4^1 = (\alpha h)^2e_8R_1e_8^T - \Pi_2^1\begin{bmatrix} \bar{R}_1 & \mathcal{S}_1 \\ * & \bar{R}_1 \end{bmatrix}\Pi_2^{1T},\\
&\Xi_4^2 = (\alpha h)^2e_8R_1e_8^T - [e_1-e_2]R_1[e_1-e_2]^T\\
&\qquad -3[e_1+e_2-2e_7]R_1[e_1+e_2-2e_7]^T,\\
&\Xi_5^1 = ((1-\alpha)h)^2e_8R_2e_8^T - [e_2-e_4]R_2[e_2-e_4]^T\\
&\qquad -3[e_2+e_4-2e_7]R_1[e_2+e_4-2e_7]^T,\\
&\Xi_5^2 = ((1-\alpha)h)^2e_8R_2e_8^T - \Pi_2^2\begin{bmatrix} \bar{R}_2 & \mathcal{S}_2 \\ * & \bar{R}_2 \end{bmatrix}\Pi_2^{2T},\\
&\Upsilon_1 = \Xi_1^1 + \Xi_2 + \Xi_3 + \Xi_4^1 + \Xi_5^1,\\
&\Upsilon_2 = \Xi_1^2 + \Xi_2 + \Xi_3 + \Xi_4^2 + \Xi_5^2,\\
&\bar{\Gamma} = [\bar{A}\ \ 0\ \ \bar{A}_d\ \ 0\ \ 0\ \ 0\ \ 0\ \ -I],\\
&\Gamma_i = [A_i\ \ 0\ \ A_{di}\ \ 0\ \ 0\ \ 0\ \ 0\ \ -I].
\end{aligned}$$

Now we have the following theorem.

Theorem 1 For given scalars h, μ, the system (1) is globally asymptotically stable if there exist symmetric positive matrices $\mathcal{P} \in \mathcal{R}^{3n\times 3n}, \mathcal{M} \in \mathcal{R}^{2n\times 2n}, Q, R_1, R_2, N_1, N_2 \in \mathcal{R}^{n\times n}$, a positive scalar ϵ and any matrix $\mathcal{S}_j(j = 1, 2, 3, 4) \in \mathcal{R}^{2n\times 2n}$ such that the following LMIs hold for all $h(t) \in [0, h]$

$$(\Gamma_i^{\perp})^T\Upsilon\Gamma_i^{\perp} < 0, i = 1, 2 \quad (8)$$

$$\begin{bmatrix} \bar{R}_1 & \mathcal{S}_1 \\ * & \bar{R}_1 \end{bmatrix} \geq 0, \quad \begin{bmatrix} \bar{R}_2 & \mathcal{S}_2 \\ * & \bar{R}_2 \end{bmatrix} \geq 0, \quad (9)$$

where $\bar{R}_1 = \begin{bmatrix} R_1 & 0 \\ * & 3R_1 \end{bmatrix}, \bar{R}_2 = \begin{bmatrix} R_2 & 0 \\ * & 3R_2 \end{bmatrix}$.

Proof The detailed proof is omitted.

4. Numerical example

Consider the system with the following parameters

$$A_1 = \begin{bmatrix} -3.2 & 0.6 \\ 0 & -2.1 \end{bmatrix}, \quad A_{d1} = \begin{bmatrix} 1 & 0.9 \\ 0 & 2 \end{bmatrix},$$
$$A_2 = \begin{bmatrix} -1 & 0 \\ 1 & -3 \end{bmatrix}, \quad A_{d2} = \begin{bmatrix} 0.9 & 0 \\ 1 & 1.6 \end{bmatrix}.$$

The maximum value of upper bound h compared with the results in [1-3] with different μ u are listed in Table 1.

Table 1: Upper delay bound h for different μ.

μ	0.03	0.1	0.5	0.9
[2]	0.7805	0.5906	0.5392	0.5268
[1]	0.8369	0.7236	0.7154	0.7014
[3]	0.8771	0.7687	0.7584	0.7524
Theorem 1 ($\alpha = 0.5$)	1.5835	1.2444	1.2216	1.1686
Theorem 1 ($\alpha = 0.6$)	1.5906	1.2698	1.2445	1.1852

5. Conclusion

The robust stability for T-S fuzzy systems with sampled-data has been investigated. Less conservative criteria have been obtained by employing new delay-partitioning technique, integral inequality and reciprocally convex approach. Numerical examples have been given to demonstrate the effectiveness of the proposed method.

Acknowledgment

The research of S.M. Lee was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2014R1A1A4A01003860). Also, the work of H.Y. Jung was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0011096).

References

[1] O.M. Kwon, M.J. Park, S.M. Lee, Ju H. Park, "Augmented Lyapunov-Krasovskii functional approaches to robust stability criteria for uncertain Takagi- Sugeno fuzzy systems with time-varying delays", Fuzzy Sets Syst., vol. 201, pp. 1-19, 2012.

[2] F. Yang, S. Guan, D. Wang, "Quadratically convex combinatio approach to stability of T-S fuzzy systems with time-varying delay", Journal of the Franklin Institute, vol. 351, pp.3752-3765, 2014.

[3] H.-B. Zeng, Ju.H. Park, J.-W Xia, S.-P Xiao, "Improved delay-dependent stability criteria for T-S fuzzy systems with time-varying delay", Appl. Math. Comput., vol. 235, pp. 492-501, 2014.

[4] P.G. Park, J.W. Ko, C.K. Jeong, "Reciprocally convex approach to stability of systems with time-varying delays", Automatica, vol. 47, pp. 235-238, 2011.

[5] A. Seuret, F. Gouaisbaut, "Wirtinger-based integral inequality: application to time-delay systems", Automatica, vol. 49, pp. 2860-2866, 2013.

Automating Academic Advising and Course Schedule Planning with a CLIPS Expert System

Edwin Rudolph and Adel Abunawass
Department of Computer Science, University of West Georgia, Carrollton, Georgia, USA

Abstract – *A significant aspect of advising students is the process of reviewing the students' academic course history, determining progress toward completion of degree requirements, verifying satisfaction of various pre-requisites, and identifying potential course schedule plans for future enrollment. To assist advisors in accurately and efficiently completing these tasks and to enable more time for qualitative and personalized discussion with individual students, we developed a simple expert system that automates those tasks. Related to advising is the administrative problem of determining what courses should be offered to meet students' future needs. To aid and support decision making, the advising expert system is designed to process data for multiple students simultaneously and project the number of students needing to take certain courses in the future. This poster discusses the general design and implementation of this system using the CLIPS expert system tool.*

Keywords: Expert system; decision support system; academic advising; CLIPS

1 Motivation

As part of efforts to improve and streamline advising of students in our department, we began looking for ways to ensure students would receive consistent advice on their academic progress as well as on issues relating to various university administrative polices and procedures, while also ensuring that each student had the opportunity to receive substantive and personalized advisement on issues such as personal and career goals and research and job opportunities. The challenge with the former was that the routine and largely mechanical advising tasks were dominating the advisement process, and the fact that several faculty members participated in this process also made it difficult to ensure that students received consistent information.

Our approach to addressing this challenge was to implement a two-tier mandatory advising process where the routine aspects of advising (reviewing progress toward a degree, consulting on what courses to take for upcoming semesters, discussion about administrative policies and procedures, etc.) are centralized with a professional staff advisor, with whom the student must first meet, and following this, the student meets with an assigned faculty advisor where academic, career, research, and personal issues may be discussed. This process, which all students must complete during each term of enrollment, achieved both the goal of ensuring students receive consistent advice about degree requirements and various policies and procedures, while improving the quality and content of the consultation with a faculty member. However, to do this in an effective manner, we needed to find a way to automate the analysis of the student's progress to further ensure consistency and accuracy as well as enable the staff advisor to efficiently meet with a large number of students over a relatively short time period (typically a one to two month period preceding the course registration period).

2 Requirements

We identified three functional requirements for a system that would help us to automate key advising tasks:

- To aid in reviewing a student's progress towards degree completion, list all requirements for the degree and indicate for each whether or not the student has satisfied the requirement. For satisfied requirements, indicate the course(s) used to satisfy those requirements. Since different sets of requirements may exist depending on when the student matriculated, the system must allow for multiple requirement sets and allow the appropriate set to be selected on demand.

- To aid in planning a student's future course schedule, list important program courses the student needs to take, the future term(s) the department expects to offer those courses, and whether or not the student is eligible to take each course based on whether or not the student has satisfied the pre-requisite(s).

- To assist in planning what courses should be offered in future terms by projecting students' needs, list important program courses and for each indicate how many students are eligible to take the course (based on having satisfied applicable pre-requisites).

Important non-functional requirements included the ability to easily code new and/or modified degree requirements as requirements change over time and to be able to handle input and output in a simple text-based format.

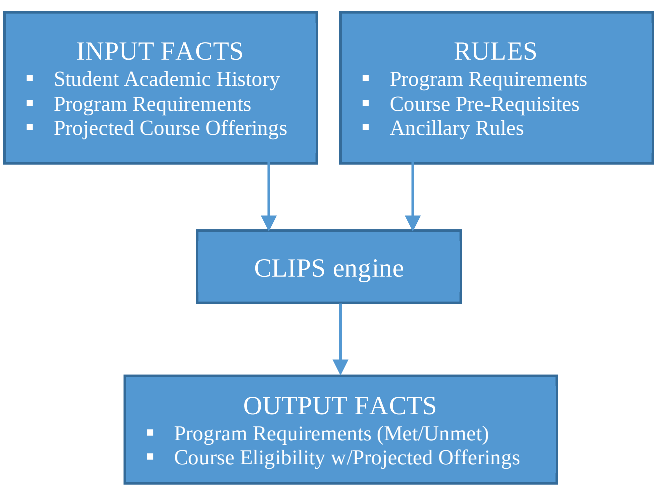
3 Implementation

An initial prototype system was developed as a web-based application using procedural code in PHP. However, it quickly became obvious that this approach was both tedious to code and difficult to maintain. This initial effort did prove beneficial in that it highlighted the fact that the system was primarily based upon a set of *if-then* rules, which led us to rethink our implementation approach in terms of a rule-based expert system. We chose the CLIPS expert system shell for several reasons: low-cost (public domain), fast (shell implemented in C), a straightforward syntax, and the ability use and parse simple text input and output.

3.1 Overview

The expert system is forward chaining, and uses input facts about the student's academic history (courses taken), the requirements of the student's program of study, and facts about future planned course offerings to produce conclusions (output facts) about what program requirements have or have not been satisfied, and what courses a student may be able to take in the future and when those courses are expected to be offered. The rules that produce those conclusions consist of: program requirement rules that match courses the student has taken with individual program requirements; course pre-requisite rules that match courses the student has taken with pre-requisites needed to enroll in courses that the student has not taken; and ancillary rules that handle issues such as grade replacements for retaking a course, manual overrides to enable course substitutions, and output conversion to enable reporting and parsing of results.

The following figure illustrates the basic components of the expert system as described above:



3.2 Degree Program Requirements Progress Report

To produce the list of all requirements for the degree along with an indication of whether or not a student has satisfied the requirement, the expert system works as follows. First, individual input facts for each course the student has taken along with the grade earned are asserted. Then, facts that describe the individual degree program requirements (e.g. *"two elective courses are required from the following…"*) are asserted. For program requirements where there is a single specific course required, the requirement fact is coded to match the course number that would appear in the student's academic history facts. Then, a single rule is used to automatically match all such simple requirements. For more complex requirements where one or more courses are used to determine if the requirement is met, a specific rule is defined to perform the match. Once a match occurs between course(s) from the student's academic history and a program requirement, the requirement fact is modified to indicate that the requirement was met and which course(s) were used to meet it.

3.3 Course Enrollment Eligibility

To produce a list of courses a student has not taken but is eligible to take based on courses already completed (or in progress), the system uses the student's academic history facts (courses taken) to activate pre-requisite satisfaction rules. For example: *"CS 201 requires completion of CS 102 with a grade of C or higher"*. This rule would fire and generate a new eligibility fact for CS 201 if the student's academic history contained a fact for CS 102, the grade for that course was a C or better, and the academic history did not already contain a fact for CS 201.

To aid the student in schedule planning, additional input facts are asserted for projected course offerings, indicating that the department expects to offer a course during a given term (e.g. "CS 201 will be offered in Spring 2016"). These facts are then matched with the eligibility facts; resulting in a list of courses the student can take in the future along with the term(s) the department expects to offer those courses.

3.4 Course Offering Needs Projection

To aid in projecting how many students may need to take a certain course, the system uses the same functionality described above for course enrollment eligibility. Instead of only considering a single student, data for all students is asserted (using student IDs to differentiate). Eligibility facts are asserted for each student and these are then counted to produce a projected number of students who may need to take certain courses in the future. For this use case, the program requirement rules for analyzing students' degree progress are not utilized and are disabled so as to minimize processing time.

4 Results

Since 2007, the system has been successfully used in support of over 3,000 advisement sessions and in the planning of course offerings. The CLIPS implementation has enabled us to easily update the system over time to support changes to degree requirements and course pre-requisites (while maintaining older ones), as well as, adapt the analysis engine for use with different user interfaces and reporting needs.

A Verification Technique for Self-Adaptive Software by Using Model-Checking

Euijong Lee[1] and Doo-Kwon Baik[2]
Department of Computer and Radio Communications Engineering,
Korea University
Seoul, Republic of Korea
{kongjjagae[1], baikdk[2]}@korea.ac.kr

***Abstract*— Self-adaptive software refers to software that can change its behavior by itself to perform an intended objective according to changes in the surrounding environment. In this study, a technique is proposed to detect the possibility of problem occurrence in advance, before the self-adaptive software carries out self-adaptation according to changes in the surrounding environment. For the proposed technique, a model-checking technique is applied, which is a software modeling verification technique. The proposed technique was applied to ZNN.com, which provides a self-adaptive software scenario. Through this, the possibility was shown that an adaptive-strategy can be prepared in advance by self-adaptive software through the model-checking technique in the stage prior to the execution of self-adaptation.**

Keywords— Self-adaptive software, Model-Checking, Software verification

I. Study Background

For conventional software, a problem is defined and the method, according to which it is solved, is constructed at the development stage. With these types of software development methods, when a problem that did not exist at the development stage occurs, a software error occurs or in a serious case, the software becomes inoperable. To solve this problem, the concept of self-adaptive software has emerged. Self-adaptive software refers to software that changes by itself according to environmental changes, thereby solving a problem. Currently, studies are being carried out on self-adaptive software in various aspects in different fields. This study was conducted for problem detection, which is one of the research areas for self-adaptive software, and it aims to detect and prevent the possibility of problem occurrence in advance before a change of environment occurs.

II. Proposed Method

This study was carried out to identify a software problem before it happens by using the model-checking technique, and to prevent it through self-adaptation on the basis thereof. The proposed method was based on MAPE-K proposed by IBM. Because this study aims to detect a problem in advance in a stage before it happens, the model-checking part was added and subdivided in the monitoring stage, and each stage of MAPE-K, except for monitoring, performs the inherent roles.

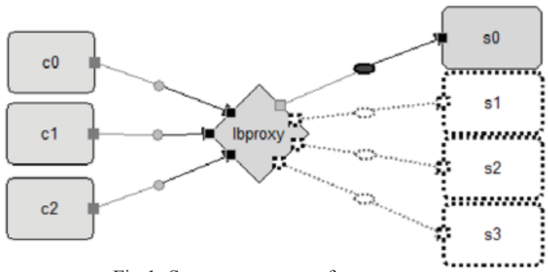


Fig 1. System structure of znn.com

III. Case Study

A. Scenario: ZNN.com

ZNN.com is a case study used by Rainbow to evaluate self-adaptive software, and it consists of the structure shown in Fig. 1 [1]. Znn.com is a virtual news media website. Znn.com has the goal of providing a reasonable response time to a user. Znn.com provides three service stages to its users according to the server load. For example, when a server overloads, only texts are provided, and when a server has available capacity, articles are provided along with various media by improving the quality of service. Furthermore, when the data traffic increases considerably, a strategy is adopted to reduce the time required to respond to users by increasing the number of operable servers. Consequently, two criteria can be selected to execute self-adaptation for znn.com. The first one is performance, which can be the response time, the server load, and the network bandwidth. The second one is cost, which can be expressed by the number of operating servers.

B. Application of Case Study

An experiment was carried out to show that the possibility of problem occurrence can be identified in advance by using the model-checking technique. The znn.com scenario mentioned in 3.A was used as the scenario for the experiment. In accordance with the scenario, the server loads were defined in Table 1 according to the service types, and the amount that can be processed by the server was defined as 300 M/s. In accordance with the experimental scenario of znn.com, the environmental variables that will be used by the proposed technique before defining the FSM were defined in Table 2.

TABLE 1. TRAFFIC ACCORDING TO SERVICE

Quality of Service	Used Traffic
High	3 MB/S
Medium	2 MB/s
Low	1 MB/s

TABLE 2. THE ENVIRONMENTAL VARIABLES USED IN THE SCENARIO

Environment Variable	Description
User number	The number of users accessing the ZNN.com service
Service quality	It means the quality of service and it is 3 in the case of highest service quality, 2 in the case of medium service quality, and 1 in the case of the lowest service quality
Running server number	The number of servers currently operating

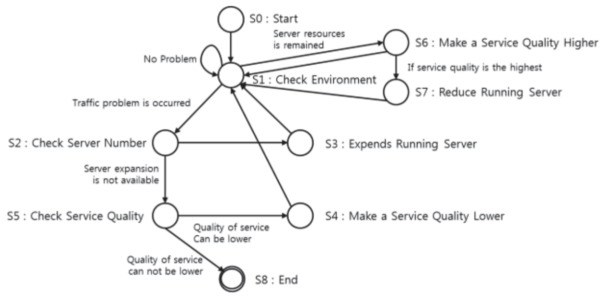


Figure 2. FSM representing znn.com

Based on the above assumption, the FSM that was constructed to use the model-verification technique for verification is shown in Fig. 3. S0 is the stage for starting the model verification. S1 is the stage for confirming the current status through the environmental variables. In the pertinent model, the service status was identified with the following equation.

$$\mathrm{S(e)} = \frac{Service\ quality\ \times User\ Number}{Running\ Server\ Number\ \times Capacity\ of\ Server}$$

If the value of S(e) is larger than 1, it is determined that there will be no problem in providing the service, and if smaller than 1, it is determined that there will be a problem in providing the service. S2 is the stage for confirming the available servers in addition to the currently operating servers, and when there are available servers, S3 is the state in which they are used. When there is no available server, S4 is the stage for checking the service quality to determine if it can be reduced, and if there is a level to which the service can be lowered, the service quality is decreased through the S5 stage. S6 and S7 means that if server resource remain, make better quality or reduce running server to reduce cost. Lastly, S8 means that the service is terminated because there are no further possible methods. Furthermore, the case of not satisfying the S1 stage means that self-adaptation is required, and the S3 and S5 stages mean that self-adaptation has been applied.

For the proposed FSM model, the number of users was defined as 0–10,000 by using a NuSMV model-verifier; and the values were checked for the environmental variables occurring in S3 and S5 where self-adaptation is required, and S6 where the service operation is impossible. The pertinent results are shown in Table 3.

TABLE 3. VALUES OF ENVIRONMENTAL VARIABLES THAT REQUIRE SELF-ADAPTATION

Environmental Variable Value	Status	Adaptation Strategy
User number: 101/201 Service quality: 3/3 Running server number: 1/2	S3	Increase a running server
User number: 301 / 451 Service quality: 3 / 2 Running server number: 3 /3	S5	Decrease the service quality
User number: 99 / 199 Service quality: 3 / 3 Running server number: 2 /3	S7	Decrease a running server
User number: 299 / 449 Service quality: 2 / 1 Running server number: 3 /3	S6	Increase the service quality
User number: 901 Service quality: 1 Running server number: 3	S5	Impossible to provide the service

IV. CONCLUSION

This study was carried out to extract the adaptation stages of self-adaptive software and prepare for them in advance. For this, a technique was proposed for extracting the environmental variables, which are required for self-verification in the monitoring stage during the MAPE-K process used in conventional self-adaptive software, and based on this, extracting the self-adaptive stage in advance through model-checking. The proposed technique was applied to the znn.com scenario, which is used as a case study in self-adaptive software studies. Through the applied results, the surrounding environment where self-adaptation is required was extracted prior to executing an adaptation-strategy by using the model-checking technique.

In future, a follow-up study will be carried out to use the proposed technique during actual execution time. Furthermore, a study will be carried out to develop a technique that extracts or generates an adaptation strategy through model-checking.

ACKNOWLEDGMENT

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF 2012M3C4A7033346). Doo-Kwon Baik is corresponding author.

REFERENCES

[1] Cheng, Shang-Wen, David Garlan, and Bradley Schmerl. "Evaluating the effectiveness of the rainbow self-adaptive system." Software Engineering for Adaptive and Self-Managing Systems, 2009. SEAMS'09. ICSE Workshop on. IEEE, 2009.

[2] Mazeiar Salehie and Landan Tahvildari, "Self-Adaptive Software: Landscape and Research Challenges", In TAAS, Vol 4, Issue 2, pp14-42, May 2009.

[3] Biere, Armin, et al. Symbolic model checking without BDDs. Springer Berlin Heidelberg, 1999.

[4] Baier, Christel, and Joost-Pieter Katoen. Principles of model checking. Vol. 26202649. Cambridge: MIT press, 2008.

[5] Garlan, David, et al. "Rainbow: Architecture-based self-adaptation with reusable infrastructure." Computer 37.10 (2004): 46-54.

[6] Principles of model checking. Vol. 26202649. Cambridge: MIT press, 2008.

Judging Emotion from EEGs Using SVM and EEG Features

Seiji Tsuchiya, Mayo Morimoto, Misako Imono and Hirokazu Watabe
Dept. of Intelligent Information Engineering and Science, Faculty of Science and Engineering,
Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract - *For a robot to converse naturally with a human, it must be able to accurately gauge the emotional state of the person. Techniques for estimating emotions of a person from facial expressions, intonation and speech content have been proposed. This paper presents a technique for judging the emotion of a person using EEGs. Accuracy of emotion judgment using EEG features of all subjects was 43.6% and using EEG features of only principal subject was 55.7%. However, performance accuracy remains low, and continued development is required through further development of methods for both reducing noise mixed in with EEGs.*

Keywords: EEG, judging emotion, SVM

1 Introduction

For a robot to converse naturally with a human, it must be able to accurately gauge the emotional state of the person. Techniques for estimating emotions of a person from facial expressions, intonation and speech content have been proposed. This paper presents a technique for judging the emotion of a person using EEGs, differing from the data used to date.

2 Overview of Proposed Technique

The objective of this technique was to read the emotions of a conversation partner from EEGs.

EEGs acquired from the subject are used as source EEGs. Emotions of the subject at that time are acquired simultaneously. Spectrum analysis of the source EEGs to which emotion flags have been assigned is performed every 1.28 s, and the EEG features of θ waves (4.0 Hz to 8.0 Hz), α waves (8.0 Hz to 13.0 Hz) and β waves (13.0 Hz to 30.0 Hz) are determined (Fig 1).

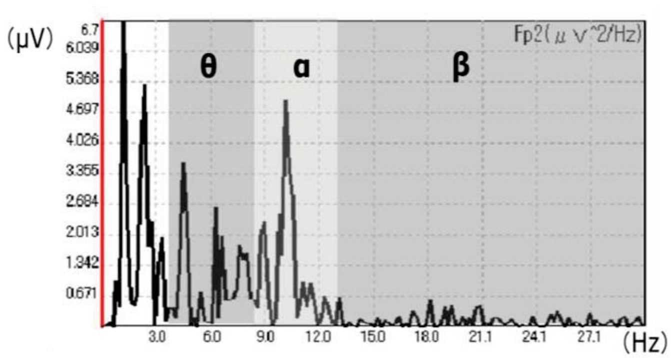


Fig.1 Spectrum analysis of the source EEGs

Emotion are judged from EEGs by support vector machine (SVM) [1] using the learning data which is EEG features to which emotion flags have been assigned was determined in this study. Emotions judged in this study were pleasure, anger, sadness, and no emotion.

3 Acquisition of Source EEGs and Emotions

EEGs were measured at 14 locations, at positions conforming to the International 10-20 system (Fig.2) [2]. Subjects fitted with an electroencephalography [3] cap were asked to watch a Japanese film for approximately 2 h while trying to gauge the emotions of the speakers in the film, and source EEGs were acquired. Images were frozen for each of the 315 speakers in the film, and the subject was asked what emotion the speaker was feeling at that time.

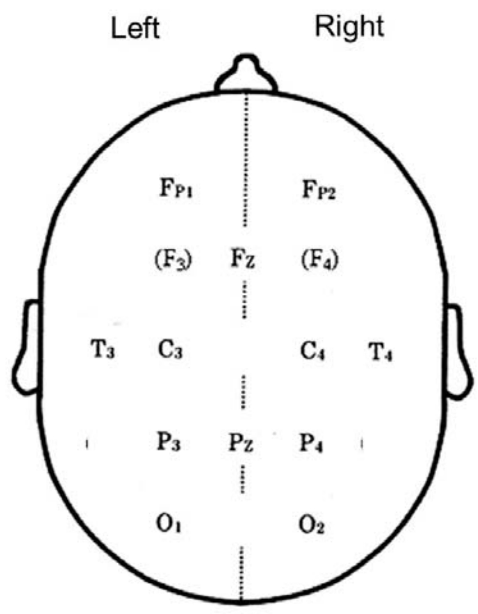


Fig2. 14 locations to measure EEG conforming to the International 10-20 system

Eighteen subjects were used, and viewing was divided into four sessions to reduce the physical burden on subjects. Before and after the film, EEGs corresponding to open-eye and closed-eye states were measured for approximately 1 min each, and these data were used when normalizing EEG features.

4 Normalization of EEG Features

EEGs show changes in voltage intensity over time within an individual, and base voltage intensity differs among individuals. For this reason, the possibility of misjudgment exists because those values differ greatly even among EEGs with similar waveforms. To solve this problem, linear normalization and non-linear normalization were performed.

4.1 Linear Normalization

This was performed to take into account how EEGs vary over time depending on the subject. Since the eyes were open while viewing the film, linear normalization was performed based on EEG features from the eyes-open state, acquired both before and after the experiment.

EEG feature $Linear\_al_{ij}$, obtained by linear normalization of first EEG feature al_{ij} at a certain point in time during the experiment, is expressed by Formula 4.1:

$$Linear\_al_{ij} = al_{ij} + \left\{ \left(\frac{q_1 - q_2}{p_2 - p_1} \times l + q_2 \right) - \left(\frac{q_2 - q_1}{p_2 - p_1} \times l + q_2 \right) \right\} / 2 \quad (4.1)$$

4.2 Non-linear Normalization

This was performed to take into account the differences in base voltage intensity among individuals.

Non-linear normalized values were determined using Formula 4.2, where $f(x)$ is the EEG feature after non-linear normalization has been applied, x is the EEG feature applied in non-linear normalization, x_{min} is the minimum EEG feature of the individual, and x_{max} is the maximum of the same data. As a result, EEG features with large values are compressed and EEG features with small values are expanded. The degree of intensity of voltage of an individual's EEGs can thus be accounted for.

$$f(x) = \frac{\log(x - x_{\min})}{\log(x_{\max} - x_{\min})} \quad (4.2)$$

5 Evaluation Experiment

5.1 Experimental Method

The following two kinds of evaluations were done. The method of the first kind used was a leave-one-out cross-validation, a technique in which one data point from all test data was extracted and compared with all the remaining data.

This study used 2887 EEG features obtained by excluding outliers from the total of 5670 EEG features. The emotions of the 2887 EEG features used in this study comprised 541 anger features, 726 sadness features, 1226 no-emotion features, and 394 pleasure features.

The method of the second kind is a method of using the EEG features only of the principal subject for the learning data of SVM when leave-one-out cross-validation is done.

5.2 Evaluation of Accuracy

Figure 3 shows the result of the emotion judgement from EEGs. Accuracy of emotion judgment from EEGs using EEG features of all subjects was 43.6% and using EEG features of only principal subject was 55.7%. As a comparison, the result of doing the emotion judgment at random is accuracy of 25.0%. The method proposed herein thus appears valid.

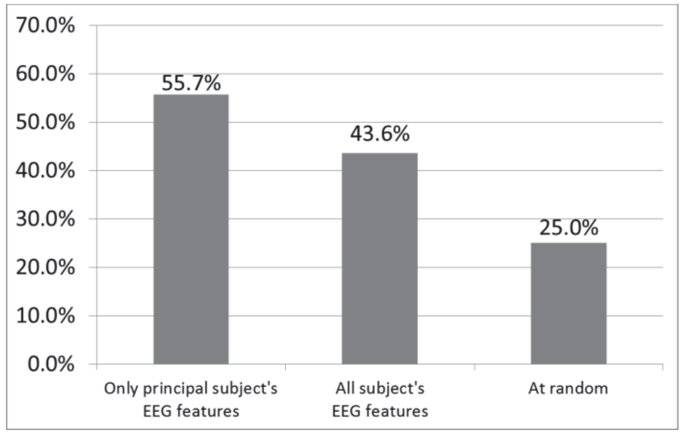


Fig.3 result of the emotion judgement from EEGs

However, performance accuracy remains low, and continued development is required through further development of methods for both reducing noise mixed in with EEGs.

6 Conclusion

We have presented a technique for gauging the emotions felt by a person from EEGs. Accuracy of emotion judgment using EEG features of all subjects was 43.6% and using EEG features of only principal subject was 55.7%. It can be said that a personal adaptation is very effective as the results. However, performance accuracy remains low, and continued development is required through further development of methods for both reducing noise mixed in with EEGs.

We plan to continue research aimed at improving the accuracy of emotion judgment by EEGs in the hopes of developing robotic systems that can participate in conversation and activities while gauging human emotional states.

Acknowledgements

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

References

[1] Chih-Chung Chang, Chih-Jen Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology, 2(3), No.27, 2011.

[2] J. Clin, Guideline therteen : guidelines for standard electrode position nomenclature, American Electroencephalographic Society, Neurophysiol 11, pp.111-113, 1994.

[3] T. Musha, Y. Terasaki, H. A. Haque, G. A. Iranitsky, Feature extraction from EEG associated with emotions, Art Life Robotics, pp.15-19, 1997.

Learning-Based Adaptation Determination Method for Problem Recognition of Self-Adaptive Software

Kwangsoo Seol[1], and Doo-Kwon Baik[2]
[1]Department of Computer and Radio Communications Engineering,
Korea University, Seoul, Republic of Korea

***Abstract*— In this paper, we propose a method for identifying the adaptation period when a problem occurs in a system in order to reduce the unnecessary adaptation of self-adaptive software. Consequently, the dangerous situation information is defined, the behavior information at the time of problem occurrence is learned, and the adaptive performance is determined by comparing it with the existing similar situations by using the k-nearest neighbors algorithm. By the use of the proposed method, a situation where an unnecessary adaptation process is performed while running the self-adaptive system could be avoided, system load may be reduced, and service quality may be enhanced.**

Keywords—Self-adaptive software; Machine learning; Problem recognition

I. INTRODUCTION (*HEADING 1*)

Self-adaptive software-related studies have been conducted for a long time, but a number of current self-adaptive software studies follow a reactive method[1]. In the case where real-world information is collected through a sensor to detect the change in the state of reactive self-adaptive software, and the current state is determined to have a value over a threshold defined within a system, the state is judged to be a dangerous situation and an adaptation is performed. Because such an existing adaptation method determines whether to perform an adaptation without considering the situation that might change in the future, the problem of conducting unnecessary adaptation occurs. If an unnecessary adaptation process is repeated, the system load becomes severe and the service quality declines.

In this paper, we propose a learning-based adaptation period determination method by using a kNN algorithm in order to determine whether to execute an adaptation when it is determined that a problem has occurred in self-adaptive software. Through the proposed method, the system reduces unnecessary adaptation processes with respect to a problem that occurs during the operation, and conducts only the necessary adaptations, thereby cutting the cost required for conducting adaptations.

II. RELATED WORK

MAPE-K, proposed by IBM, is deemed to be representative research on the self-adaptive system model [2]. MAPE-K is a feedback control loop consisting of four-step classes for conducting an adaptation. Each class is composed of the monitoring (M), analyzing (A), planning (P), and executing (E) steps, and the abbreviation MAPE is formed by taking the first letter of each step. In the monitoring step, surrounding environmental information is collected; in the analyzing step, the matter with respect to whether a problem has occurred is determined on the basis of the collected information. If a problem has occurred, an adequate adaptation strategy is identified in the planning step. Finally, the executing step is where the adaptation strategy is applied. "K" in MAPE-K refers to knowledge (K) information required in conducting the above steps. Recently, a number of studies on self-adaptive systems have used the MAPE-K model.

III. PROPOSED METHOD

The system collects state information in real-time through a sensor on the basis of the monitoring elements defined in the design phase during runtime, and analyzes whether the current state has a problem. When analyzing the problem, the existing self-adaptive system compares the value of the collected information with the threshold value. In reality, however, a situation where an adaptation is unnecessary even if the collected value exceeds the threshold may occur.

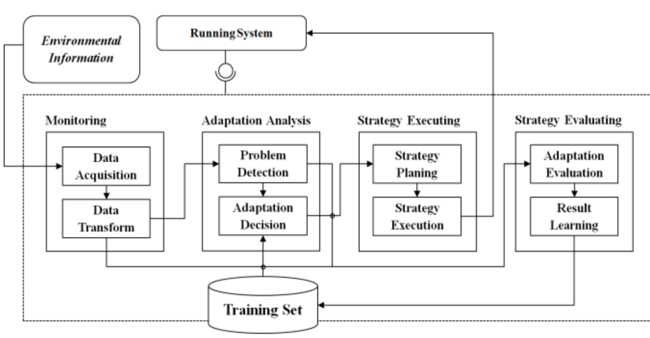


Fig. 1 Schematic representation of the proposed system

In order to overcome such a problem, a method of determining an adaptation period using the probability based on learning is proposed in this paper. The proposed system is composed of a total of four steps as shown in Fig. 1.

A. Definition of dangerous situation information

In the proposed method, a process of determining the learning elements for recording dangerous situation information is required. *Dangerous situation information* is used for finding a similarity between the collected environmental information and the existing training set within the process of determining whether the period when the value of collected environmental information has exceeded the threshold and has led to a dangerous situation is an adaptation period; the *dangerous situation information* is defined in the system design phase. *Dangerous situation information* defines the information on the change in the value obtained until the collected data indicate the dangerous situation. The related example is as follows:

(1) Mean gradient from previous t seconds to dangerous level

(2) Time consumed until the previous dangerous situation

(3) Frequency of danger occurrence for t seconds

Each learning element refers to the behavior information value right before the collected value reaches the dangerous level. The elements defined in this way are additionally required to be verified whether they are significant in the inference through actual learning. To this end, a cross-validation should be conducted after designing the system.

Because each learning element F_i might be distributed in various numerical values, a normalization process is required to measure similarities by using F.

NF_i refers to the normalized value and has values between 0 and 1. The n^{th} dangerous situation information R designed as above can be defined in the following form:

$$R_n = < NF_1, NF_2, \dots, NF_m > \qquad (2)$$

If the values collected in the monitoring step exceed the threshold, each normalized learning element (NF_i) is calculated to generate R_n.

B. Dangerous situation analysis

If the collected data exceed the threshold and lead to a dangerous step, the system executes the adaptation analysis step. In this step, the dangerous situation information R_n and the learned data are checked to find situations similar to the current situation, and the matter with regards to executing an adaptation is determined through the statistics of the result values recorded in the relevant situations.

To this end, the similarity between the learned data and the current situation is measured by using the kNN algorithm and the Euclidean distance measurement method. The similarity measurement equation is as follows: NF_i refers to the "i" normalized learning element of the current dangerous situation information and sNF_i refers to the "i" normalized learning element stored in the learning data.

$$sim(x, y) = \sqrt{\sum_{i=1}^{n}(NF_i - sNF_i)^2} \qquad (3)$$

Based on the measured similarity, the mean value (mv) of result values (D) having k top values is measured. The result values have the value of 0 or 1, and the process of generating the result values is described in Section 3.3. Then, the system determines whether to execute an adaptation by comparing the finally calculated mean value with the reference adaptation value (rv) defined by the user. If mv is larger than rv, the adaptation is executed, and in the opposite case, the adaptation is not executed.

C. Evaluation of adaptation result

The step of evaluating the adaptation of the proposed method is the step of creating learning data. In this step, the matter with respect to whether the system should have or should not have executed an actual adaptation of the result t seconds after a dangerous step is recorded and stored as learning data on the basis of the actual data, not a prediction. The result value D is 1 if an adaptation is necessary and 0 if it is not. The learning data format sR_n stored in the learning storage is as follows:

$$sR_n = < NF_1, NF_2, \dots, NF_m, D > \qquad (4)$$

IV. CONCLUSION

In this paper, we proposed a method for determining the kNN learning-based adaptation period in order to execute an effective adaptation of self-adaptive software. Through this method, the execution of an adaptation can be determined on the basis of not only the current situation but also multi-dimensional elements. The proposed method can contribute to reducing the cost incurred in an adaptation by avoiding the execution of an unnecessary adaptation while running the system.

V. ACKNOWLEDGEMENT

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2012M3C4A7033346). Doo-Kwon Baik is corresponding author

VI. REFERENCES

[1] M Salehie, L Tahvildari. "Self-adaptive software: Landscape and research challenges", Journal of ACM Transactions on Autonomous and Adaptive Systems (TAAS) Vol. 4 no. 2, 2009

[2] Kephart, J.O., and Chess, D.M. "The Vision of Autonomic Computing", Computer. Vol. 36 no.1, 41-50, 2003

An Intelligent Robotic System for Localization and Path Planning Using Depth First Search

Andrea Doucette[1] **and Wei Lu**[1,2]
[1]Department of Computer Science, Keene State College, USNH, Keene NH USA
[2]Department of Electrical and Computer Engineering, University of Victoria, Victoria BC Canada

Abstract - *LeJOS is an open source project created to develop a technological infrastructure and a tiny Java virtual machine to develop software into Lego Mindstorm Products using Java technology. LeJOS has been widely applied to programming Lego robotic system since it was created in 2006. In this paper we propose and develop a new depth first search algorithm and integrate it into the LeJOS system so that the intelligent robotic system is able to complete the localization and path planning automatically. Such a new depth first search algorithm is a significant improvement to the current path finding class provided by the LeJOS and experimental evaluations with the Mindstorm robotics system show that our approach can achieve the goal state intelligently in a short time.*

Keywords: LeJOS, Depth First Search, Java Virtual Machine

1 Introduction

Searching falls under Artificial Intelligence (AI). A major goal of AI is to give computers the ability to think, or in other words, mimic human behavior. The problem with this mimicry is that unfortunately, computers don't function in the same way as the human brain: they require a series of well-reasoned out steps in order to find a solution. Therefore our goal is then to take a complicated task and convert it into simpler steps that the robotics system can handle. That conversion from something complex to something simple is what the search algorithm would do [1]. In this paper we propose and implement a new depth first search (DFS) on a tiny Java Virtual Machine (JVM), called LeJOS [2]. LeJOS is an open source project created to develop a technological infrastructure in which Java technology is applied for programming software for robots. Java is an Object Oriented programming language and one of the most important features implemented in LeJOS is the *LeJOS navigation API* that can be used to achieve the goal in which a convenient set of classes and methods provided to control the robot. The classes that control vehicles deal with several levels of abstraction. At bottom, there are the motors that turn the wheels, controlled by the *NXTRegulatedMotor* class. The *DifferentialPilot* class uses the motors to control elementary moves: rotate in place, travel in a straight line, or travel in an arc. At the next level, the *NavPathController* uses a *DifferentialPilot* to move the robot through a complicated path in a plane. To perform navigation, the path controller needs the robot location and the direction it is heading. It uses an *OdometeryPoseProvider* to keep this information up to date.

The contributions of the paper mainly include (1) a new depth first search (DFS) algorithm that can be applied to build arbitrary tree structures generically and (2) applying and integrating the proposed DFS algorithm in the LeJOS based robotics system for localization and path planning, enhancing the existing pathfinding approaches within LeJOS system. The rest of the paper is structured as follows. Section 2 introduces the depth first search algorithm. Section 3 presents our proposed leJOS based DFS algorithm. Section 4 outlines the experimental evaluation settings and results for our system. Finally Section 5 makes concluding remarks and outlines our future work.

2 Overview of depth first search

Let's first learn how we humans would solve a search problem. First, we need a representation of how our search problem will exist. The following Figure 1 is an example of our search tree. It is a series of interconnected nodes that we will be searching through:

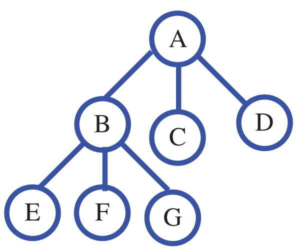


Figure 1. Tree structure of a path.

Depth first search works by taking a node, checking its neighbors, expanding the first node it finds among the neighbors, checking if that expanded node is our destination, and if not, continue exploring more nodes. For example if we want to find a path from A to E, we can use two lists to keep track of what we are doing - an *open list* and a *closed List*. An Open list keeps track of what you need to do, and the Closed List keeps track of what you have already done. At the beginning, we only have our starting point, node A. We haven't done anything to it yet, so let's add it to our *open list*. Then we have *open list* including <A> and *closed list* including <empty>. Now, let's explore the neighbors of our A node. Node A's neighbors are the B, C and D nodes. Because we are now done with our A node, we can remove it from our *open list* and add it to our *closed list*. Then our current *open list* include <B,C,D> and *closed list* contain <A>. Now our *open list* contains three items. For depth first search, you always explore the first item from our *open list*. The first item in our *open list* is the B node. B is not our destination, so let's explore its neighbors. Because we have now expanded B, we are going to remove it from the *open list* and add it to the *closed list*. Our new nodes are E, F and G, and we add these nodes to the beginning of our open list. Then we have *open list* including <A,B> and *closed list* including <E,F,G,C,D>. We now expand the E node. Since it is our intended destination, we stop. Therefore we receive the route A->B->E that is interpreted from the *closed list* by using the regular depth first search algorithm.

3 LeJOS based DFS algorithm

In our LeJOS based DFS algorithm each node on the path is a class node called WPNode defined as:

```
public WPNode(String newname, WayPoint newwp) {
   nodename = newname;
   nodewp = newwp;
   seen = false;
   parent = this;
   connections = new ArrayList<WPNode>();
}
```

The pseudocode for the LeJOS based DFS algorithm is described in the following:

(1). Constructing the generic tree for search space, such as:

```
A = new WPNode("A", new WayPoint(0, 0));
B = new WPNode("B", new WayPoint(-5, 5));
C = new WPNode("C", new WayPoint(5, 5));
A.addLeaf(B);
A.addLeaf(C);
```

(2). Declare a stack to save the route path, such as:

```
Stack<WPNode> DFSpath = new Stack<WPNode>();
```

(3). Set the current node to root node, say *A*. While the destination node is not found, loop the following:

a. if current node has children, set first node unseen node to current node then return;

b. if current node has no unseen children, set its parent to current node then return;

(4). Once the destination node is found, push the destination node to the stack and then push each parent node to the stack;

(5). Generate pilot for two-motor movement and then set pilot to use appropriate dimensions and motors

(6). Pop waypoint of each path node and apply goto method to direct robots moving to the next node.

The completed source code package can be found at [3] in more details.

4 Experimental evaluation

As illustrated in the Figure 1, we evaluate our Mindstorms robotic system to find the path from both *node A* to *node G* and *node A* to *node F*. Lego Mindstorms NXT is an educational product designed to build easy robots with an intelligent, computer-controlled NXT brick that lets a robot come alive and perform different operations [4]. In our experiment, the coordinate at A is (0,0), the coordinate at B is (-5,5), the coordinate at C is (0,5), the coordinate at D is (5,5), the coordinate at E is (-10,10), the coordinate at F is (-5,20) and the coordinate at G is (0,20).

The path from *node A* to *Node G* is A → B → G and path from *node A* to *Node F* is A → B → F, the Mindstorms robotics system with LeJOS DFS algorithm traverses each node until it reaches the destination G and F, respectively, showing the algorithm works successfully in the LeJOS JVM.

5 Conclusions

In this paper we propose and implement a LeJOS based DFS algorithm running under the LeJOS JVM. Experimental evaluation shows that our algorithm works successfully to achieve its goal within a short time. In the near future we will implement the Breadth First Search (BFS) and Heuristic based Hill Climbing algorithms to improve the current LeJOS path finding class. To the best of our knowledge the proposed work is the first attempt to improve the path finding class under the LeJOS JVM by integrating more generic searching algorithms in the LeJOS API functions.

6 References

[1] Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, 2009.

[2] LeJOS, http://www.lejos.org/ retrieved in Mar. 28 2015.

[3] LeJOS DFS, http://sl.keene.edu/lejosDFS.zip

[4] Lego Mindstorms NXT 2.0, http://www.lego.com

Design and Implementation of a Simulator for the Operational Control of Creative 3D Assembly*

Sanguk Noh
School of Computer Science and Information Engineering
The Catholic University of Korea
43 Jibong-ro, Wonmi-gu, Bucheon-si, Gyeonggi-do 420-743, Republic of Korea
E-mail: sunoh@catholic.ac.kr

Abstract - *This paper presents the design and development of a simulator for the operational control of creative 3D assembly. We model the creative 3D assembly as a moving agent, which generates a sequence of actions according to the scripts previously defined at any given domain. To guide or control the operation of the creative 3D assembly, we formulate logical rules represented as if-then rules into knowledge base, and construct various knowledge bases for a series of 3D assemblies in different settings. We design and implement our simulator including programming blocks, which tests the operational control of 3D assemblies in simulated settings.*

Keywords: Object-oriented modeling and simulation, Creative 3D assembly, Definition and interpretation of behaviors, Operational control using reactive rules

1 Introduction

In this paper, we design and implement a simulator for the operational control of creative 3D assembly. The simulator shows the creative 3D assembly's behavior that maps any given percept sequence to an action. We model the creative 3D assembly as a moving agent, which generates a sequence of actions according to the scripts previously defined at any given domain. To guide or control the operation of the creative 3D assembly, we formulate logical rules represented as if-then rules into knowledge base [1], and construct various knowledge bases for a series of 3D assemblies in different settings. We design and implement our simulator including programming blocks, which tests the operational control of 3D assemblies in simulated settings.

The paper is organized as follows. In the following section, for the operational control of creative 3D assembly, we describe two kinds of programming blocks, which consist of behavior blocks and control blocks. The behavior blocks define a set of available actions at a given domain, and the control blocks decide what actions are chosen, when they are performed, and how many times they are executed. In the concluding section, we summarize the preliminary implementation of our simulator and discuss further research issues.

2 Defining behaviors using programming blocks

Our work is built on efforts by several other research groups who focus on making educational programming language, such as SCRATCH, which is developed by the Lifelong Kindergarten Group at the MIT [2, 3]. We also develop "drag-and-drop" programming blocks for the operational control of creative 3D assembly in simulated settings.

To define the behaviors of the 3D assemblies and test their operational control, our approach uses a purely reactive reasoning procedure [4, 5, 6, 7]. The goal of reactive systems is to directly respond to the condition with a predefined action. These systems could react very quickly to environmental conditions. We formulate logical rules represented as if-then rules into knowledge base, and construct various knowledge bases for a series of 3D assemblies in different settings.

We provide programming blocks with users to experience definition of behaviors for creative 3D assemblies in a virtual environment. For the operational control of creative 3D assembly, we design and implement two kinds of programming blocks, which comprise behavior blocks and control blocks. The behavior blocks define a set of available actions at a given domain, and the control blocks decide what actions are chosen, when they are performed, and how many times they are executed. Three panels are shown in Fig. 1. The leftmost panel presents a set of behaviors, for example, move forward, turn some degrees in a counterclockwise and in a clockwise direction, grab, and so on. The mid panel shows control programming blocks, for example, condition block and loop block. The rightmost panel illustrates the combination of action blocks and control blocks, and their sequence as a result. The "drag-and-drop" programming blocks, as depicted in Fig. 1, generate the scripts for the

* This work has been supported by the ICT R&D program of MSIP/IITP, Republic of Korea, [2014-044-063-003, "Development of the graphic engine and framework of 3D contents for the creative hands-on science education"].

operational control of creative 3D assembly in a simulated environment.

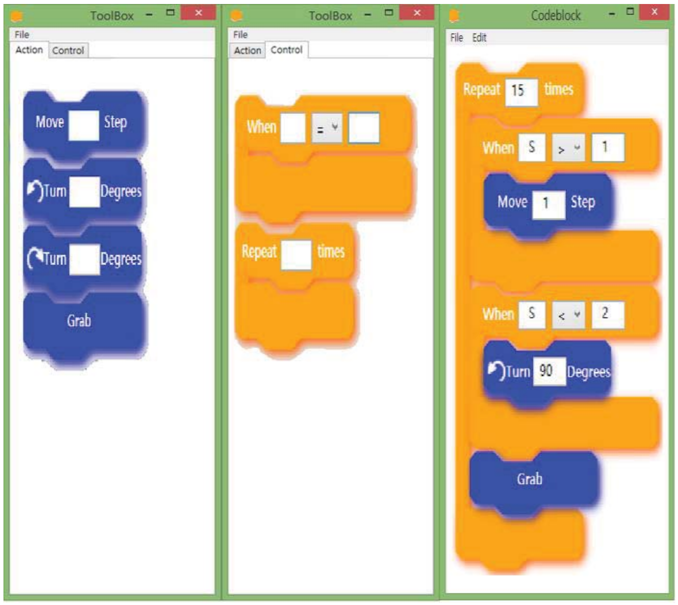


Fig. 1. Three panels representing action programming blocks, control programming blocks, and their scripts as a resulting sequence.

Fig. 2 illustrates a simulated testbed in which agents in the form of 3D assemblies can show their behaviors given a specific environment. The agent can move forward, turn left and right by 90 degrees, grab the ball in a 10×10 cell of rooms, as depicted in Fig. 2. If the agent keeps moving forward and bumps into a wall, then the agent does not move. Users can test the operational control of creative 3D assemblies along the various dimensions given a simulated environment, before they are made in real shape using 3D printers.

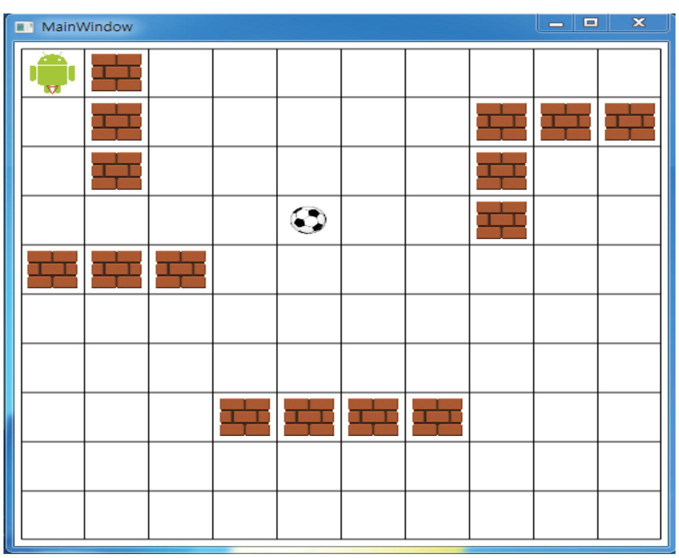


Fig. 2. Our simulator including a moving agent, an array of walls, and a ball.

3 Conclusions

In a virtual reality environment, we model the creative 3D assembly as a moving agent, which generates a sequence of actions according to the scripts previously defined at any given domain. To guide or control the operation of a specific 3D assembly, we formulated logical rules represented as if-then rules into knowledge base. The prototype of our simulator was designed and implemented, and the behaviors of 3D assembly defined by "drag-and-drop" programming blocks were also tested by in a simple grid environment. We are working on the expansion of the set of behaviors, and the capabilities of our simulator.

Based upon the preliminary implementation of our system, we will develop and construct various knowledge bases for 3D assemblies, and apply them to the assemblies built with 3D bricks, which are made for creative science education. Further, we will extend our inference system for the fuzzy operational control of creative 3D assemblies. Given vague input variables, our system using fuzzy logic [8, 9] could be used to control fuzzy operations, for instance, steering, accelerating, and braking, which usually could happen in operational control settings. Our inference system to support the assembly of 3D bricks and the operational control of creative 3D assembles, as well, then can be equipped with both crispy rules and fuzzy rules to be practical in real-world domains.

4 References

[1] Hao, C., *Research on Knowledge Model for Ontology-Based Knowledge Base,* In Proceedings of International Conference on Business Computing and Global Informatization, pp. 397-399, 2011.

[2] MIT News, *Kids coding in the cloud*, May 2013.

[3] SCRATCH, http://scratch.mit.edu/, 2015.

[4] Agre, P.E. and Chapman, D., *Pengi: An implementation of a theory of activity*, In Proceedings of the National Conference on Artificial Intelligence, Seattle, Washington, pp. 268-272, 1987.

[5] Fox, J. and Krause, P., *Symbolic decision theory and autonomous systems*, In Proceedings of the 7th Conference on Uncertainty in Artificial Intelligence, UCLA, California, pp. 103-110, July 1991.

[6] Abraham, A., *Intelligent Systems: Architectures and Perspectives,* Recent Advances in Intelligent Paradigms and Applications, Studies in Fuzziness and Soft Computing, Vol. 113, Springer Verlag, Germany, pp. 1–35, 2003.

[7] Giarratano, J. and Riley, G., *Expert Systems: Principles and Programming*, Fourth Edition, Course Technology, October 15, 2004.

[8] Cordon, O., Gomide, F., Herrera, F., Hoffmann, F. and Magdalena, L., *Ten Years of Genetic Fuzzy Systems: Current Framework and New Trends*, Fuzzy Sets and Systems, 141(1):5-31, Elsevier, 2004.

[9] Zadeh, L.A., *Toward extended fuzzy logic—A first step,* Fuzzy Sets and Systems, 160(21):3175-3181, Elsevier, 2009.

SESSION

ARTIFICIAL INTELLIGENCE: THEORY, ALGORITHMS AND APPLICATIONS + COGNITIVE SCIENCE + MODELING

Chair(s)

TBA

Visual Intelligence: Toward Machine Understanding of Video Content

Michael C. Burl, Russell L. Knight, Anthony C. Barrett
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109

Abstract - *This paper describes progress toward developing visual intelligence algorithms (VI) that can produce human-like text descriptions (captions) from video inputs. Video frames are assumed to be generated according to an underlying "script' that specifies a camera model and the content and action in a scene. VI is formulated as the problem of recovering the script (or relevant portions of the script) given a sequence of video frames. Three types of scripts at different levels of abstraction are recovered: C-scripts contain object detections, poses, and descriptive information on a frame-by-frame basis; B-scripts assign persistent IDs to objects across frames and "smooth" frame-by-frame information; A-scripts provide a symbolic representation of video content using a sparse timeline in which Planning Executing Agent (PEA) graphical models (behavior snippets) are associated with agents in the scene. From the script representations, a compact text description (caption) of the action in the scene, as well as an envisionment (3D rendering) showing what the algorithm believes happened, can be generated. Scripts have been derived automatically and evaluated on a set of 240 publicly available video vignettes containing over 100,000 frames.*

Keywords: video understanding, natural language, text description, caption, surveillance, behavior recognition.

1 Introduction

On the TV show *Jeopardy*, IBM's *Watson* provided convincing proof that a machine could answer challenging natural language questions on par with, or even better than, human experts [1]. Emerging consumer products, such as *WolframAlpha* [2] and Apple's *Siri* [3], have also shown significant progress on this aspect of AI. We are interested in the related, but arguably more difficult, problem of visual intelligence (VI). *Can a machine, given only video input, reliably answer complex natural language questions about the content of a video and/or produce human-like text descriptions of what it has seen?* In this paper, we target automatic generation of text captions. Unlike the *Jeopardy* problem, simply looking up and conjoining readily available facts from the Internet is not likely to produce a good caption for a specific video input. *(*However, Barnard and Forsyth [4] did achieve some early success with text and image feature co-learning.)

A robust VI capability will provide the foundation for a number of new applications. For the military, placing VI-enhanced, persistent surveillance on unmanned air and ground vehicles could provide situational awareness without endangering personnel or requiring a large number of human eyes to monitor video feeds [5]. Similar benefits could be expected in law enforcement and homeland security. Other applications include human-robot interaction, video indexing and retrieval, sports analysis, retail intelligence, elder care, video games, and anonymization of video.

2 Approach

Our approach combines a front-end computer vision pipeline that leverages state of the art work in human detection, pose estimation, object recognition, and tracking with a back-end, AI-based plan recognition system that uses Planning Executing Agent (PEA) graphical models to recognize and reason about higher level behaviors.

The basic assumption in our approach is that video frames are generated according to a "script". One can imagine that such a script would specify a camera model and the content and action in the scene. VI is the inverse problem of recovering the script (or relevant portions thereof) given the sequence of video frames. Three types of scripts at different levels of abstraction are employed. At the top (agent) level, "A-scripts" built from Planning Executing Agent (PEA) graphical models are used to represent and reason about agent behaviors and video content symbolically. At the mid (tracking) level, "B-scripts" consisting of object pose trajectories provide a more literal encapsulation of what happened in the scene. At the low (detection) level, "C-scripts" consist of information extracted on a frame-by-frame basis.

Figure 1 illustrates the forward and inverse problems. The top path in the figure flows from right to left, taking a high-level A-script as input and producing video frames as output. One can envision this process occurring in two stages: first, the physical state of the actors and objects in the scene are altered over time (either in physical reality or in a simulation) according to the script; second, the world in its updated state is imaged.

The bottom (VI) path flows from left to right taking as input a sequence of frames and generating an estimate of the original

A-script. Computer vision attempts to invert the imaging process and recover a description of the state of the world versus time; the intermediate C-script and B-script representations describe the state in terms of the physical properties of the agents and objects, e.g., size, position, orientation, as well as any internal pose parameters versus time. Plan recognition techniques are used to recover the A-script that was responsible for this evolving state. The result is a higher-level representation of the video content that associates PEAs embodying specific semantic concepts with agents.

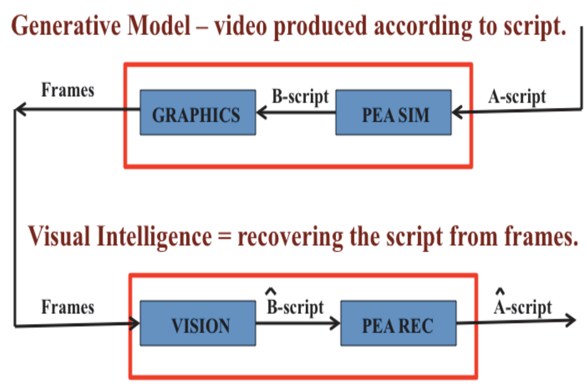


Figure 1. Video frames are produced according to an underlying script. Visual intelligence is formulated as the problem of recovering the script from the frames. Three levels of abstraction are used: A-scripts, B-scripts, and C-scripts. (C-scripts are used internally in the vision block.)

3 Related Work

The survey paper by Aggarwal and Ryoo [6], which served as the basis for a CVPR tutorial [7], provides an extensive discussion and taxonomy of work in activity recognition. Under their taxonomy, our approach would be classified as a hierarchical description-based approach.

The earlier work of Hongeng, Nevatia, and Bremond [8] is similar in some respects to ours. One difference is that we do not rely on temporal change detection to locate people and objects; we directly detect people and objects on a frame-by-frame basis making our approach suitable for a moving platform or a scanning camera. While their activity representation is based on 2D shape and trajectory features, our approach includes detailed human pose information allowing us, for example, to identify and describe colors of individual pieces of clothing. More importantly, our representations are *fully generative*, meaning we preserve enough information to create an envisionment (3D rendering versus time) that captures the essence of the original video. Other approaches from DARPA's Mind's Eye program can be found in [9-13].

A key difference between our work and many others is that others tend to use finite state machines and "Allen" temporal logic relations [14] to model and recognize complex behaviors involving multiple actors or actors and objects. In our approach we use PEAs, which are general graphical models in which the nodes represent states, and arcs represent transitions between states. PEAs subsume state machines and Markov models, but are more powerful because PEAs provide an explicit representation of resources through member variables. Resources are values that change by persisting in, entering, or exiting a state. Resource "gateways" can be established that prevent transition to other states until a resource collection requirement is met. Our system allows for the efficient encoding of such a resource collection state without adding linearly many states per resource, or exponentially many states for combined resources, as would normally be required by a simple finite state automaton. Our pre-compilation of resource goals allows us to plan, in constant time, to achieve required resource levels.

PEAs also provide more flexibility by associating arbitrary predicates with the arcs. For Hidden Markov Models (HMMs), which have been widely used for activity [15,16] and speech recognition [17], the transitions from state-to-state happen randomly according to some fixed probability distribution. In PEAs transitions are triggered by predicates, which can be complex (probabilistic, if desired) rules implemented as *general procedural computer code*, and as noted above may incorporate resource constraints. This flexibility enables PEAs to correctly model multi-agent interactions, e.g., the action of an agent can force another agent to make a state transition.

PEAs are hierarchical allowing complex behaviors to be built up from simpler behaviors. A node in a PEA graph can correspond to another PEA graph that represents a subordinate behavior. Figure 2 shows a PEA model for the verb GIVE, which makes use of subordinate concepts such as HOLD, APPROACH, RUN, and WALK.

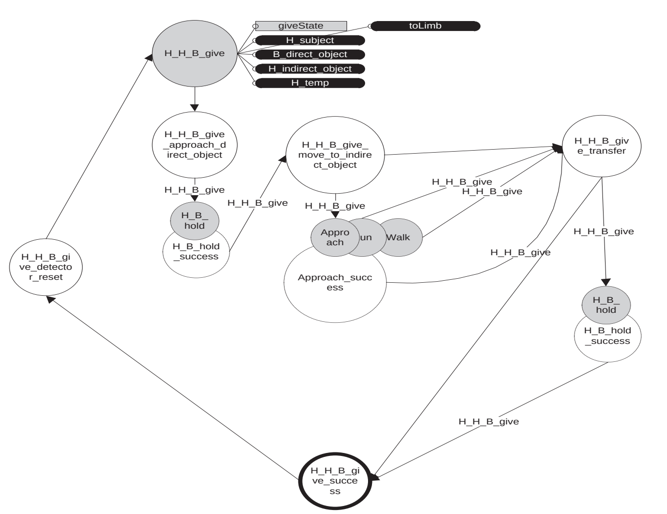


Figure 2. PEA model for GIVE in which subject is a human, direct object is a ball, and the indirect object is a human. The model for this complex verb is hierarchical making use of subordinate verbs such as HOLD, APPROACH, RUN, and WALK.

There are some preliminary VI efforts moving toward commercial applications in "video analytics" [18]. Some companies, such as Brickstream Corp. [19], provide analysis of overhead video taken in retail store scenarios, e.g., to detect shoplifting and to aid in marketing. Sports video analysis, especially for team sports such as football and soccer, has received significant attention in the academic research community. However, it is common in sports domains to use multiple cameras deployed in favorable vantage points. Work on activity recognition with camera systems that provide 3D range measurements, e.g., the KINECT sensor or stereo vision systems, as well as sensor-based activity recognition have shown some success. The annual NIST-sponsored TRECVID competition [20] also supports development of VI capabilities. *Compared to existing work, however, we target a more general-purpose VI capability suitable for a monocular camera in unconstrained environments with a richer range of agent appearances and behaviors.*

4 From Video to B-scripts

Figure 3 shows the vision pipeline that we use to convert raw video into a B-script representation. In addition to the raw video, the pipeline takes as input the known[1] camera model consisting of the camera intrinsic parameters (focal parameters, skew, and principal point) and the camera extrinsic parameters (position and orientation with respect to the world coordinate system).

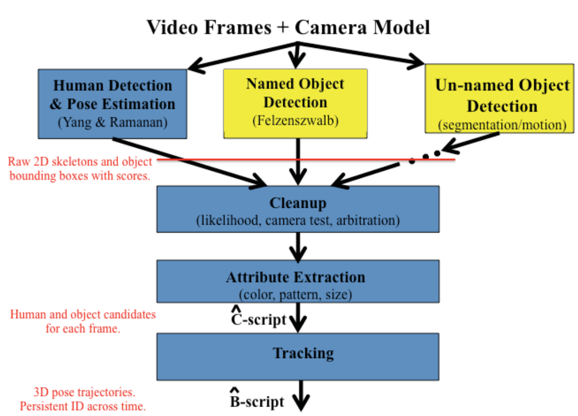


Figure 3. Vision pipeline used to recover the C-script and B-script representations from raw video frames. Note that we assume the camera model is known a priori.

Detection: There are three detection branches or pathways through the pipeline, which we refer to as the human pathway, the named object pathway, and the un-named object pathway. The human pathway is based on the human detection and 2D pose estimation algorithm of Yang and Ramanan [21]. This approach uses a training set to learn part detectors and a tree-structured model of the geometrical layout of the parts. The resulting human detector produces skeleton hypotheses, which delineate believed positions in the image plane of various body parts, along with an overall log-likelihood score that combines the part scores and geometry score. The named object pathway is based on the Deformable Parts Model (DPM) object detection algorithm of Felzenszwalb et al [22]. It is also based on learning part detectors and their geometrical layout from a training set. The output is a set of object hypotheses consisting of 2D bounding boxes and log-likelihood scores. DPM is most effective for objects that have a well-defined characteristic structure such as cars, motorcycles, bicycles, etc. It is less useful or robust for objects that are: rare (for which there may not be a pre-trained model on hand), highly variable in appearance or deformable in structure (e.g., handbags), or relatively small (few pixels) compared to the image resolution. The un-named object pathway is intended to catch these types of objects for which there is not a reliable, pre-trained detector. The un-named object pathway was not used in the experiments of Section 6.

Cleanup: Following the top-level detection blocks, there is a "cleanup" stage in which the candidate humans and objects are pruned. This stage includes thresholding the likelihood scores, applying a "camera test" to any human detections to ensure that the implied physical sizes under the known camera model are reasonable, and arbitration, which is a form of non-maximum suppression to eliminate or reduce multiple overlapping detections of people or objects. Arbitration is applied separately to humans and objects. The highest scoring detection is allowed to claim real estate in the image. Lower scoring detections are rejected if their area of support overlaps too much with an existing higher-scoring detection. This process continues until all detections meeting a minimum score threshold have been considered.

Attribute Extraction: This stage measures various properties of the surviving candidates that will be useful for composing a description and for tracking. This stage also involves situating the candidates in 3D space. For human candidates the Levenberg-Marquardt (L-M) nonlinear optimization algorithm is used to lift the 2D skeleton representation into a full 3D joint angle representation of pose. The optimization adjusts the joint angle parameters of an articulated humanoid model[2] to bring the projected positions of the model joints into agreement with the 2D image plane observations of joint positions (junctions between links in the 2D skeleton). The errors in joint position projections are insufficient to uniquely determine the joint angles. This fact is clearly shown in the work by Taylor [23], which highlights that there is a depth sign ambiguity for each

[1] For arbitrary videos downloaded from websites such as YouTube, the camera model may not be provided; we assume

[2] We use a 17-bone model for the skeleton. Internal pose consists of 36 degrees of freedom with 6 additional degrees of freedom for the overall position and orientation with respect to world frame.

link (‘bone”) in the humanoid model. We have added several ad hoc constraints (freezing some degrees of freedom such as trunk torsion in the humanoid model, adding penalty terms to keep one foot close to the ground plane, allowing an overall scale factor, etc.), but incorporating a stronger prior model of probable joint angle configurations as in [24] and leveraging pose information from temporally-nearby frames could improve the results. The Levenberg-Marquardt optimization is run with four initial seeds corresponding to different facing directions. The best scoring result yields an estimate of the position and orientation in 3D, as well as an overall skeleton scale factor and the internal pose parameters (joint angles). For monocular localization of the humans in 3D, we found that the L-M whole body procedure was more robust than reverse ray-tracing feet pixels to the ground plane, since foot estimates from the pose estimator tend to be unreliable.

In addition to the joint angles, colors are extracted for each body part of the human detections. The colors are found over a rectangular region centered along the corresponding “bone” in the 2D skeleton estimate.

For objects, the algorithm currently extracts a 3D position and a single average color for the object. We have yet to implement an estimator for object pose or for the physical size of objects. Even estimating the 3D position of objects is non-trivial since it requires reasoning about support relationships. Is the object supported by the ground, by a human, by another object, or is it free flying? Currently, if the bounding box of an object intersects with a dilated version of a detected human, then we assume the object is at the same depth from the camera as the human. We then reverse ray-trace the object’s position in the image plane to a vertical plane in the world that is at the same depth from the camera as the human. For objects that are not close enough to be in a support relationship with a human, we ray-trace the bottom of the object’s bounding box in the image plane back to the horizontal ground plane to determine its 3D coordinates.

Although this object support logic works well most of the time, there are still some problem cases. One situation is when the detection of a human carrying an object is unreliable so the human detection drops out in some frames. In these cases, the object is not deemed to be in a support relationship and is projected out into the distance onto the ground plane rather than being put closer but off the ground. Another situation occurs if there is an object on the ground in the distance beyond the person. When the two bounding boxes in the image plane get close together, the object is brought forward to the human’s depth.

The output from the attribute extraction stage is a C-script, consisting of frame-by-frame detected humans and objects with color and 3D position information, and estimated pose (humans only for now). Each appearance of a human or object in a frame is given a separate label.

Tracking: A tracking algorithm is applied to upgrade C-scripts into full-fledged B-scripts in which the humans and objects maintain a persistent identity (trackID) across time. We currently use a greedy data association strategy based on a combination of distance in world coordinates, distance in pixel coordinates, and distance in color space to match detections from a new frame with previously seen agents/objects (tracks). We plan to incorporate a multiple hypothesis tracker (MHT) [25,26] in future work.

5 From B-scripts to A-scripts

Recovering an “A-script” from a video clip enables the system to not only recognize what happened, but also why it happened, what is likely to happen next, etc. An A-script consists of a sparse timeline in which parameterized behaviors (PEA graphs) are associated with agents.

5.1 Planning Executing Agent (PEA) Models

As shown earlier in Figure 2, PEAs are graphical models consisting of a set of resources, states (nodes), and transitions between states (arcs). PEAs have provided the core agent reasoning capability in several real time games and simulations. PEAs are computationally lightweight enabling simulation of millions of agents in a real-time, distributed, massively multiplayer game; the same PEA models can be used both to simulate and to recognize behaviors.

Transitions in PEA graphs are regulated by a set of predicates that indicate whether particular conditions required for transition are true. For example, if an insurgent is in the **observe** state and is notified that an IED is arriving, he immediately transitions to the **egress** state. Resource requirements may be imposed as part of a predicate, e.g., a shopper is not allowed to transition to the **buyingFromMerchant** state unless *money* > 0. Predicate values can be produced by random number generators or any user-defined procedural computer code.

Goals levied on PEAs come in two basic types: resource goals and state goals. A resource goal is an expression about one of the resources that should be achieved during the execution of the PEA, e.g., money > 100. A state goal is merely a state in the PEA graph, e.g., **playing_tag**. These basic goals can imply subgoals. For example, if we have a goal that *food* > 10, but transitioning into a food-increasing state requires that money > 10, then we have a subgoal that *money* > 10. The process of PEA compilation provides a set of lookup tables that allow us to know what our next step should be, given our current state, resources, and goals. More specifically, PEA compilation provides tables that are indexed in goal-state pairs for each PEA. These tables represent what edge a PEA should prefer or take, if given the opportunity, given the state and goal being achieved. Note that a goal could be another state or a resource request.

The PEA model in Figure 2 represents the complex verb, *give*, which involves a subject, direct object, and indirect object. The hierarchical nature of the PEA framework allows complex concepts to be built up from simpler concepts. Many subordinate verbs may also be observed during the execution of a complex transaction.

PEA models are hand-constructed as opposed to learned from data. Since the state transitions are governed by arbitrary computer code, there is no compact way to present full details on all the PEAs used in our system (short of listing the code).

5.2 Recognizing Behaviors

To determine which PEA/goal combination best describes a series of observations, we must first map observable actions to states in each PEA. This process is via notification from the vision system that certain predicates, such as movement and proximity to other agents, have occurred. Given this mapping, we can compute the likelihood that one PEA/goal combination is more likely than another by computing the odds of each transition. Gaps in knowledge about an agent are patched by adding a goal at the start of the gap that gets us to the state at the end of the gap. *Transitioning through a PEA graph to the end state of a particular verb means that verb is believed to have occurred.*

PEAs can also be used to project future behavior by simply simulating the PEA in its environment. Multiple runs that include all agents will provide a set of execution traces. The accuracy of conclusions on the likely short-term future behavior depend highly on how probabilistic the environment is and how faithful the model is to reality. Projection has been demonstrated against human opponents in games and is reliable for determining interception paths and likely subgoals the human is trying to achieve.

6 Experiments and Results

We have applied our system to the 240 videos in the mindseye-y1-description-task available from visint.org [27]. No camera models or calibration data, e.g., checkerboard images [28,29], are available for these videos; hence, approximate camera models for each video were derived offline through various methods (horizon line, assumption of standard human height, etc.). A Matlab mex implementation of the Yang-Ramanan pose estimation algorithm was applied to every frame of every video with the output results (2D skeletons) saved in files. The part detection models were trained on a completely different video corpus. A third party [30] kindly provided object detection results from the Felzenszwalb DPM algorithm for every frame of every video. The DPM models were trained on a development corpus that was similar in style and content to the evaluation corpus.

One quirk with the object detection results, however, was that only the DPM-based detectors for objects known to occur in a particular video were applied. For example, if a video contained two people and a bicycle, only the DPM detector for "bicycle" was applied. (DPM-based human detections were not used; only Yang-Ramanan results were used for human detection.)

6.1 Envisionments and Text Descriptions

An advantage of our approach is that the scripts we derive are fully generative; they can be reformed into a graphical display (synthesized movie) called an *envisionment* or *playback*. Figure 4 (top) shows a frame of an envisionment constructed from an automatically extracted B-script.

Figure 4. (Top) A single frame in a B-script reconstruction of a video. The background is a textured canvas, while the humans and bicycle are full-fledged 3D objects. (Bottom) Original frame.

Another result is shown in Figure 5, which is a hybrid between a B-script and A-script: the detailed pose trajectories from the B-script and the text from the A-script are combined. The text annotations say: "Addison stopped", "Bailey walking", "Bailey held the dark olive green skateboard", "Bailey carried the dark olive green skateboard" and the skateboard itself is labeled with "the dark olive green skateboard". The names of the people are arbitrarily assigned (first person is assigned a name that starts with "A"; there is no gender recognition or face recognition at present). Further information readily available in the script indicate that Bailey is the person in the blue shirt. We have, in separate work, demonstrated noun-adjective queries over the video corpus such as "find bicycle and red shirt".

The PEAs currently implemented do not include a concept of "ride", so the system (somewhat incorrectly) concludes that Bailey is walking, based on his rate of travel, and that he is holding/carrying the skateboard since it travels with him. The people and the skateboard are full-fledged 3D mesh objects that are rendered according to the camera model. Only the position and color of the skateboard were derived from the image data, however; the size, orientation and CAD model are based on default values. Pose estimation for inanimate objects is intended for future work.

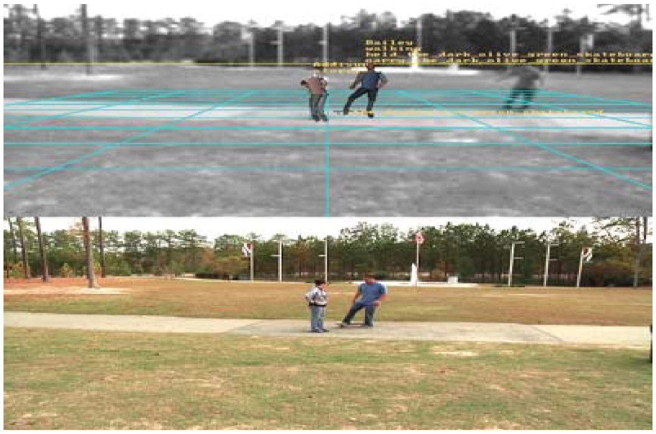


Figure 5. (Top) Hybrid AB-Reconstruction. (Bottom) Original Frame. The text annotations say: "Addison stopped", "Bailey walking", "Bailey held the dark olive green skateboard", "Bailey carried the dark olive green skateboard" and the skateboard itself is labeled as "the dark olive green skateboard". Bailey is the person in the blue shirt.

6.2 Quantitative Performance Metrics

While envisionments are useful for qualitative performance assessment, it is desirable to have quantitative metrics as well. Currently, we have only systematically assessed the person detection performance, which combines the Yang-Ramanan detector with the "cleanup" described in Section 4. Head positions reported in the C-scripts were projected back to image coordinates and compared to manually clicked ground truth locations. If the C-script head circle[3] encompassed the ground truth location, it was counted as a correct detection.

Table 1 shows the head-detection performance over the 113,268 frames of the corpus. Ignoring the *FAs* column for the moment, the entry in (row=*i*, column=*j*) is the *number of frames* for which there were *i* people present in the ground truth and the algorithm correctly detected *j* of them. The (0,0) entry is simply the *number of frames* in which there were no people present. The FAs column is the *number of false alarms* given that the true person count of the frame was *i*. Converting into normalized values we see that the conditional probability of correctly detecting one person when only one person is present is about 64.4%. The conditional probability of correctly detecting two people when two people are present is only 30.41%. The probability of detecting a person (without conditioning on the number of people present) is 60.4%. The average number of false positives per frame over all situations is about 0.3751. Videos where the person detection rate is below 20% typically have the person in an elevated position relative to the ground (e.g., on a ladder, atop stairs, on a fence) or sitting on the ground. These comprise about 12% of the videos in the corpus.

[3] Because the manual clicking was only done every 10th frame with interpolation between, we expanded the head radius by a factor of 1.5 for scoring.

	0	**1**	**2**	**3**	**4**	**FAs**
0	19105	-	-	-	-	4900
1	26507	47993	-	-	-	28876
2	4537	8335	5625	-	-	8159
3	52	381	694	8	-	498
4	3	11	8	8	1	55

Table 1. Human (head) detection performance from C-scripts evaluated over 113,268 frames of the corpus. The entry in row *i* and column j is the *number of frames* in which j persons were correctly detected given that i persons were present. The *FAs* column is simply the total number of false alarms (not the number of frames) that occurred in situations where *i* people were present in the frame. For example, from the row with *i*=4, one can conclude that only 31 frames had 4 people present and in only one of those frames did the algorithm correctly detect and localize all 4 people. In those 31 frames, a total of 55 false alarms occurred.

7 Conclusion

We have developed a visual intelligence system for automatically processing HD video into a highly compressed[4] script that identifies the actors, objects, and actions in the original video. The script is fully generative and can be used to produce graphical renderings ("envisionments") of the action and/or natural language text descriptions. The system combines state of the art vision components for human pose estimation and object detection with extremely powerful backend graphical models that allow video content to be represented and reasoned about at a symbolic level.

The performance of the overall system is currently limited by the human detection and pose estimation step. We have avoided the use of background subtraction so that the method can be applied from a moving platform or in situations where there is significant background motion. However, the state-of-the-art in person detection from static frames does not appear to be sufficiently robust to achieve desired performance levels. To be sure, in some cases, the person detection and pose estimation work well yielding convincing envisionments and text descriptions that match the action in the scene. But in many cases, the human detection is not reliable enough, particularly for non-standing poses. Making human detection and pose estimation robust, is a much-needed improvement.

Another interesting direction is to automatically determine which adjectives most clearly separate and uniquely identify the individuals involved in an action. Is it more informative to say "the person in the blue jeans" or "the taller person"? From descriptions provided by human annotators, it appears that gender designation ("the man approached the woman") is very common. Adding a gender classification component could provide a more human-like character to the descriptions.

[4] Five orders of magnitude more compact than the original raw video.

8 Acknowledgments

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The authors thank James Donlon and Pietro Michelucci for organizing the Mind's Eye program and providing support, Deva Ramanan of UCI for making his pose estimation code available, Jeff Siskind of Purdue University for sharing the Felzenszwalb object detection results he generated over this dataset, and Allan Runkle of JPL for developing the *DisplayServer* tool used to show the envisionments.

9 References

[1] Ferrucci, D., Brown, E., Chu-Caroll, J., Fan, J., Gondek, D., Kalyanpur, A.A., Lally, A., Murdock, J.W., Nyberg, E., Prager, J., Schlaefer, N., Welty, C., "Building Watson: An Overview of the DeepQA Project", *Assoc. for Advancement of AI*, pp. 59-79, (2010).

[2] Wolfram, Stephen, "WolframAlpha: Computational Knowledge Engine", URL: http://www.wolfram.com/

[3] *Siri*, URL: http://www.apple.com/ios/siri/

[4] Barnard, K. and Forsyth, D., "Learning the Semantics of Words and Pictures", *Int. Conf. on Computer Vision (ICCV)*, vol. 2, pp. 408-415, (2001).

[5] Ackerman, S., "Beyond Surveillance: DARPA wants a thinking camera", *Wired*, (2011/01/05).

[6] Aggarwal, J.K., Ryoo, M.S., "Human Activity Analysis: A Review", *ACM Computing Surveys*, 43(3), (2011/04/XX).

[7] Aggarwal, J.K., Ryoo, M.S., Kitani, K. "Tutorial on Human Activity Recognition – Frontiers of Human Activity Analysis", *IEEE Conf. on Computer Vision and Pattern Recognition, (CVPR)*, (2011).

[8] Hongeng, S., R. Nevatia, F. Bremond, "Video-based event recognition: activity representation and probabilistic recognition methods", *Computer Vision and Image Understanding (CVIU)*, 96, pp. 129-162, (2004).

[9] Van den Broek, S., ten Hove, J.-M., den Hollander, R., Burghouts, G., "Automated recognition of human activities in video streams in real time", *SPIE Newsroom*, (2014).

[10] Das, P., Xu, C., Doell, R.F., and Corso, J.J, "A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching." *CVPR*, (2013).

[11] S. O'Hara and B. Draper, "Using a Product Manifold Distance for Unsupervised Action Recognition", Image and Vision Computing, 30(3):206-216, (2012).

[12] Siddharth, N., Barbu, A., and Siskind, J.M., "Seeing What You're Told: Sentence-Guided Activity Recognition In Video", *CVPR*, pp. 732-739, Columbus, OH, (2014/06/XX).

[13] Guadarrama, S., Krishnmoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R., Darrell, T., Saenko, K., "YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-Shot Recognition", *ICCV*, (2013).

[14] Allen, J.F., "Maintaining Knowledge about Temporal Intervals", *Comm. of the ACM*, pp. 3-843, (1983/11/26).

[15] Brand, M., Oliver, N., Pentland, A., "Coupled Hidden Markov Models for Complex Action Recogntion", *CVPR*, pp. 994-999, (1997).

[16] Oliver, N., Horvitz, E., Ashutosh, G., "Layered Representations for Human Activity Recognition", *Fourth IEEE Int. Conf. on Multimodal Interfaces*, (2002).

[17] Rabiner, L.R., "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *Proc. of IEEE*, vol. 77, no. 2, (1989/02/XX).

[18] IP Video Market, URL: http://ipvideomarket.info/comapnies/videoanalytics/

[19] Brickstream, Inc., URL: http://www.brickstream.com/

[20] Smeaton, A.F., Over, P. , Kraaij, W., "Evaluation Campaigns and TRECVid", *Eigth ACM Int. Workshop on Multimedia Information Retrieval*, Santa Barbara, CA, pp. 321—330, (2006).

[21] Yang, Y., and Ramanan, D., "Articulated pose estimation with flexible mixtures-of-parts", *CVPR, (2011).*

[22] Felzenszwalb, P., Girshick, D. McAllester, D. Ramanan, "Object Detection with Discriminatively Trained Part Based Models", *IEEE Trans. on PAMI (TPAMI)*, vol. 32, no. 9, (2010/09/XX).

[23] Taylor, C., "Reconstruction of articulated objects from point correspondences in a single uncalibrated image", *CVIU*, 81(3):269-284, (2001/03/XX).

[24] Ramakrishna, V., Kanade, T., Sheikh, Y., "Reconstructing 3D Human Pose from 2D Image Landmarks", *Euro. Conf. on Comp. Vision (ECCV)*, pp. 573-586, (2012).

[25] Cox, I.J., and Hingorani, S.L., "An Efficient Implementation of Reid's Multiple Hypothesis Tracking Algorithm for the Purpose of Visual Tracking", *TPAMI*, vol. 18, no. 2, pp. 138-150, (1996/02/XX).

[26] Blackman, S.S., "Multiple Hypothesis Tracking for Multiple Target Tracking", *IEEE Trans. on Aerospace and Electronics Systems (AES)*, vol. 19, no. 1, pp. 5-18, (2004/01/XX).

[27] Visint.org, "Resources for visual intelligence research", URL: http://www.visint.org/index.html

[28] Bouguet J.-Y., "Camera Calibration Toolbox for Matlab", URL: http://www.vision.caltech.edu/bouguetj/calib_doc/

[29] Zhang, Z., "Flexible Camera Calibration by Viewing a Plane from Unknown Orientation", *ICCV*, (1999).

[30] Siskind, J., Purdue University, private communication via e-mail. (2011/07/19)

Applications of Mixed Pairwise Comparisons

Abeer Mirdad and Ryszard Janicki

Department of Computing and Software, McMaster University
Hamilton, Ontario, Canada, L8S 4K1
Email: {mirdadar, janicki}@mcmaster.ca

***Abstract*—Mixed pairwise comparisons is a systematic composition of quantitative and qualitative pairwise comparisons proposed in [9], [11] based on the use of non-linear scale proposed in [10]. We will show how this method can improve accuracy of weights assignment for attributes used for assessment of Healthcare in Canada, Quality In Use of Software, Smart Energy Grids and Medical Devices Managements Systems.**

***Keywords*-weights assignment, pairwise comparisons, consistency, quality assessment, subjective judgments**

I. Introduction

Pairwise comparisons are a very popular method of assigning weights when classification is mainly based on subjective judgments [19].

The *pairwise comparisons* method is based on the observation that it is much easier to rank the importance of two objects than it is to rank the importance of *several* objects. This very old idea goes back to Ramon Llull in the end of XIII century. Its modern *quantitative* version is due to papers by Marquis de Condorcet (1785), Fechner (1860), Thorstone (1927) and Saaty (1977). The last two papers provided mathematical foundations that are used today (see [10], [13], [19] for detailed references).

Quantitative pairwise comparisons use numbers to describe the preference relationship between objects, while *qualitative* pairwise comparisons use abstract relations (as 'slightly in favour' etc.) for description of this relationship. The model used in this paper originates from [8], [12].

In reality, most initial subjective judgments are always qualitative and transition from qualitative to quantitative is the weakest point of this technique [10], which is usually based on experience and domain knowledge of judges. A systematic composition of qualitative and quantitative pairwise comparisons was proposed in [9], [11]. This approach is based on the results of [10] where both approaches were compared and their mutual relationship was analyzed. In this paper we apply the technique of [9], [11] for finding weights assignments used in quality of Healthcare in Canada, 'in use' quality of software, attributes of smart energy grid and quality system in medical devices.

II. Quantitative Pairwise Comparisons

Let $E_1,...,E_n$ be a finite set of *entities* (alternatives, attributes, etc.) to be judged and/or ranked. The relationship between entities E_i and E_j is represented by a positive number a_{ij}. We assume $a_{ij} > 0$ and $a_{ij} = \frac{1}{a_{ij}}$, for $i, j = 1,...,n$ (which implies $a_{ii} = 1$ for all i).

If $a_{ij} > 1$ then E_i is more preferred than E_j and a_{ij} is a measure of this relationship (the bigger a_{ij} the bigger the difference), if $a_{ij} = 1$ then E_i and E_j are indifferent. Intuitively a_{ij} is interpreted as E_j is a_{ij} *times preferred (more important, etc.) than* E_j.

The matrix of such relative comparison coefficients:

$$A = [a_{ij}]_{n\times n}$$

is called a *pairwise comparison matrix*.

Since the entities E_1,...,E_n are not random, on contrary, they are usually carefully chosen and interrelated, the values of a_{ij} are not random, they should be somehow *consistent*.

A pairwise comparison matrix $A = [a_{ij}]_{n\times n}$ is *consistent* [18] if and only if

$$a_{ij}a_{jk} = a_{ik} \tag{1}$$

for $i, j, k = 1, ..., n$. It is known ([18]) that a pairwise comparison matrix A is consistent if and only if there exists positive numbers $w_1, ..., w_n$ such that $a_{ij} = \frac{w_i}{w_j}, i, j = 1, ..., n$. The values w_i are unique up to a multiplicative constant. They are often called *weights* and interpreted as a measure of importance. The summation of w_i often scaled to $w_1 + ... + w_n = 1$ or (100%). For a consistent pairwise comparison matrix A, the values w_i create a ranking (i.e. a weak order):

$E_i < E_j \iff w_i < w_j$ and $E_i \approx E_j \iff w_i = w_j$.

In practice, the values of a_{ij} rarely consistent, so some measurements of inconsistency and the ways to lower it, are needed.

In this paper we will use *distance based consistency index* [14] which is defined as follows:

$$cm_{\mathbf{A}} = \max_{(i,j,k)} \left(\min \left(\left| 1 - \frac{a_{ij}}{a_{ik}a_{kj}} \right|, \left| 1 - \frac{a_{ik}a_{kj}}{a_{ij}} \right| \right) \right) \tag{2}$$

In this case the most inconsistent triad, a_{ij}, a_{ik}, a_{kj}, is localized, which helps a lot in the process of inconsistency reduction. Other indexes do not have this useful property [2], [18].

Acceptable levels of inconsistency depend on particular interpretation of E_i, however for the index cm_A given by formula 1, the value 0.3 has some justification [2], [14].

Lowering a distance based consistency index cm_A is rather straightforward. Since the biggest 'troublemakers' are localized, we can improve consistency step by step, by small changes of values of the triple that results in the maximal inconsistency index. It was proved in [13] that this process converges.

TABLE I: Relationship between *additive*, *multiplicative* and *relational* scales. Formulas $a_{ij} = \frac{b_{ij}}{1-b_{ji}}$ and $b_{ij} = \frac{a_{ij}}{a_{ij}+1}$ are used to calculate the relationships between Columns I and III and Columns II and V. If no other data is available, default values are recommended.

quantitative scales [2], [10], [14]						**qualitative (relational) scale [8], [10]**	
additive scale of [10]		**multiplicative scales**				relation symbols for R_{ij}	**definition of intensity or importance** (E_i **vs** E_j)
range of b_{ij}		**from [10]** range of a_{ij}		a_{ij} defaults derived from Column II	**from [2], [14]**		
range	defaults	range	defaults		defaults		
0.44-0.55	0.5	0.79-1.27	1.0	1.0	1	$E_i \approx E_j$	*indifferent/equal/unknown*
0.56-0.65	0.6	1.28-1.94	1.6	1.5	2	$E_i \sqsupset E_j$	*slightly in favour/weak importance*
0.66-0.75	0.7	1.95-3.17	2.6	2.3	3	$E_i \supset E_j$	*in favour/moderate importance*
0.76-0.85	0.8	3.18-6.14	4.7	4.0	4	$E_i > E_j$	*strongly better/demonstrated imp.*
0.86-1.00	0.9	6.15-	7.0	9.0	5	$E_i \succ E_j$	*extremely better/absolute imp.*
Column I	Col. II	Col. III	Col. IV	Column V	Col. VI	Col. VII	Column VIII

Since usually we are not interested in lowering inconsistency index to zero, but rather to same acceptable level, we need a method for deriving a suitable value w_i from an inconsistent, but with acceptable level of inconsistency, matrix A. In this paper we will use a method proposed in [1] and calculate weights as the geometric means of columns (or equivalently, rows) of the matrix A, i.e. for $i = 1,...,n$,

$$w_i = \sqrt[n]{\prod_{j=1}^{n} a_{ij}} \tag{3}$$

When applying pairwise comparisons to various problems (see for example [11]) we have noticed that experts often felt much more comfortable and more confident when they were asked to divide 100 quality points between entities E_i and E_j than to provide multiplicative relationship, i.e. ratio a_{ij} .

Dividing of 100 between E_i and E_j means that we are replacing the multiplicative relationship $a_{ij}a_{ji} = 1$, with the additive relationship $b_{ij} + b_{ji} = 1$. Since the analysis of inconsistency requires $a_{ij}a_{ji} = 1$ relationship, we have to translate b_{ij} into appropriate a_{ij}.

The following simple transformation was proposed and analyzed in [9], for all $i, j = 1, \dots, n$:

$$a_{ij} = \frac{b_{ij}}{1-b_{ji}} = \frac{b_{ij}}{b_{ji}} \tag{4}$$

We may now analyze and reduce inconsistency by using the formula from equation 2.

III. Qualitative Pairwise Comparisons

In qualitative model [8], [12], numerical values a_{ij} or b_{ij} are replaced by the binary relations $\approx, \sqsupset, \supset, >, \succ$ and their inverses $\sqsubset, \subset, <, \prec$. The relations are interpreted as:

- $a \approx b$: a and b are *indifferent*,
- $a \sqsubset b$: *slightly in favor* of b,
- $a \subset b$: *in favor* of b,
- $a < b$: b is *strongly better*,
- $a \prec b$: b is *extremely better*.

The number of relations has been limited to five because of the known restrictions of human mind when it comes to subjective judgments [3], [15].

The relations $\approx, \sqsupset, \supset, >, \succ, \sqsubset, \subset, <, \prec$ are disjoint and cover the all cases. The relation $\approx$ is symmetric and includes identity.

The set of relations $RS = \{\approx, \sqsubset, \subset, <, \prec\}$ is called a *ranking system* if the relations $\prec, \prec, \cup <, \prec \cup < \cup \subset$ and $\prec \cup < \cup \subset \cup \sqsubset$ are *partial orders*, i.e. irreflexive and transitive relations [5]. Quite often it is also required the relation $\approx$ to be an equivalence relation (c.f. [8], [12]).

The consistency of RS is defined by a set of axioms it must satisfy. Intuitively consistency means that the relationships E_i vs E_j and E_j vs E_k influence the relationship E_i vs E_k (as $a_{ij}a_{jk} = a_{ik}$ for quantitative case). The number of axioms is substantial [12] as all combinations of all relational compositions must be taken care of, however the idea on which all those axioms are constructed is very simple, namely: *"composition of relations should be relatively continuous and must not change preferences in a drastic way"*.

For example the Axiom 2.1 of [12] looks as follows:

$$(a \approx b \wedge b \sqsubset c) \vee (a \sqsubset b \vee b \approx c) \implies (a \approx c \vee a \sqsubset c \vee a \subset c).$$

There are two algorithms that start with arbitrary ranking data and provide consistent ranking systems RS.

IV. Mixed Model

In reality each subjective judgment starts with qualitative assessment and quantitative estimation of the assessment is often the most influential part of the entire process. Linear scales (wrt. a_{ij}) transforming qualitative assessments into numerical values have been proposed and discussed in [2], [14], [18], [19]. They all have only intuitive, heuristic and experimental justifications. The scale from [2], [14] is presented in Column VI of Table I. It is a scaled down initial ten points scale of [18]. According to [3], [15], the length of the scale should be between four [3] and seven alternatives [15].

In [10] a new *non-linear* (wrt. a_{ij}) five point scale has been proposed. It was derived from comparing quantitative consistency (i.e. $a_{ij}a_{jk} = a_{ik}$) with qualitative consistency given by axioms in [12], and the assumption that:

> "*transformation in either way should preserve appropriate consistencies*".

The linear (wrt. b_{ij}) and non-linear (wrt. a_{ij}) scales from [10] are presented in Columns I–V of Table I.

Composing the results of [2], [14], [8], [10], [12] resulted in *Mixed Model*, first suggested in [11] and formally proposed in [9], which is explained by the following procedure.

Procedure 1 (Mixed Model)**.**

1) Experts provide *qualitative* judgments using relations from Columns VII and VIII of Table I.
2) Experts transform their qualitative judgments into *quantitative* judgments either using *additive* scale (Columns I and II), or *multiplicative* scale (Columns III, IV and V) from Table I, or *both* scales. It is recommended to use the additive scale for the relations $\approx$ and $\sqsupset$. Sometimes the multiplicative scale (from Columns III and IV) works better for the relation $\succ$.
3) If the additive scale has been used, the values b_{ij} are transformed into a_{ij} using the equation (4).
4) A standard procedure for *distance based* inconsistency reduction is used (equations (2)).
5) The outcome, which is a pairwise comparison matrix with acceptable inconsistency, is transformed back into *qualitative* matrix using ranges from Column III of Table I, i.e. a_{ij}'s are transformed into appropriate R_{ik} from Column VII.
6) In some cases the outcome is also transformed into additive pairwise comparison matrix $[b_{ij}]_{n\times n]}$ by using the formula $b_{ij} = \frac{a_{ij}}{a_{ij}+1}$.
7) All final qualitative and quantitative tables are sent back to experts for final adjustments and potential changes.
8) The whole process is repeated as many times as necessary.
9) After the results are accepted, the weights are calculated as the geometric means of columns of the final matrix $A = [a_{ij}]_{n\times n}$, i.e. $w_i = \sqrt[n]{\prod_{j=1}^{n} a_{ij}}$, for $i = 1, \dots, n$.
10) If ranking is required, it is derived from the weights in a standard manner. $\square$

A version of the above procedure will be used in all applications that follow.

V. Weights Assignment Process

Procedure 1 is fairly general, it many different concrete instantiations. The following version of Procedure 1 was used in all four applications discussed in this paper.

Procedure 2.

1) A small group of experts were given a number of attributes and they were asked to provide *qualitative* values using relations from Table I.
2) Then the same experts were asked to provide *quantitative* evaluation using the *additive scale* and ranges that provided in Column I of Table I. *In all cases the experts decided to use used default values from Column II.*
3) Next the pairwise comparisons matrix $[a_{ij}]_{n\times n}$ was produced by using the formula $a_{ij} = \frac{b_{ij}}{1-b_{ij}}$. The distance based consistency index cm_A (equations (2)) is calculated, and, if necessary, reduced to the level smaller than 0.3. *JConcluder* software [23] was used to reduce inconsistency to an acceptable level.
4) After few days the same experts were asked to transform their qualitative judgments into quantitative form, but now by using *multiplicative* scale and ranges from Columns III and IV of from Table I. *Again in all cases the experts decided to use used default values, in this case it was Column IV*. *JConcluder* [23] was again used to reduce inconsistency to an acceptable level. *We observed that it is easier for experts to differentiate between different, say, indifferences, when b_{ij} were used; than when a_{ij} were used.*
5) The matrices from steps (3) and (4) were then translated back into qualitative form using Column II of Table I. Even though the quantitative matrices were slightly different, their qualitative representations were identical (*which can be interpreted as yet another validation of the distance based consistency*).
6) For all four applications the final result was different than the initial one. The final results were send back to the experts for final analysis and acceptance. $\square$

VI. Quality of Healthcare in Canada

The quality of Healthcare in Canada has been analyzed and assessed in *six* key domains [21], [22]: the *effectiveness of* the healthcare sector in improving health outcomes; *access* to healthcare services; the *capacity* of systems to deliver appropriate services; the *safety* of care delivered; the degree to which healthcare in Canada is *patient-centred*; and *equity* in healthcare outcomes and delivery.

However the importance of particular domains/attributes is far from obvious and open for a discussion.

Using pairwise comparisons based techniques can provide ranking and weights assignments that are more trustworthy than these derived from informal or other semi-formal derivations. Procedure 2 has been used by a group of experts and the results of using it are presented in Tables II–VIII. Table VIII contains the final qualitative results and its bottom row contains final calculated weights. The weights in Table VIII are just averages of the appropriate weights from Tables V and VII. Note that the difference of appropriate weights from Table V and Table VII is rather small, which could also be seen as validation of our method. The experts have approved our final results, both qualitative and quantitative.

The difference between initial and final judgments were not severe, but the initial judgments were not consistent, while the final judgments were consistent (i.e. with acceptable level of inconsistency), hence they were much more trustworthy.

Our finding could help to develop initiatives to address specific quality problems in Canada's Healthcare, and ultimately lead to better outcomes for patients.

The weights obtained by our analysis (see Table VIII) have the following values (scaled to 100%):

TABLE II: **Healthcare in Canada**. Initial *qualitative* judgments.

Relations	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	$\approx$	$\supset$	$>$	$\approx$	$\approx$	$\supset$
Access	E_2	$\subset$	$\approx$	$\supset$	$\subset$	$<$	$\supset$
Capacity	E_3	$<$	$\subset$	$\approx$	$<$	$<$	$\approx$
Safety	E_4	$\approx$	$\supset$	$>$	$\approx$	$\approx$	$>$
Patient-Centredness	E_5	$\approx$	$>$	$>$	$\approx$	$\approx$	$>$
Equity	E_6	$\subset$	$\subset$	$\approx$	$<$	$<$	$\approx$

TABLE III: **Healthcare in Canada**. Initial *quantitative* judgments derived from Table II by using defaults values from Column II of Table I.

b_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	0.5	0.7	0.8	0.5	0.5	0.7
Access	E_2	0.3	0.5	0.7	0.3	0.2	0.7
Capacity	E_3	0.2	0.3	0.5	0.2	0.2	0.5
Safety	E_4	0.5	0.7	0.8	0.5	0.5	0.8
Patient-Centredness	E_5	0.5	0.8	0.8	0.5	0.5	0.8
Equity	E_6	0.3	0.3	0.5	0.2	0.2	0.5

TABLE IV: **Healthcare in Canada**. The values of a_{ij} obtained from Table III.

$a_{ij} = \frac{b_{ij}}{1-b_{ij}}$	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	1.0	2.3	4.0	1.0	1.0	2.3
Access	E_2	0.43	1.0	2.3	0.43	0.25	2.3
Capacity	E_3	0.25	0.43	1.0	0.25	0.25	1.0
Safety	E_4	1.0	2.3	4.0	1.0	1.0	4.0
Patient-Centredness	E_5	1.0	4.0	4.0	1.0	1.0	4.0
Equity	E_6	0.43	0.43	1.0	0.25	0.25	1.0
inconsistency coefficient $cm_A = 0.57 > 0.3$.							

TABLE V: **Healthcare in Canada**. Consistent (i.e. with acceptable inconsistency) matrix derived from Table IV, using distance-based consistency. The weights were calculated using the geometric means.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	1.0	2.3	5.2	1.3	1.1	4.6
Access	E_2	0.43	1.0	2.3	0.5	0.5	2.0
Capacity	E_3	0.2	0.43	1.0	0.25	0.25	1.0
Safety	E_4	0.77	1.8	4.0	1.0	1.0	4.0
Patient-Centredness	E_5	0.9	2.0	4.0	1.0	1.0	4.0
Equity	E_6	0.22	0.5	1.0	0.25	0.25	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5	w_6
values		**29%**	**12%**	**6%**	**23%**	**24%**	**6%**
inconsistency coefficient $cm_A = 0.15 < 0.3$.							

TABLE VI: **Healthcare in Canada**. Initial *quantitative* judgments derived from Table II by using defaults values from Column IV of Table I.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	1.0	2.6	4.7	1.0	1.0	2.6
Access	E_2	0.38	1.0	2.6	0.38	0.21	2.6
Capacity	E_3	0.21	0.38	1.0	0.21	0.21	1.0
Safety	E_4	1.0	2.6	4.7	1.0	1.0	4.7
Patient-Centredness	E_5	1.0	4.7	4.7	1.0	1.0	4.7
Equity	E_6	0.38	0.38	1.0	0.21	0.21	1.0
inconsistency coefficient $cm_A = 0.62 > 0.3$.							

Effectiveness = 30%, Access = 12%, Capacity = 5%, Safety = 24%, Patient-Centredness = 24%, Equity = 5% .

TABLE VII: **Healthcare in Canada**. Consistent pairwise comparisons matrix derived from Table VI. This table is almost identical as Table V. The different cells are shaded.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	1.0	2.6	5.7	1.3	1.1	5.7
Access	E_2	0.38	1.0	2.6	0.5	0.5	2.3
Capacity	E_3	0.17	0.38	1	0.21	0.21	1.0
Safety	E_4	0.77	2	4.7	1.0	1.0	4.7
Patient-Centredness	E_5	0.9	2.0	4.7	1.0	1.0	4.7
Equity	E_6	0.17	0.43	1.0	0.21	0.21	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5	w_6
values		**30%**	**12%**	**5%**	**24%**	**24%**	**5%**
inconsistency coefficient $cm_A = 0.15 < 0.3$.							

TABLE VIII: **Healthcare in Canada**. Final *qualitative* judgments of key attributes derived from Tables V and VII by using the intervals from Column III of Table I. Corrected cells are shaded. Also final weights.

Relations	Name	E_1	E_2	E_3	E_4	E_5	E_6
Effectiveness	E_1	$\approx$	$\supset$	$>$	$\sqsupset$	$\approx$	$>$
Access	E_2	$\subset$	$\approx$	$\supset$	$\subset$	$\subset$	$\supset$
Capacity	E_3	$<$	$\subset$	$\approx$	$<$	$<$	$\approx$
Safety	E_4	$\sqsubset$	$\supset$	$>$	$\approx$	$\approx$	$>$
Patient-Centr.	E_5	$\approx$	$\supset$	$>$	$\approx$	$\approx$	$>$
Equity	E_6	$<$	$\subset$	$\approx$	$<$	$<$	$\approx$
Final weights		**29.5%**	**12%**	**5.5%**	**23.5%**	**24%**	**5.5%**

VII. Software Quality in Use

ISO (c.f. [7], [16]) has recently developed a new more comprehensive definition of quality in use, which has usability, flexibility and safety as subcharacteristics that can be quantified from the perspectives of different stakeholders, including users, managers and maintainers. Quality in use depends not only on the software or computer system, but also on the particular context in which the product is being used and it can be assessed by observing representative users carrying out representative tasks in a realistic context of use [17].

The standard ISO/IEC 25010 [7] proposes the following attributes (called 'characteristics') for the assessment of software product quality in use: *effectiveness, efficiency, satisfaction, freedom from risk* and context coverage with which users can achieve goals in a specified context of use.

To provide trustworthy assignment importance indicators, Procedure 2 has been used by a group of software experts. The results are presented in Tables IX–XV. Notes that in this case the weights from Table XII and Table XIV are identical.

The difference of appropriate weights from Table XII and Table XIV is rather small and the experts have approved our final results, both qualitative and quantitative.

As in the previous case, the difference between initial and final judgments were not severe, but the initial judgments were not consistent, while the final judgments were consistent (i.e. with acceptable level of inconsistency), hence they were much more trustworthy.

The weights obtained by our analysis (see Table XV) have the following values (scaled to 100%):
Effectiveness =21%, Efficiency = 12%, Satisfaction = 26% , Freedom From Risk = 25%, Context Coverage = 16%.

TABLE IX: **Software Quality In Use**. Initial *qualitative* judgments

Relations	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	$\approx$	$\sqsupset$	$\sqsupset$	$\sqsubset$	$\sqsupset$
Efficiency	E_2	$\sqsubset$	$\approx$	$\subset$	$\sqsubset$	$\sqsubset$
Satisfaction	E_3	$\sqsubset$	$\supset$	$\approx$	$\supset$	$\supset$
Freedom From Risk	E_4	$\sqsupset$	$\sqsupset$	$\subset$	$\approx$	$\supset$
Context Coverage	E_5	$\sqsubset$	$\sqsupset$	$\subset$	$\subset$	$\approx$

TABLE X: **Software Quality In Use**. Initial *quantitative* judgments derived from Table IX by using defaults values from Column II of Table I.

b_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	0.5	0.6	0.6	0.4	0.6
Efficiency	E_2	0.4	0.5	0.3	0.4	0.4
Satisfaction	E_3	0.4	0.7	0.5	0.7	0.7
Freedom From Risk	E_4	0.6	0.6	0.3	0.5	0.7
Context Coverage	E_5	0.4	0.6	0.3	0.3	0.5

TABLE XI: **Software Quality In Use**. The values of a_{ij} obtained from Table X.

$a_{ij} = \frac{b_{ij}}{1-b_{ij}}$	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	1.0	1.5	1.5	0.67	1.5
Efficiency	E_2	0.67	1.0	0.43	0.67	0.64
Satisfaction	E_3	0.67	2.3	1.0	2.3	2.3
Freedom From Risk	E_4	1.5	1.5	0.43	1.0	2.3
Context Coverage	E_5	0.67	1.5	0.43	0.43	1.0
inconsistency coefficient $cm_A = 0.81 > 0.3$.						

TABLE XII: **Software Quality In Use**. Consistent pairwise comparisons matrix derived from Table XI.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	1.0	1.8	0.80	0.80	1.4
Efficiency	E_2	0.57	1.0	0.43	0.5	0.8
Satisfaction	E_3	1.15	2.3	1.0	1.03	1.64
Freedom From Risk	E_4	1.0	1.8	0.97	1.0	1.64
Context Coverage	E_5	0.67	1.14	0.43	0.6	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5
values		**21%**	**12%**	**26%**	**25%**	**16%**
inconsistency coefficient $cm_A = 0.12 < 0.3$.						

TABLE XIII: **Software Quality In Use**. Initial *quantitative* judgments derived from Table IX by using defaults values from Column IV of Table I.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	1.0	1.6	1.6	0.63	1.6
Efficiency	E_2	0.63	1.0	0.38	0.63	0.63
Satisfaction	E_3	0.63	2.6	1.0	2.6	2.6
Freedom From Risk	E_4	1.6	1.6	0.38	1.0	2.6
Context Coverage	E_5	0.63	1.6	0.38	0.38	1.0
inconsistency coefficient $cm_A = 0.85 > 0.3$.						

VIII. Smart Grid

A smart grid is a modernized electrical grid that uses analog or digital information and communications technology to gather and act on information - such as information about the behaviours of suppliers and consumers - in an automated

TABLE XIV: **Software Quality In Use**. Consistent pairwise comparisons matrix derived from Table XIII. This table is almost identical as Table XII. The different cells are shaded.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	1.0	1.9	0.80	0.9	1.4
Efficiency	E_2	0.52	1.0	0.43	0.5	0.8
Satisfaction	E_3	1.15	2.3	1.0	1.03	1.64
Freedom From Risk	E_4	1.1	1.8	0.97	1.0	1.64
Context Coverage	E_5	0.67	1.14	0.43	0.6	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5
values		**21%**	**12%**	**26%**	**25%**	**16%**
inconsistency coefficient $cm_A = 0.12 < 0.3$.						

TABLE XV: **Software Quality In Use**. Final *qualitative* judgments of key attributes derived from Tables XII and XIV by using the intervals from Column III of Table I. Corrected cells are shaded. Also final weights.

Relations	Name	E_1	E_2	E_3	E_4	E_5
Effectiveness	E_1	$\approx$	$\sqsupset$	$\approx$	$\approx$	$\sqsupset$
Efficiency	E_2	$\sqsupset$	$\approx$	$\subset$	$\supset$	$\approx$
Satisfaction	E_3	$\approx$	$\supset$	$\approx$	$\approx$	$\sqsupset$
Freedom From Risk	E_4	$\approx$	$\subset$	$\approx$	$\approx$	$\sqsupset$
Context Coverage	E_5	$\sqsubset$	$\approx$	$\sqsubset$	$\sqsubset$	$\approx$
Final weights		**21%**	**12%**	**26%**	**25%**	**16%**

fashion to improve the efficiency, reliability, economics, and sustainability of the production and distribution of electricity [20]. The recommended assessment attributes are usually: *reliability, flexibility, efficiency, sustainability* and *market-enabling*.

The results of using Procedure 2 for these attributes are presented in Tables XVI–XXII. In this case the weights from Tables XIX and XXI are slightly different and the final weights presented in Table XXII are just averages of these from Tables XIX and XXI. However, qualitative tables Tables XIX and XXI are identical and equal to Table XXII. As in the previous two cases, the experts approved final results of our analysis.

The scaled weights have the following values:
Reliability = 18%, Flexibility = 8%, Efficiency = 31.5%, Sustainability = 8%, Market-enabling = 31.5%.

TABLE XVI: **Smart Grid** Initial *qualitative* judgments.

Relations	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	$\approx$	$\supset$	$\sqsupset$	$\sqsupset$	$\approx$
Flexibility	E_2	$\subset$	$\approx$	$<$	$\sqsubset$	$<$
Efficiency	E_3	$\sqsubset$	$>$	$\approx$	$>$	$\supset$
Sustainability	E_4	$\sqsubset$	$\sqsupset$	$<$	$\approx$	$<$
Market-enabling	E_5	$\approx$	$>$	$\sqsubset$	$>$	$\approx$

TABLE XVII: **Smart Grid**. Initial *quantitative* judgments derived from Table XVI by using default values from Column II of Table I.

b_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	0.5	0.7	0.6	0.6	0.5
Flexibility	E_2	0.3	0.5	0.2	0.4	0.2
Efficiency	E_3	0.4	0.8	0.5	0.8	0.7
Sustainability	E_4	0.4	0.6	0.2	0.5	0.2
Market-enabling	E_5	0.5	0.8	0.4	0.8	0.5

TABLE XVIII: **Smart Grid**. The values of a_{ij} obtained from Table XVII.

$a_{ij} = \frac{b_{ij}}{1-b_{ij}}$	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	1.0	2.3	1.5	1.5	1.0
Flexibility	E_2	0.43	1.0	0.25	0.67	0.25
Efficiency	E_3	0.67	4.0	1.0	4.0	2.3
Sustainability	E_4	0.67	1.5	0.25	1.0	0.25
Market-enabling	E_5	1.0	4.0	0.67	4.0	1.0
inconsistency coefficient $cm_A = 0.75 > 0.3$.						

TABLE XIX: **Smart Grid**. Consistent pairwise comparisons matrix derived from Table XVIII.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	1.0	2.3	0.57	2.3	0.57
Flexibility	E_2	0.43	1.0	0.25	1.0	0.22
Effecincy	E_3	1.7	4.0	1.0	4.0	1.0
Sustainability	E_4	0.43	1.0	0.25	1.0	0.25
Market-enabling	E_5	1.7	4.5	1.0	4.0	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5
values		**19%**	**8%**	**32%**	**8%**	**33%**
inconsistency coefficient $cm_A = 0.12 < 0.3$.						

TABLE XX: **Smart Grid**. Initial *quantitative* judgments derived from Table XVI by using the default values from Column IV of Table I.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	1.0	2.6	1.6	1.6	1.0
Flexibility	E_2	0.38	1.0	0.21	0.63	0.21
Efficiency	E_3	0.63	4.7	1.0	4.7	2.6
Sustainability	E_4	0.63	1.6	0.21	1.0	0.21
Market-enabling	E_5	1.0	4.7	0.63	4.7	1.0
inconsistency coefficient $cm_A = 0.79 > 0.3$.						

TABLE XXI: **Smart Grid**. Consistent pairwise comparisons matrix derived from Table XX. This table is almost identical as Table XIX. The different cells are shaded.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	1.0	2.3	0.57	2.3	0.54
Flexibility	E_2	0.43	1.0	0.25	1.0	0.22
Efficiency	E_3	1.7	4.0	1.0	4.0	0.9
Sustainability	E_4	0.43	1.0	0.25	1.0	0.25
Market-enabling	E_5	1.8	4.5	1.1	4.0	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5
values		**18%**	**8%**	**31%**	**8%**	**30%**
inconsistency coefficient $cm_A = 0.12 < 0.3$.						

TABLE XXII: **Smart Grid**. Final *qualitative* judgments of key attributes derived from Tables XIX and XXI by using the intervals from Column III of Table I. Corrected cells are shaded. Also final weights in bottom row.

Relations	Name	E_1	E_2	E_3	E_4	E_5
Reliability	E_1	$\approx$	$\supset$	$\sqsubset$	$\supset$	$\sqsubset$
Flexibility	E_2	$\subset$	$\approx$	$<$	$\approx$	$<$
Efficiency	E_3	$\approx$	$>$	$\approx$	$>$	$\approx$
Sustainability	E_4	$\subset$	$\approx$	$<$	$\approx$	$<$
Market-enabling	E_5	$\sqsupset$	$>$	$\approx$	$>$	$\approx$
Final weights		**18%**	**8%**	**31.5%**	**8%**	**31.5%**

IX. MEDICAL DEVICES

The ISO 13485 [4], [6] standard proposes the following attributes (called 'characteristics') for the assessment of Medical Devices Quality Managements Systems:

- *Developing an effective design*
- *Conducting risk analyses*
- *Following adequate and appropriate standard operating procedures and protocols for testing*
- *Using validated methods and procedures*
- *Monitoring and auditing*
- *Ensuring adequate staff training*
- *Adopting and implementing appropriate corrective and preventive action plans*

We used Procedure 2 to provide appropriate weights for each of the above attributes. The results are presented in Tables XXIII–XXIX. While the Tables XXVI and XXVIII are slightly different, the weights they generate are identical (as in the case of Software Quality In Use). Similarly as in the previous three cases, the qualitative tables Tables XXVI and XXVIII are identical and equal to Table XXIX, and the experts approved final results of our analysis. The final results are consistent, so they are more trustworthy than the initial ones, that were based only on the experts opinions.

The scaled weights have the following values in this case: *Effective Design = 7%, Conducting Risk Analyses = 5%, Following Standard = 12%, Using Validated Methods = 29%, Monitoring and Auditing= 15%, Staff Training = 22%, Adopting and Implementing = 10%.*

X. FINAL COMMENTS

A novel weights assignment techniques, proposed in [9], [11] and called Mixed Pairwise Comparisons, has been applied to improve accuracy of weights assignment for attributes used for assessment of Healthcare in Canada, Quality In Use of Software, Smart Energy Grids and Quality System in Medical

TABLE XXIII: **Medical Devices**. Initial *qualitative* judgments.

Relations	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	$\approx$	$\sqsupset$	$\subset$	$<$	$\subset$	$<$	$\sqsupset$
Conducting Risk Analyses	E_2	$\sqsubset$	$\approx$	$\subset$	$\prec$	$\subset$	$<$	$\sqsubset$
Following Standard	E_3	$\supset$	$\supset$	$\approx$	$\subset$	$\approx$	$\sqsubset$	$\sqsubseteq$
Using Validated Methods	E_4	$>$	$\succ$	$\supset$	$\approx$	$\sqsupset$	$\approx$	$>$
Monitoring and Auditing	E_5	$\supset$	$\supset$	$\approx$	$\sqsubset$	$\approx$	$\approx$	$\sqsupseteq$
Staff Training	E_6	$>$	$>$	$\sqsupset$	$\approx$	$\approx$	$\approx$	$\supset$
Adopting and Implem.	E_7	$\sqsubset$	$\sqsupset$	$\sqsupset$	$<$	$\sqsubset$	$\subset$	$\approx$

TABLE XXIV: **Medical Devices**. Initial *quantitative* judgments derived from Table XXIII by using the default values from Column II of Table I.

b_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	0.5	0.6	0.3	0.2	0.3	0.2	0.6
Conducting Risk Analyses	E_2	0.4	0.5	0.3	0.1	0.3	0.2	0.4
Following Standard	E_3	0.7	0.7	0.5	0.3	0.5	0.4	0.4
Using Validated Methods	E_4	0.8	0.9	0.7	0.5	0.6	0.5	0.8
Monitoring and Auditing	E_5	0.7	0.7	0.5	0.4	0.5	0.5	0.6
Staff Training	E_6	0.8	0.8	0.6	0.5	0.5	0.5	0.7
Adopting and Implem.	E_7	0.4	0.6	0.6	0.2	0.4	0.3	0.5

TABLE XXV: **Medical Devices**. The values of a_{ij} obtained from Table XXIII.

$a_{ij} = \frac{b_{ij}}{1-b_{ij}}$	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	1.0	1.5	0.43	0.25	0.43	0.25	1.5
Conducting Risk Analyses	E_2	0.67	1.0	0.43	0.11	0.43	0.25	0.67
Following Standard	E_3	2.3	2.3	1.0	0.43	1.0	0.67	0.67
Using Validated Methods	E_4	4.0	9.0	2.3	1.0	1.5	1.0	4.0
Monitoring and Auditing	E_5	2.3	2.3	1.0	0.67	1.0	1.0	1.5
Staff Training	E_6	4.0	4.0	1.5	1.0	1.0	1.0	2.3
Adopting and Implem.	E_7	0.67	1.5	1.5	0.25	0.67	0.43	1.0
inconsistency coefficient $cm_A = 0.81 > 0.3$.								

TABLE XXVI: **Medical Devices**. Consistent pairwise comparisons matrix derived from Table XXV.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	1.0	1.5	0.54	0.25	0.43	0.3	0.67
Conducting Risk Analyses	E_2	0.7	1.0	0.43	0.18	0.32	0.21	0.52
Following Standard	E_3	1.8	2.3	1.0	0.43	0.82	0.54	1.1
Using Validated Methods	E_4	4.0	5.5	2.3	1.0	1.85	1.16	3.29
Monitoring and Auditing	E_5	2.3	3.1	1.1	0.5	1.0	0.76	1.5
Staff Training	E_6	3.3	4.7	1.8	0.8	1.3	1.0	2.3
Adopting and Implem.	E_7	1.5	1.9	0.8	0.3	0.6	0.43	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5	w_6	w_7
values		**7%**	**5%**	**12%**	**29%**	**15%**	**22%**	**10%**
inconsistency coefficient $cm_A = 0.19 < 0.3$.								

TABLE XXVII: **Medical Devices**. Initial *quantitative* judgments derived from Table XXIII by using the default values from Column IV of Table I.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	1.0	1.6	0.38	0.21	0.38	0.21	1.6
Conducting Risk Analyses	E_2	0.63	1.0	0.38	0.14	0.38	0.21	0.63
Following Standard	E_3	2.6	2.6	1.0	0.38	1.0	0.63	0.63
Using Validated Methods	E_4	4.7	7.0	2.6	1.0	1.6	1.0	4.7
Monitoring and Auditing	E_5	2.6	2.6	1.0	0.63	1.0	1.0	1.6
Staff Training	E_6	4.7	4.7	1.6	1.0	1.0	1.0	2.6
Adopting and Implem.	E_7	0.63	1.6	1.6	0.21	0.63	0.38	1.0
inconsistency coefficient $cm_A = 0.99 > 0.3$.								

TABLE XXVIII: **Medical Devices**. Consistent pairwise comparisons matrix derived from Table XXVII. This table is almost identical as Table XXV. The different cells are shaded.

a_{ij}	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	1.0	1.15	0.54	0.25	0.43	0.3	0.80
Conducting Risk Analyses	E_2	0.8	1.0	0.43	0.18	0.32	0.21	0.52
Following Standard	E_3	1.8	2.3	1.0	0.43	0.82	0.54	1.1
Using Validated Methods	E_4	4.0	5.5	2.3	1.0	1.85	1.16	3.29
Monitoring and Auditing	E_5	2.3	3.1	1.1	0.5	1.0	0.76	1.5
Staff Training	E_6	3.3	4.7	1.8	0.8	1.3	1.0	2.3
Adopting and Implem.	E_7	1.15	1.9	0.8	0.3	0.6	0.43	1.0
weights of criteria		w_1	w_2	w_3	w_4	w_5	w_6	w_7
values		**7%**	**5%**	**12%**	**29%**	**15%**	**22%**	**10%**
inconsistency coefficient $cm_A = 0.19 < 0.3$.								

TABLE XXIX: **Medical Devices**. Final *qualitative* judgments of key attributes derived from Tables XXVI and XXVIII by using the intervals from Column III of Table I. Corrected cells are shaded. Final weights are also presented.

Relations	Name	E_1	E_2	E_3	E_4	E_5	E_6	E_7
Effective Design	E_1	$\approx$	$\approx$	$\sqsubset$	$<$	$\subset$	$<$	$\sqsupseteq$
Conducting Risk Analyses	E_2	$\approx$	$\approx$	$\subset$	$<$	$\subset$	$<$	$\sqsubseteq$
Following Standard	E_3	$\sqsupset$	$\supset$	$\approx$	$\subset$	$\approx$	$\sqsubset$	$\sqsubset$
Using Validated Methods	E_4	$>$	$>$	$\supset$	$\approx$	$\sqsupset$	$\approx$	$>$
Monitoring and Auditing	E_5	$\supset$	$\supset$	$\approx$	$\sqsubset$	$\approx$	$\sqsubset$	$\sqsupset$
Staff Training	E_6	$>$	$>$	$\sqsupset$	$\approx$	$\sqsupset$	$\approx$	$\supset$
Adopting and Implem.	E_7	$\sqsubset$	$\sqsupset$	$\sqsupset$	$<$	$\sqsubset$	$\subset$	$\approx$
Final weights		**7%**	**5%**	**12%**	**29%**	**15%**	**22%**	**10%**

Devices Managements System.

ACKNOWLEDGMENT

Abeer Mirdad acknowledges a support by the Ministry of Higher Education in Saudi Arabia through The Saudi Arabian Cultural Bureau in Canada, and Ryszard Janicki acknowledges a partial support by NSERC grant of Canada.

REFERENCES

[1] J. Barzilai. Deriving weights from pairwise comparison matrices. *Journal of the operational research society*, 48(12):1226–1232, 1997.

[2] S. Bozóki, T. Rapcsák. On Saaty's and Koczkodaj's inconsistencies of pairwise comparison matrices. *Journal of Global Optimization*, 42(2):157–175, 2008.

[3] N. Cowan. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioural and Brain Sci.*, 24:87–185, 2001.

[4] C.T. DeMarco, *Medical Device Design and Regulation*, ASQ Quality Press, 2011

[5] S. French. *Decision theory.* Halsted Press, 1986.

[6] ISO 13485:2003 Standard, *Medical devices. Quality management systems. Requirements for regulatory purposes*, http://www.iso.org/iso/catalogue_detail?csnumber=36786

[7] ISO/IEC 25022 Standard, *System and Software Quality Requirements and Evaluation*, http://www.iso.org/iso/catalogue_detail.htm?csnumber=35746

[8] R. Janicki. Pairwise comparisons based non-numerical ranking. *Fundamenta Informaticae*, 94(2):197–217, 2009.

[9] R. Janicki. On Qualitative and Quantitative Pairwise Comparisons, Proc. of ICAT'2015, to appear.

[10] R. Janicki, Y. Zhai. Remarks on pairwise comparison numerical and non-numerical rankings. *Lecture Notes in Computer Science* 6954, pp. 290–300. Springer, 2011.

[11] R. Janicki, M. H. Soudkhah, On Classification with Pairwise Comparisons, Support Vector Machines and Feature Domain Overlapping, *The Computer Journal* (2014), doi:10.1093/comjnl/bxu085

[12] R. Janicki, Y. Zhai. On a pairwise comparison-based consistent non-numerical ranking. *Logic Journal of IGPL*, 20(4):667–676, 2012.

[13] W. W. Koczkodaj, S.J. Szarek. On distance-based inconsistency reduction algorithms for pairwise comparisons. *Logic Journal of IGPL*, 18(6):859–869, 2010.

[14] W.W. Koczkodaj. A new definition of consistency of pairwise comparisons. *Mathematical and Computer Modelling*, 18(7):79–84, 1993.

[15] G. A. Miller. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2):81, 1956.

[16] B. Nigel. Extending quality in use to provide a framework for usability measurement *Lecture Notes in Computer Science* 5619, pp. 13–22, Springer, 2009

[17] J. Oh, D. Park, B. Lee, J. Lee, E. Hong, C. Wu. Certification of software packages using hierarchical classification *Lecture Notes in Computer Science* 3024, pp. 209–224, Springer, 2004

[18] T. L. Saaty. A scaling method for priorities in hierarchical structures. *Journal of mathematical psychology*, 15(3):234–281, 1977.

[19] T. L. Saaty. *Theory and applications of the analytic network process: decision making with benefits, opportunities, costs, and risks.* RWS publications, 2005.

[20] *Smart Grids and Renewables: A Guide for Effective Deployment*, International Renewable Energy Agency (IRENA), 2013.

[21] K. Sutherland, S. Leatherman, S. Law, J. Verma, S. Petersen. Chartbook: shining a light on the quality of healthcare in Canada: what can be learned to catalyze improvements in healthcare quality? *Healthc Pap.* 12(1):10-24, 2012.

[22] J.I. Williams, W. Young, A Summary of Studies on the Quality of Health Care Administrative Databases in Canada. In *Patterns of Health Care in Ontario: the ICES Practice Atlas*, 2^{nd} ed., pp. 339–345, Canadian Medical Association, 1996.

[23] D. Xu. Concluder, http://sourceforge.net/projects/concluder/?source=directory,April2013.

Artificial Psychology Revisited: Constructs for Modeling Artificial Emotions

James A. Crowder, John N. Carbone, Shelli Friess
Raytheon Intelligence, Information, and Services, Aurora, CO, USA
Raytheon Intelligence, Information, and Services, Garland, TX, USA
Psychology, Argosy University, Denver, CO, USA

Abstract - *Understanding the DNA of feelings and emotions are becoming increasingly important mechanisms for facilitating learning in intelligent systems. Here we present constructs for modeling human emotions within a foundation of artificial cognitive architectures[1] We model artificial neural emotions through the use of a weighted spatial-temporal emotional memory system, based upon knowledge relativity threads [2&3] human Autonomic Nervous System States. We artificially evolve the granularity of "emotional triggers" to determine importance thresholds which are context specific and relative to sensory inputs and environmental conditions to advance the capability of emotional learning and processing.*

We believe this has the potential to enhance artificial neural processing environments by allowing emotional memories and emotional learning to facilitate coalitions and cooperation between artificial neural intelligent software agents. We believe shared emotional states between intelligent software agents will more easily allow information sharing between agents, based on the "emotional reaction" to the systems sensory inputs, much in the way humans do.

Keywords: Artificial Emotions, Artificial Cognition, Artificial Psychology, Emotion Modeling

1. Introduction

For decades, psychological and psychiatric research has continued in parallel with the influx of advanced high performance computing and big data analytics architectures while early development has been undertaken into a hybrid fuzzy-neural processing architectures with an overarching objective to create genetic learning algorithms capable of human-like learning, reasoning and thought processing and, in particular, capable of learning about, processing, and utilizing emotions. Here we describe a modular architecture, based upon a mixture of neural structures and artificial connective neural tissue for evolving knowledge threads [2&11] within a system with weighted relativistic emotions to add flexibility and diversity to overall system capabilities. What we strive for is a continually adaptable neural processing system capable of dynamically adding and pruning basic building blocks of the neural system as the real-time environments (including emotional events) of the system change. The modular architecture artificial emotions architecture described here is based on fuzzy, genetic perceptron objects, called Cognitrons, The algorithms for which the Cognitrons are evolutionarily generated by the neural system or may be predetermined. The high-level artificial emotion architecture and information flow is illustrated in Figure 1.

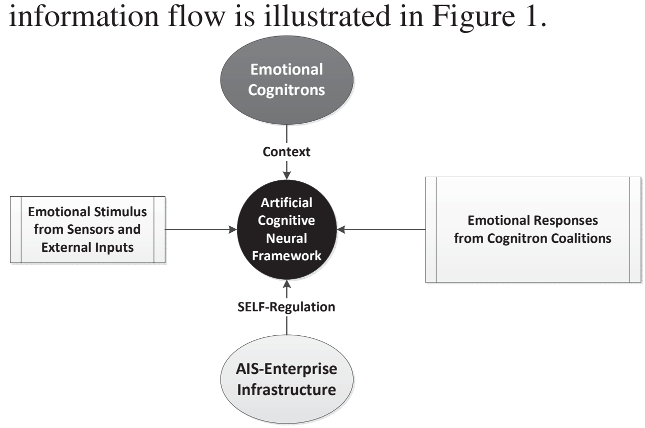


Figure 1 – Artificial Emotion High-Level Architecture

The purpose of this paper is to describe a neural framework for an Artificially Intelligent System (AIS) that provides “conscious” software agents; autonomous agents that range in functionality and are situated in the processing environment [7], allowing for development and use of emotions. Here we present structures within the AIS architecture required to provide artificial feelings and emotions and discuss the roles they can play. These agents would be actively involved in every instance of action selection, and at least potentially involved in each learning event. The pervasive, central role that feelings and emotions would play in the control structure of these conscious software agents mimics the roles they play in human cognition [8], and, over time, may give rise to clarifying hypotheses about human decision-making and several forms of human learning [1].

2. Descriptions of the Artificial Emotion Memory Components

In order to provide the AIS with emotional models, specialized roles for feelings in cognition, are created which all combine to produce motivations, actions and to facilitate emotional learning using artificially created and posited weighted stateful threads [2]. The various sorts of Cognitrons, perceptual, attentional, behavioral, and expectational, as well as their interactions will are described below:

1. ***Perception*** - Sensory stimuli, external or internal, are received and interpreted by perception creating meaning. Note that this stage is unconscious.
    a. **Early perception**: Input arrives through sensors. Specialized perception Cognitrons descend upon the input. Those that find features relevant to their specialty activate and broadcast their needs to the system. If perceptions are new and specialized perception Cognitron does not exist, then one is created followed by needs broadcasted.
    b. **Coalition perception**: Activation passes from Cognitron to Cognitron within the system. The Attention Manager brings about the convergence Cognitrons from different senses and into coalitions. *Pertinent thresholds of feeling/emotions are identified (recognized) along with objects and their relations by the perceptual memory system. This could entail simple reactive feelings based upon a single input or more complex feelings requiring the convergence of several different percepts; possibly defining an autonomic nervous system state* [9].
2. ***Perception to Preconscious Buffer***. Perception, including some of the data plus the meaning, is stored in preconscious buffers working memory [3]. These buffers may involve visuo-spatial, phonological, and other kinds of information. *Feelings/emotions are part of the preconscious perception written during each cognitive cycle into the preconscious working memory buffers. These buffers can be managed hierarchically or depth based upon time relevance to manage the storage or knowledge economy of the buffers.*
3. ***Local Associations***. Using incoming perception and nascent contents of the preconscious buffers as cues including weighted emotional threshold content, local Cognitron associations are automatically retrieved from transient episodic memory and from long-term associative memory. *Feelings/emotions are part of the cue that results in local associations from transient episodic and declarative memory. These local associations contain records of the agent’s past feelings/emotions in associated situations*.
4. ***Competition for Consciousness***. Attention Cognitrons, whose job it is to bring relevant, urgent, or insistent events to consciousness, view short and long-term working memory. Some gather information, form coalitions and actively compete for access to consciousness.

Competition can include attention Cognitrons from a recent previous cycle. *Present and past weighted feelings/emotions thresholds influence the competition for consciousness in each cognitive cycle. Strong affective content strengthens a coalition's chances of rising to consciousness.*

5. ***Conscious Broadcast***. A coalition of Cognitrons, typically an attention Cognitron and its covey of related information Cognitrons carrying content, gains access to the system and has its contents broadcast. This broadcast is hypothesized to correspond to phenomenal consciousness. *The conscious broadcast contains the entire content of consciousness including the affective portions.* The contents of perceptual memory are updated in light of the current contents of consciousness, *including feelings/emotions*, as well as objects, and relations. *The stronger the affect, the stronger the encoding in memory.* Transient episodic memory is updated with the current contents of consciousness, *including feelings/emotions*, as events. *The stronger the affect, the stronger the encoding in memory.* (At recurring times not part of a cognitive cycle, the contents of transient episodic memory are consolidated into long-term declarative memory.) Procedural memory (recent actions) is updated (reinforced) with the strength of the reinforcement influenced by the strength of the affect.
6. ***Recruitment of Resources***. Relevant behavioral Cognitrons respond to the conscious broadcast. These are typically Cognitrons whose variables can be bound from information in the conscious broadcast [4]. If the successful attention Cognitron was an expectation Cognitron calling attention to an unexpected result from a previous action, the responding Cognitron may be those that can help to rectify the unexpected situation. This is known as Cognitronic stochastic diffusion where importance thresholds per each impending need are dynamically processed. Thus artificial consciousness moves towards solving the relevancy problem in recruiting resources. *The affective content (feelings/emotions) together with the cognitive content help to attract relevant resources (processors, neural assemblies) with which to deal with the current situation.*
7. ***Action Chosen***. The behavior subsystem chooses a single behavior (goal context), perhaps from a just instantiated behavior stream or possibly from a previously active stream. *This selection is heavily influenced by activation passed to various behaviors influenced by the various feelings/emotions*. The choice is also affected by the current situation, external and internal conditions, by the relationship between the behaviors, and by the residual activation values of various behaviors.
8. ***Action Taken***. The execution of a behavior (goal context) results in the behavior Cognitrons performing their specialized tasks, which may have external or internal consequences. The acting Cognitrons also include an expectation Cognitron whose task it is to monitor the action and to try and bring to consciousness any failure in the expected results.

Figure 2 illustrates the AIS emotional trigger development that comes from steps 1-8, which includes a fuzzy inference engine that utilizes past behaviors and current circumstances to derive the emotional triggers.

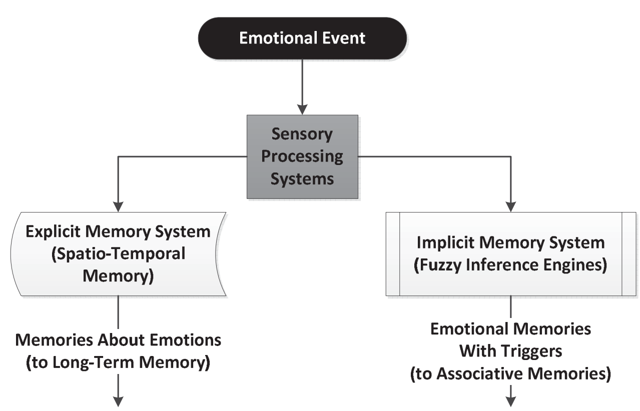


Figure 2 – Emotional Trigger Development

In humans, it is expected that the cycle of steps 1-8 happens 5-10 times a second, overlapping, with some happening in parallel [2].

3. Emotional Learning

Figure 3 illustrates the basic emotional learning architecture for the AIS. The genetic learning agents inherit initial states from the memory system and inherit the initial parameters for behavior from the behavioral center of the AIS. The consciousness mechanism, along with the mediator, controls the response of the learning agent, and direct its constraints based on the environment and the problems to be solved currently. This provides the priorities, preferences, goals, needs, and activation constraints (when you know you've learned something). The genetic agents (called genomes) adapt to the environment and gather information in order to make conclusions (learn) about the problem to be solved [13].

In the genetic environment, genomes are transferred to other agents, in order to speed up the adaptation of new generations of conscious agents in the behavioral environment. In the AIS, drives, priorities, and constraints influence emotions. The behavioral subsystem receives situations and computes actions, while memories provide personality parameters and the various conscious agents' sensitivities to emotional computation. We can think of the cross-connectivity of the neural layers as a matrix, and can compute emotional response from the column-wise fuzzy weightings, and the action response from the row-wise fuzzy weightings.

It is assumed that each matrix element E_{aj} represents an emotion. *Emotion* (a, j) of performing action *a,* in situation *j*. Given this, the genetic learning agents perform an emotion learning procedure, which has four steps [10]:

1. State *j*: choose an action in situation – (let it be action *a*; let the environment return situation *k*).
2. State *k*: feel the emotion for state *k* – *emotion* (*k*).
3. State *k*: learn the emotion for a in *j* – *emotion* (*a, j*).
4. Change state: *j* = *k*; return to 1.

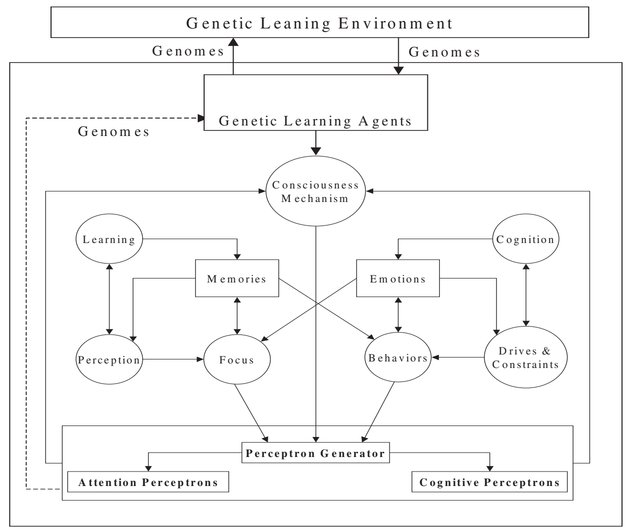


Figure 3 – Genetic Emotional Learning Architecture

This learning procedure is an emotion secondary reinforcement learning procedure. The learning constraint used in step 3 is:

$Emotion^0\ (a, j) = \text{genome}^0$ (inherited)

$Emotion^1\ (a, j) = Emotion^0\ (a, j) + emotion\ (k)$

This learning rule adds the emotion of being in the consequence situation, *k*, to the emotion toward performing action *a* in situation *j* on which *k* is the consequence. This drives us to define an ontology for each action driven by an emotional response for our AIS. Figure 4 illustrates this Cognitron Emotional Response Ontology.

4. Discussion: Human vs. Artificial Emotions

Many researchers have postulated that human cognition is implemented by a multitude of relatively small, special purpose processes, almost always unconscious communication between them is rare and over a narrow bandwidth. Experiments have shown that humans make judgments about the degree of resemblance to meaningful representations or prototypes during classification and identification of content; robins are judged as more prototypical birds than penguins [14]. Hence, the existence of gradients within concepts or what is known as continuous perception [15] Coalitions of such gradient processes find their way into consciousness. This limited capacity workspace of our cognition serves to broadcast the message of the coalition to all the unconscious processors, in order to recruit other processors to join in handling the current novel situation, or in solving the current problem. Thus consciousness in this theory allows us to deal with novelty or problematic situations that can't be dealt with efficiently, or at all, by habituated unconscious processes. In particular, it provides access to appropriately useful resources, thereby solving the relevance problem [9].

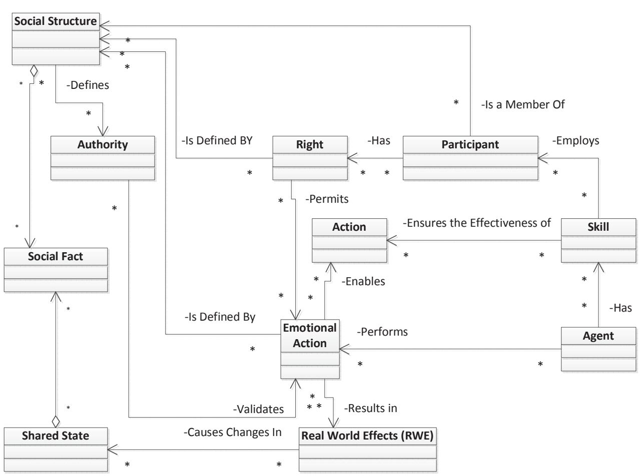


Figure 4 – Emotional Cognitron Action Ontology

All this takes place under the auspices of contexts: goal contexts, perceptual contexts, conceptual contexts, and/or cultural contexts. These may look like goal hierarchies, dominant goal contexts, a dominant goal hierarchy, dominant context hierarchies, and lower level context hierarchies. Each context is, itself a coalition of processes. Though contexts are typically unconscious, they strongly influence conscious processes [5].

Baars [4] postulates that learning results simply from conscious attention, that is, that consciousness is sufficient for learning. There's much more to the theory, including attention, action selection, emotion, voluntary action, meta-cognition and a sense of self. It can be seen as a high level theory of cognition.

The AIS emotional cognitive processing model suggests that software agents and robots can be designed to use feelings/emotions to implement motivations, offering a range of flexible, adaptive possibilities not available to the usual more tightly structured motivational schemes such as causal implementation, or explicit drives and/or desires/intentions [6].

So, what can we conclude? Explicit drives seem likely to suffice for quite flexible action selection in artificial agents. It is expected that feelings and emotions will be required in agent architectures requiring sophisticated learning. Specifically, it is expected that artificial feelings and emotions can be expected to be of most use in software agents or robots in which online learning of facts and/or skills is of prime importance. If this requirement were present, it would make sense to also implement primary motivations by artificial feelings and emotions.

References

1. Crowder, James A., John N. Carbone, and Shelli A. Friess. Artificial Cognition Architectures. Springer, 2014.
2. Carbone, J. "A Framework for Enhancing Transdisciplinary Research Knowledge." Texas Tech University (2010).
3. Crowder, James A., and John N. Carbone. "Eliminating Cognitive Ambiguity with Knowledge Relativity Threads."
4. Ackley, D. and Litman, M. 2002. Interactions between Learning and Evolution. Artificial Life XII. Addison-Wesley.
5. Anderson, J. 2004. Cognitive psychology and its implications: John r. Anderson: Worth Pub.

6. Ashcraft, M. 1994. Human Memory and Cognition. Harpercollins College Division, New York, NY.
7. Baars B. 1988. A Cognitive Theory of Consciousness. Cambridge University Press, New York.
8. Crowder, J. 2010. Flexible Object Architectures for Hybrid Neural Processing Systems. Proceedings of the 12th annual International Conference on Artificial Intelligence, Las Vegas, NV.
9. Crowder, J. 2012. Cognitive System Management: The Polymorphic, Evolutionary, Neural Learning and Processing Environment (PENLPE). Proceedings for the AIAA Infotech@Aerospace 2012 Conference, Garden Grove, CA.
10. Crowder, J. 2012. The Artificial Cognitive Neural Framework. Proceedings for the AIAA Infotech@Aerospace 2012 Conference, Garden Grove, CA.
11. Crowder, J. and Carbone, J. 2011. The Great Migration: Information to Knowledge using Cognition-Based Frameworks. Springer Science, New York, NY.
12. Crowder, J. and Friess, S. 2012. Artificial Psychology: The Psychology of AI. Proceedings of the 3rd Annual International Multi-Conference on Informatics and Cybernetics. Orlando, FL.
13. Crowder, J. 2014. Psychological Constructs for AI Systems: The Information Continuum. Proceedings of the 16th annual International Conference on Artificial Intelligence, Las Vegas, NV.
14. [Harnad, 1987] Harnad, S., Categorical Perception: The Groundwork of Cognition, Cambridge University Press.
15. Gärdenfors, Peter, and Mary-Anne Williams. "Reasoning about categories in conceptual spaces." IJCAI. 2001.

Sam and the Chinese Room

Roseli Gimenes

Universidade Paulista UNIP and PUC – SP TIDD

Abstract - The aim of this article is to describe the computer program SAM, introduced by Schank and Abelson, as well as its relationship with the Chinese room, presented by John Searle. Both programs, which belong to the field of Cognitive Science and Artificial Intelligence, are paths to approach the theory of knowledge and the ability of machines to tell a story.

Key words: Cognitive Sciences, Artificial Intelligence, Artificial Narratives.

1. Introduction

Cognitive Science has an interdisciplinary perspective with Philosophy, Anthropology, Linguistics, Cognitive Psychology, Neuroscience and Artificial Intelligence. This is fundamental for Psychology to understand how people think and act. Basically, Cognitive Science is the search for mind understanding and its goal is to explain how the mind works. Neuroscience studies the relations between mind and brain. Artificial Intelligence searches for model processes of human thinking in computer software and hardware. Linguistics investigates language use structures and what they can tell us about the mind. Anthropology observes mental thinking by analysing cultural features. Cognitive Psychology is the key to interdisciplinarity in these studies. [3]

What is Artificial Intelligence about? According to Boden [1] it is not about studying computers; it is about studying computer programming, that is, the use of computer programs and computing techniques to make a list of intelligence principles in general and human thinking in particular. Thus, computers would not be number triturators but symbol manipulators.

In this article, discussion will focus on John Searle's [6] inquiries on whether computers can think, the differences of the terminologies *strong Artificial Intelligence* and *weak Artificial Intelligence* Boden [1] and Schank and Adelson's [4] work commented by Searle as a simulation of human ability to understand stories.

If computers can think, can they narrate and understand stories?

2. Turing machine and the chinese room

In 1950, discussion on Artificial Intelligence was proposed by Alan Turing in one of his articles. [2]

Turing's Test aims at determining whether machines can show intelligent behavior. In Turing's original example, a judge talks to a man and to a machine which was created to perform just like a human being. The judge is not supposed to know when he is talking to the man and when he is talking to the machine. If the judge cannot tell one from the other it means the machine has passed the test. The conversation is limited to written texts (for example, a keyboard and a video monitor) so that the result does not depend on the ability of the machine to randomize words in audio.

Turing begins his article with some philosophic questions related to artificial intelligence, such as "Can machines think?". Considering "thinking" is difficult to define, Turing preferred to substitute his question for another one less ambiguous: "Is it possible to figure out digital computers which would be successful in the *game of imitation*? Turing believed there was an answer to that question. Then, he continues his article arguing against the great objections to "machines can think".

In 1980, John Searle wrote an article named *Minds, Brains and Programs* [7] proposing an argument known as *The Chinese Room* which was meant to prove that a computer program cannot give the computer a mind, comprehension or conscience, no matter how intelligent the machine may seem.

The *Chinese Room* is a thinking experience. It assumes there is a program which provides the computer with enough capacity to develop an intelligent conversation in Chinese writing. Searle supposes a man locked in a room. This man is given a paper pad with a text in Chinese. The man does not know written or spoken Chinese, so he does not recognize Chinese writing. A second paper pad, also in Chinese, is then given to that man with a syllabus of rules and he is supposed to relate the first information with the second one. The rules are in English and he knows this language. That allows him to relate a set of formal symbols with the other one. Formal is understood here as the ability to identify symbols by observing their shapes. Finally, the man is given a third paper pad which contains symbols in Chinese and he also receives some instructions in English to relate this one with the first and the second ones. The rules allow him to relate certain symbols in Chinese with certain configuration types. The man does not know it, but he is provided with texts with the following symbols: the first one is a syllabus; the second one is a story; the third one contains questions. The man in the room is then able to pass out the Chinese symbols which are correct answers to the questions. After some time, from the outside, Chinese people will say the man did really well for he answered the questions just like a Chinese speaker would have done. According to Searle, the answers are good enough, but the formal symbols in Chinese are meaningless. The man behaved as the computer: he executed computational operations; it is only a computer program instance.

2.1 John's Propositions

In *Minds, Brains and Science* [6] and *Minds, Brains and Programs* [7]. Searle discusses strong Artificial Intelligence. He points to weak Intelligence in the sense of valorizing the computer for the study of the mind, as a powerful tool to formulate and test hypotheses precisely. On the other hand, strong Artificial Intelligence sees the computer not as a tool for the study of the mind, but as a computer program, and the brain as a digital computer. Thus, mind is to the brain as the program is to the computer hardware.

According to this conception, human mind would not have anything biological. Any physical system which had a correct program with the correct inputs and outputs would have a mind.

Searle [6] mentions that some researchers of strong Artificial Intelligence such as Simon, Newell, Dyson, Minsky and Mc Carthy state, among other things, that intelligence is a matter of physical symbol manipulation; there are no metaphors; as far as evolution is concerned, computers would have advantage over human beings and that even thermostats have beliefs. According to Searle [6]:

> *Mc Carthy says that even "machines as simple as thermostats have – one can say – beliefs". I admire Mc Carthy's courage. Once I asked him: 'Which beliefs does my thermostat have?' And he answered me: 'Mine has three beliefs: it is too hot here, it is too cold here and it is right here.' As a philosopher I appreciate these statements, they are reasonably clear and admit a simple and decisive refutation*. [6: 30]

According to the author, his refutations to the ideas of these researchers is that a digital computer has purely formal operations of abstract symbols – sequences of zeros and ones printed on a tape. These symbols have no meaning and no semantic content.

Back to the *Chinese Room*, [6] states that if the man cannot understand Chinese, a digital computer cannot as well because the computer has syntax, but no semantics:

> *And the digital computer can only have formal symbols because the operation of a computer occurs in terms of its capacity to run programs. And these programs are run in a purely formal way, that is, there is no semantic content*. [6:33]

Concerning the objections made by the researchers about the *Chinese Room*, Searle [7:72-80] points out:

> *System Objections (Berkeley): system cannot understand anything because system does not have anything that man does not have. If the man cannot understand, then system will not understand. System is merely a part of man.*
>
> *Robot Objection (Yale): a robot does not have intentional actions; it moves as a result of its electric systems and its programming. The creation of a program does not produce intentional states. All that it is done is to follow formal instructions about the manipulation of formal symbols.*
>
> *Brain Simulator Objection (Berkeley and MIT): Schank's works – which will be presented later - are mentioned here. According to Searle, the problem with the brain simulator is that it simulates wrong things about the brain. It simulates the formal structure of neural activities sequences, its causal properties and its ability to reproduce intentional states.*
>
> *Combination Objection (Berkley and Staford): Searle sees the robot as a mechanic puppet.*
>
> *Other minds Objection (Yale): Cognitive Science presupposes mental state can be accessed just like physical sciences presuppose physical objects can be accessed.*
>
> *Several groups Objection (Berkeley): Searle says that it trivializes strong Artificial Intelligence project by redefining it as anything that can produce and explain cognition artificially*.

His most important question comes out at the end of both his articles: Can a machine think? Can a digital computer think?

The author asserts that our brains are like digital computers because they carry out any number of computer programs. And our brains can think.

According to the author, the digital computer cannot think because it has only got syntax information. Thinking is more than meaningless symbol manipulation. What a digital computer can actually do, however, is to simulate human behavior. Even so, simulating is one thing and being real is another. When a storm is simulated, a shelter does not need to be reached because we know it is not for real, it is only a simulation.

Thus, Searle converges both articles like this: computers are not minds; a mind does not work by just activating a computer program. A computer does not make the causal connections a brain makes. Mental states are biological phenomena.

2.2 Schank's and Abelson's positions

Robert P. Abelson, a psychologist in Yale, and Roger Schank, Artificial Intelligence Researcher, wrote the book

Scripts, Plans, Goals and Understanding – An inquiry into human Knowledge structures [4].

As it has been mentioned before, Schank is quoted by John Searle [7] for his project in Yale with machines which could understand stories. The book mentioned above became a classic and has been quoted by many social scientists:

> *I will analyze the work of Roger Schank et al. in Yale because I am more acquainted to it than to other similar works. Besides, it provides a clear example of the kind of work I wish to look into.* [7:65]

On his trajectory as a researcher, Schank showed how computers were able to process daily sentences in the English language as well as read newspaper articles. In 1976, he launched the first computer program which could read newspaper stories. With his projects, he realized computers had problems with memory – ability human beings have – but, on the other hand, computers could actually "remember" whole volumes, which is impossible for humans. What computers lacked was the ability to generalize. They could read a story, but they were not able to recognize aspects of a certain story in another one they had read before. Computers did not understand because they did not match similar occurrences. Schank realized the ability of generalizing and memory were interconnected.

Schank's connection to Abelson made him study the learning process. If he observed how people learn, he could apply this knowledge to computers and make them understand stories. Schank started to build up real world events for the computers. People remember things all the time. If things do not happen the same way, humans ask why. Hence, computers should have expectations.

Schank realized people store memory in packs. Man reconstructs several events when he needs to remember something. This was the basis, the dynamic memory, a theory of remembering and learning.

Schank's approach was more cautious than the approach of other researchers of strong Artificial Intelligence such as Simon and Newell. He defends computing as a means of testing cognition theory. According to Searle (1990, 2005), a weak Artificial Intelligence:

> *According to weak Artificial Intelligence, the main value of computer for the study of the mind resides in the fact that it provides us with an extremely powerful tool. For example, the computer allows us to formulate and test hypotheses in a more rigorous and precise way than before.* [7: 67]

The book mentioned is a theory of Cognitive Sciences about the understanding of stories. Basically, it suggests that meaning and cognition occurs by means of comprehension of concepts and sentences.

In this study, Schank and Abelson [1] point out that the work is not only about Psychology, Artificial Intelligence or Linguistics, but about the three areas together. Interdisciplinary. And that lead them to think about causal chain. Thus, interpretation is described as a filling in the blanks in a causal chain:

> *Psychology which studies knowledge systems wants to know how concepts are structured in the human mind, how such concepts develop and how they are used in comprehension and behavior. Artificial Intelligence researchers want to know how computer programs can understand and interact with the external world.* [4: 1]

The book is distinguished by chapters which discuss:

> *Scripts: composed of branches, roles, states, entry conditions and results. People act appropriately because they have world knowledge. A story is understood because people fill in missing information when reading.*
>
> *How do people organize all their knowledge in comprehensible sequences? How do people know which behavior is appropriate in a certain situation? (…) People know how to act appropriately because they know about the world they live in. What is the nature of this knowledge?* [4: 36]
>
> *Understanding is, then, a process through which people find what to see and listen in pre-stories of group actions they have already experienced. (…) Scripts intend to contain specific knowledge that people have. Most comprehensions have a basic script*. [4: 67]
>
> *Plans: they are means to reach for satisfactory objectives. There are USE plans, for example. They are made of general information so that "actors" can reach their targets. As the authors say* [4: 71]: *"A plan is a series of actions projected to reach a goal."*
>
> *Goals: There are seven forms of goals: satisfaction, enjoyment, achievement, preservation, crisis, instrumental and Delta. Instinct, necessity, values, way of living, beliefs. The authors classify the goals: a prime goal, a specific and substitute goal, a suspended goal and stylistic goals*. [4: 103]. *The authors ask the question: "Where do goals come from?" For them, the answer lies on what they define as 'themes'*. [4: 119].
>
> *Themes: the authors ask themselves: 'Where do goals and plans come from?', 'What are stories about?' For example, interpersonal relationships. In a theme list, people's goals are determined by social rules.* [4:132].

2.3 SAM computer program

In this book, intended to present computer programs, they try to construct intelligent machines taking into account human natural language processing and natural language processing in computers. The authors show several programs which were created in order to make machines understand stories. TALESPIN, PAM and SAM.

What is SAM? It is a computer program: Script Applier Mechanism. [4: 177]. This program was presented in Yale and it was designed so that stories could be understood as scripts. SAM links a series of causal concepts.

The track described, pointed out by Searle [7: 67], is the restaurant. A person goes to a restaurant, orders for something to eat, pays and leaves. They are short narratives, apparently uninteresting, but with great potential. The scripts have metadata, they describe the basis of the event and the system recognizes which script it is supposed to use: instrumental relations, places. These scripts are social rules, procedures, conventions. [4:36], scripts are powerful elements for cognition and comprehension of the world. If the scripts are known, we learn with the experience.

SAM [4: 176] is a kind of script “expert”, just like some other programs such as FRUMP – the summary of a newspaper story based on scripts, TALESPIN – a story narrator which uses the program of plans and goals, and PAM – a program which “understands” stories by using plans, goals and themes.

Anyway, by the time the authors were working with SAM, there was not any other computer program enabled to understand stories.

Schank and Abelson [4] describe SAM by giving an example, as follows:

> *Input: John went to a restaurant. He sat down. He got mad. He left.*
>
> *John was hungry. He decided to go to a restaurant. He went to one. He sat down on a chair. A waiter did not go to the table. John became upset. He decided he has going to leave the restaurant. He left it*. [4: 189]

A script is a pre-organized inference chain related to a specific routine situation [4: 36]. It is a sequence of conceptualizations with some variables (*script* variables).

The restaurant *script* intends to capture a person’s (actor’s) knowledge about the sequence of events that occur when this person goes to a restaurant.

(1) Actor goes to the restaurant

(2) Actor sits

(3) Actor orders the waiter for a meal

(4) Waiter brings the meal to the actor

(5) Actor eats the meal

(6) Actor gives money to the restaurant

(7) Actor leaves the restaurant.

Schank and Abelson [4: 46] believe that people understand a story (an event) more easily when they have experienced it many times before. This experience is decoded in a script that, once constructed, is attached to the long-term memory and does not need to be recapitulated any longer. The script has a strong prediction power and failures in its structures can be recognized.

Actually, the detection of failures in a script is more related to the way it is organized than to the codified information. This organization, in turn, can be dynamically codified.

Initially, in the first versions of the theory, a script was seen as a structure which represented separate temporal sequences; one script did not relate to others. As the model was developed, the authors started to see scripts in a more modular way from which the interconnectivity of scripts starts to be investigated.

In this new conception, [5: 181-200] a model named Memory Organization Packets (MOPs) is developed. MOPs were meant to cut the script into small units called scenes. Then, the same scene could be shared by many MOPs because (a) it would not make any sense that the same information were represented in different “places” and (b) that would indeed facilitate learning. The example given can form the following scheme:

MOP 1: VISIT TO THE DOCTOR

MOP 2: VISIT TO A LAWYER

Shared scene: BEING IN A WAITING ROOM

A theory on memory organization “as a whole” is necessary when it comes to the dynamic modification of a MOP. Schank [5] then develops a theory named Dynamic Memory. His proposal is to connect MOPs the same way MOPs connect scenes. Hence:

MOPs would be connected by a set of abstraction hierarchies.

An example:

MOP: VISIT TO AN OFFICE (More abstract level).

MOP 1: VISIT TO A DOCTOR (Upper level MOP instance).

MOP 2: VISITI TO A LAWYER (Upper level MOP instance)

The MOPs would be connected by a set of packet links, connecting MOPs to other MOPs which frequently occur together in a broader context.

An example:

MOP: BUSINESS TRIP

MOP 1: TRAVELING BY PLANE

MOP 2: CHECKING INTO A HOTEL

MOP 3: BUSINESS LUNCH

2.4 SAM and the relation with a case study

Schank and Abelson [4: 177], still asking themselves 'Where do scripts come from?', point out that language acquisition is script acquisition.

As an example, Schank and Abelson [4: 228-237] test *script* with a child. The same script is narrated to the same child in three different moments: at the ages of two years and six months, three years and four months and four years and two months.

If the child has already been to a restaurant, he will answer the questions by using memory references: arriving, sitting, ordering, paying and leaving. This sequence may similarly repeat if the script is going to a pet shop: ordering, paying and leaving, for example.

The authors analyze that the sequence of actions is a crucial factor for the memory of the child and point out that the concept of memory was strongly activated in the first experience. Memory has made connections in context. Hence, scripts are learned to connect events and they are organized by goal structures which are used to meet the needs of these goals.

In this experience, the child is asked to tell stories. In the beginning of the experiment, when the child is younger, many details are told and these details are different from those which really matter in the scripts for adults. As the child grows older, he changes his system to tell stories: from a model based on scripts to a model based on plans. In that case, the program SAM – based on scripts – gives way to the program TALESPIN – based on plans.

The child is told two stories. In one of them a man gets on a train, sits, is robbed and leaves. In the other story, the man who left the train goes to a restaurant. He gets in, sits, orders, eats and when he is going to pay he realizes he does not have any money so he will have to wash the dishes.

At first, the child does not understand the man was robbed because that child has not experienced a robbery before. This is also the reason the child does not understand why the man has to wash the dishes in the restaurant, as a way of paying for his debt. The child's script is: the man gets into the restaurant, sits, orders, is served, pays and leaves.

The use of scripts depends on the perfect comprehension and the conditions under which someone decides the use a given script.

Schank and Abelson [4: 237] conclude the experiment with the child by asserting that 'the limit of comprehension is the limit of the world knowledge comprehension': *The child understands to the limit of his world knowledge (…) You know what you can understand. This is true for children and for adults.*

3. Conclusions

John Searle [7: 67] pointed out that Schank and Abelson [4] created SAM – the computer program – to simulate human ability to understand stories. According to Searle [7], human beings do understand a story when they can answers questions about it, even if some information necessary to answer these questions are not explicit in the text.

About Schank and Abelson's [4: 178] restaurant script, Searle [7: 68] comments that a man went to a restaurant and ordered for a hamburger. When his order came he realized the hamburger was toasted so he left the restaurant feeling furious. He did not pay the bill and did not tip the waiter.

The authors observe that if the following question was asked: 'Did the man eat the hamburger?' anyone would probably answer: 'No, he did not.' This would be the answer, given the circumstances.

A man went to a restaurant and ordered for a hamburger. When his order came he was quite satisfied with the meal and before paying for the bill he tipped the waitress generously.

If the following question was asked: 'Did the man eat the hamburger?', anyone would certainly say: 'Yes, he ate the hamburger.'

According to Searle [7: 68], Schank and Abelson's machine [4] is enabled to answer questions like those about restaurants. To do so, the machine needs to be programed with the information human beings have about restaurants.

When the machine is provided with the story and a question is asked, the machine will print out the same answers humans were expected to give.

To Searle [7: 68], people who back up strong Artificial Intelligence assert that this question-answer sequence indicates not only that the machine is simulating a human ability but also that it understands the story. And what the machine and its program do explains human ability to understand stories and answer questions about it.

To throw aside the statements above, that is, to discard the idea that SAM – the computer program created by Schank and Abelson [4] – understands the story, Searle launches the argument of the Chinese Room.

And they conclude by stating that if the man cannot understand Chinese, the machine cannot as well.

Schank and Abelson [4: 237] say that 'the limit of comprehension is the limit of the world knowledge'. Taking that into consideration it is possible to say that, just like the child observed during the experiment, anyone can answer questions to the limit of his knowledge or to the limits of the script, plans, goals… which, in a certain moment, we have.

4. References

[1] Margaret A. Boden. "What is artificial intelligence?" In: Artificial intelligence and natural man. New York: The harvest press limited, 1977.

[2] William S. Robinsom. "The Turing Test". In: Computers, minds & robots. Philadelphia: Temple University Press, 1992.

[3] Gregory Robinson and Bridget Riegler. "Cognitive Psychology". Applying the science of the mind. Boston: Pearson, 2004 (2009).

[4] Roger C. Schank; Robert P . Abelson "Scripts, plans, goals and understanding. An inquiry human knowledge structures". New Jersey: Lawrence Eribaum Associates, Inc. Publishers, 1977.

[5] Roger Schank; Alex Kass. "Knowledge representation in people and machines". In: Umberto Eco, Marco Santambrogio, Patuzia Violli (Eds). Meaning and mental representations. Indianpolis: Indiana University Press, 1988.

[6] John Searle. "Can computers think?" In: Minds, brains and science. Penguin book, 1984 (1991).

[7] John Searle. "Minds, brains, and program". In: Margaret Boden. A .The philosophy of artificial intelligence. Oxford: Univ.Press, 1990 (2005).

Turing test does not work in theory but in practice

Pertti Saariluoma[1] **and Matthias Rauterberg**[2]

[1]Information Technology, University of Jyväskylä, Jyväskylä, Finland

[2]Industrial Design, Eindhoven University of Technology, Eindhoven, The Netherlands

Abstract - *The Turing test is considered one of the most important thought experiments in the history of AI. It is argued that the test shows how people think like computers, but is this actually true? In this paper, we discuss an entirely new perspective. Scientific languages have their foundational limitations, for example, in their power of expression. It is thus possible to discuss the limitations of formal concepts and theory languages. In order to represent real world phenomena in formal concepts, formal symbols must be given semantics and information contents; that is, they must be given an interpretation. They key argument is that it is not possible to express relevance in formal concepts only. Consequently, computational models can be valid in some specific interpretations, and the Turing test can therefore only work in specific tasks.*

Keywords: Turing test, autonomous technologies, design science, mind, machine, artificial intelligence

1 Introduction

The Turing test is probably the best-known thought experiment in Artificial Intelligence (AI) [13, 14]. Its goal is to answer one question: Can machines think? Thus, tests form an important component of the argument comparing human and machine information processing [5, 11, 19]. Since the discussion on intelligence of machines underpins much of the modern computational cognitive psychology and philosophy of the mind, it is also essential in developing robots and autonomous systems [2, 7,].

The Turing test is essential, as it enables computational thinking to be viewed through multiple lenses. The notion of computational thinking has many different forms but here, it refers to the use of computational concepts to investigate intellectual information processes [15]. Prime examples of such thinking are the Turing machine and modern computers [13, 14].

Computational thinking, that is, investigating information processes in different contexts by means of computational (algorithmic and mathematical) concepts, is a vital process today, as it has allowed us to realize such intellectual processes by means of computers, information systems, and robots. In a social context, this means that machines are used for new and more demanding tasks. Thus, the issue of linking machine and human intelligence has become central to scientific thinking.

Essentially, the Turing test is an imitation game. Like any good experiment, it has two conditions. In the case of control conditions, it is assumed that there is an opaque screen. On one side of the screen is an interrogator whose task is to ask questions and assess the nature of the answers. On the other side, there are two people, A and B. The task of A and B is to answer the questions, and the task of the interrogator is to guess who has given the answer. In the case of experimental conditions, B is replaced by a machine (computer) and again, the interrogator must decide whether it was the human or the machine who answered the questions. A teletypewriter is used to eliminate the problems caused by the quality of the voice communication [14].

The decisive criterion in these experiments is the capacity of the interrogator to say whether the answer was given by human or machine. If the interrogator cannot do this, then the machine has passed the test. Therefore, the outcome of the experiment is that machines can think, as they can perform human tasks in such a way that it is impossible for a competent observer to see the difference between human and machine.

It is obvious that the Turing test is a particularly clever idea, but in what way exactly? Generations of renowned researchers have considered all aspects of the test [1]. The test itself was an innovation intended to justify Turing's computational theory of mind, which was criticized even before the test [13, 14]. Perhaps one of the most outspoken of the pre-test critics was Turing's close colleague and friend Ludwig Wittgenstein [17, 18]. Of course, his critical views became public after the publication of the test, but as Turing and Wittgenstein had close contacts in Cambridge, their differences of opinions in respect to computational thinking had developed before the Second World War [8]. Even today, we must still ask the question: Does the mind compute, hyper-compute, or even more? [9]

Since the Turing test was published, it has been discussed extensively, and many researchers such have investigated it[4, 5, 11, 12]. This is unsurprising, as the test is still considered somewhat elusive. On one hand, it is easy to identify machines that can beat people in their particular fields, for example, chess machines and pocket calculators. On the other hand, however, no general man-like Leviathan exists.

Studying the division of opinions around computational thinking and the Turing test is important, as the clarity in this issue would enhance the conceptual clarity around all of computational thinking. In order to analyse the conceptual properties of Turing's test, it is first worth considering some of the major critical arguments against the idea that human thinking is realizable using machines or that it is possible to implement all human information processes by means of computational machines. We then need to look at the practical problems specific to computers' information processing to facilitate an understanding of any hidden conceptual issues that could shed light on the problems and criticisms relating to Turing's work. Finally, it is essential to ask what implications such criticisms have for the role of the Turing test in theoretical and practical computational thinking.

2 Critique of machine thinking

The idea of the human as a computing machine has raised a number of critical points, which are important in analysing the scope and limitations of computational thinking. Presumably, this is why Turing gave so much space to the main critiques of the time when presenting his thought experiment [14]. Some of the critical arguments, such as "ESP," "theological" and "head in the sand" objections, can be put aside as they do not have much relevance in modern science. ESP, for example, is an unclear phenomenon, and the consciousness argument can also be disregarded, as it is not relevant in investigating the ideas behind the Turing test. In fact, Turing never claimed that his test was a test of consciousness. However, there are many important points in this discussion that are worth considering here.

As noted above, Ludwig Wittgenzstein was apparently the first critic of Turing's thinking; fundamentally, he really disliked the idea of people being machines [16, 17, 18]. In the Bluen and Brown books, for example, he wrote, "The trouble is rather that the sentence, 'A machine thinks (perceives, wishes)' seems somehow nonsensical. It is as though we had asked 'Has the number 3 a color '.[18]

Wittgenstein put forward a number of points against people being computers, beginning with the idea that people feel pain and machines do not. One could call this a biological argument. However, the most relevant counterargument here is the argument concerning "seeing as," i.e., interpreting elementary symbols such as percepts and words. To Wittgenstein [16, 17, 18], anything we perceive can have a multitude of aspects or interpretations. Words, for example, impart their meaning as soon as they are used. For Turing, this problem was not an issue, because he regarded symbols as a given. In essence, numbers are well-defined symbols, but Wittgenstein critically asked whether words and images are like numbers, that is, well-defined objects, and recognized the problems of automatic encoding in his ostensible "language game." Consequently, he spent the last decades of his life analysing the process of giving meanings to symbols.

Wittgenstein's point was subsequently presented in an advanced form by Searle [12]. The latter points out that machines are not able to concentrate on essential aspects of chess positions; rather, unlike people, they scan all possible alternatives mechanically in order to find a solution. This, of course, is as true today as it was in the seventies. People are still as capable of concentrating on the essentials of chess as they are on giving meanings to words, patterns, and any other type of sensory input.

A somewhat different critical argument was presented by Lady Lovelace [14]. Her argument was focused on Babbage's analytical engine, which was one of the first versions of the thinking machine. Her claim was that such engines could not initiate anything, particularly anything new. Turing could not accept this argument, but neither did he present a very convincing counterargument.

Finally, another point worth looking at again is the "informality of behaviour argument," which was also considered by Turing [14]. The core of this argument is that there are no rules that can mechanically explain human behaviour. People may stop when they see a green light, but it may also be that they do not. Thus, human behaviour cannot be determined fully by a given set of rules in the way that machines are. Instead, people apply rules flexibly in their lives and can usually decide which rules are relevant in any given context.

The above arguments are sufficient for our purposes, considering the scope of this paper. Our next question is whether it is possible to ascribe any practical meaning to these points or whether they are, in fact, merely conceptual. Second, it is vital to consider why it is so difficult for machines to select between essential and inessential courses of actions.

3 Practical problems with machine thinking

Real computational systems also have difficulties with meaning and content. Problem-solving programs are a prime example of human thinking realized by machines. A classic example of this is the chess machine, which is programmed to beat human chess masters.

In implementing problem solvers, two fundamental problems must be addressed. First, computers searching huge problem trees have to deal with an issue known as exponential search, which seems not to trouble people. Exponential growth in a search means that when the depth of the search is increased, the size search tree grows very fast. Thus, chess-playing computers have to generate millions of moves to reach human levels of performance, since people generate some 50 moves per problem position [6]

A somewhat similar problem is pattern matching. While a CEO can easily see essential patterns in economic development, it is very hard for computers to develop a pattern-matching system to facilitate such strategic management. Indeed, in many cases, pattern matching is even difficult for people. A similar problem is conflict resolution, where there may be several possible patterns leading to potential action, but one has to identify the best choice.

These practical problems are connected to original critiques. If it were possible to identify the right aspects of patterns such as words or signs syntactically, pruning search trees would not be a problem. It would only be necessary to indicate which branches could not be generated. In the same way, pattern recognition problems are easy to solve, if only it were possible to solve the problem of aspects. Both of these problems arise from the fact that syntax does not provide sufficient information for semantics selection [12].

4 Relevance: The rebuttal of computational concepts in modelling the human mind

Scientific and other representational languages have their scope and limitations. For example, it is not possible to find a natural number that expresses such issues as the relationship between the side and the diagonal in a square or the value of PI. To represent such entities, it is essential to use real numbers. Similarly, in behaviourist psychology, it is not possible to consider the properties of human memory or mental images, as they are more than simply stimulus-response concepts. Finally, had we restricted ourselves to Dalton's concept of the atom, modern nanophysics would also have been impossible. In science, progress is always about finding new concepts and exceeding the limitations of old ones [3, 10].

The limitations in the basic concepts of a scientific approach can be characterized by *power of expression* [10]. In essence, power of expression describes the limits of a conceptual system and expresses what can be thought when a particular set of concepts is being used. It can be used very naturally in this case, as it enables us to questions the limits of computational concepts and therefore the real meaning of Turing's test.

Put simply, computational concepts are based on abstractions devised by Turing. When he postulated the Turing machine, Turing assumed that he was similar to a mathematician describing how to solve any mathematical problem. What the mathematician does is to manipulate symbols on squared paper following a given set of rules. For the sake of simplicity, Turing assumed that the machine had an infinite tape featuring zeros and ones. The task of a mathematician following the rules was to manipulate the numbers on the tape. The numbers were supposed to represent symbols, which could be numbers, but also Chinese symbols. Thus, like many others after him, Turing assumed that the Turing machine was, in some senses, a model of the human mind.

However, Turing did not specify how the real world symbols and their meanings were associated with the Turing model. The associations between the number combinations on the tape, the symbols, and the references to symbols are given but not processed. Thus, the most important action is omitted from the computational thinking process, which involves figuring out how computational models can be combined with reality.

In mathematical concepts and the metascience of mathematics, *relevance* refers to the rule determining which elements of any mathematical set (of elements or functions) belong to one category (relevant) and which to another (irrelevant). In terms of Turing machines, one should be able to say which combination of zeros and ones are relevant and which are not. However, this is impossible in mathematical or formal concepts, as the theory language does not have the power of expression typical to natural languages. Since Turing machines and mathematical models are constructed by means of abstracting semantic and thought content, one can no longer present what is relevant in some concrete context.

When the contents are abstracted, it is impossible to produce sense-making semantics; in other words, it is impossible to define what is true and what is false or what is right and what is wrong. This means that only interpretation in terms of real world concepts, i.e., programmed semantics, makes it possible for AI systems to have any relevance.

Turing was not the first person that encountered this problem of linking formal systems to reality at the start of the last century. Before him, Ludwig Wittgenstein) had seen the same problems. In his "Tractatus-Logico Philosophicus," he gave a logical explication for the problem of human experience limitations [16]. Obviously, he had noticed that logical (or syntactic) symbols were void of meaning, and this is why in his later philosophy, he adopted the process of giving meaning as his topic. Of course, this explains his criticisms of Turing's computational thinking theory; he had seen the very problem that Turing had simply brushed under the carpet.

In order to be able to model reality, one must have a representation that contains the correct information. Otherwise, the model cannot be true and will misrepresent reality. "Correct" means that the symbols in the model have the correct semantics, and this presupposes that the information represented is right.

In fact, this strong AI premise is often defined to mean that the system has the right output when it has the right input. Or course, "right" in the given definition means precisely same as having the correct representational contents, as discussed above. "Meaning giving" thus becomes the most important problem. However, as Searle pointed out, syntax cannot generate semantics [12]. This is why the origins of meaning giving must be sought in human conceptualization and judgment processes, and this is why they are outside the framework of computational modeling.

The core explanation for the limitations in computational thinking is in the very abstraction process that creates symbolic information. The abstraction sets aside semantics and information contents. This is why syntactic models cannot represent a concrete state of affairs without interpretations. Proposition 3+5 = 4 is true, but that does not aid us in marketing fruit in a marketplace unless we know whether the formula describes apples, pears, bananas, or money. Similarly, it is valid to infer that "Napoleon was the Emperor of France" from the true fact that "the moon is not cheese," and "if the moon is not cheese, Napoleon was the Emperor of France." Whether the inference makes any sense is another issue. It is impossible to illustrate any relevant connection between Napoleon's role as Emperor of France and the fact that the moon is cheese.

Thus, the core issue seems to lie in the notion of relevance. Unless one can show that the semantic and information contents of a Turing model are relevant, the system does not work. Formal languages are void of content, and it is therefore impossible to determine their relevance in computational concepts. Computational concepts do not have the power to express relevance and for that reason, one needs an additional language and a scientific process to determine the relevancies and to implement systems in the real world. This, of course, works often in the modern world.

When the contents are abstracted, it is impossible to achieve sense-making semantics. It is also impossible to define what is true and what is false or what is right and what is wrong. This means that only interpretation in terms of real world concepts, namely, programmed semantics, makes it possible for AI systems to have any relevance.

However, in terms of the Turing test, the relevance requirement is devastating. It shows that machines can perform as well as people in processing information, but it does not show that people are computers. Obviously, people have capacities far beyond computational languages, and the failure of the Turing test as proof that a machine can function as a human does not influence the practical aspect of computational modelling.

5 Long live AI

Modern artificial intelligence is continuously achieving more interesting practical results. Autonomous systems, which broadly speaking, can redefine their goals during operation, are a good example of the capacity of emerging technologies. However, a merely intuitive interpretation of computational systems cannot make sense. It is time to go beyond the limits of computational concepts and admit that we need a new way of thinking that will incorporate computational representations of information contents.

In sum, this line of argument shows us that Turing's test is insufficient as proof of the identity of man as a machine. However, it can still play a very important role as the testing logic for computational models [5]. One must take the human mind as a criterion of how well machines have to perform to succeed in their main function, which is to replace human intellectual work. Only if a system passes the Turing test is it possible to view it as having a practical application. Thus, although the Turing test is theoretically dead, it certainly has a future in terms of designing new technical applications.

The important of Turing's logic is actually growing as technologies are improving. Autonomous systems, which are partly able to define their own goals, are a good example of what is in store in the future. Autonomous systems such as autonomous cars or flying devices are capable of changing their concrete goals depending on the situation. In general, autonomous systems form one of the key future developments in technology.

The core social importance of autonomous systems is rooted in their capacity to replace people in tasks that were traditionally carried out by people. In the case of many such tasks, it has never been possible to realize them technically. Typically, in those tasks, the system—the people and the machine—has to redefine ill-defined goals or redefine decision spaces.

Identifying solutions to such tasks is of central importance in future artificial intelligence. The classic example, as noted above, is the chess-playing machine. After not working well for five decades, it was suddenly possible for the machine to beat world champion chess player, Kasparow. This proves that there is a machine capable of performing an intellectual task as well as a human being. Of course, this was a machine with a special purpose; however, if chess were economically relevant, Deep Blue could have replaced all chess players, since it can be duplicated in millions.

The Turing test is vital for the technical realization of new intellectual tasks [20]. It also offers logic in assessing the performance of autonomous and other technical systems, and in designing ways to replace human work and leave new types of tasks to humans. The Turing test is essentially a test of performance rather than a test of how information is processed. Deep Blue, for example, has very little in common with human chess players in the way it processes information. However, commonalities are not essential from a technical or economic point of view. What is essential is that something can be achieved, and from this perspective, the Turing test is an excellent conceptual tool for designers. From ontological point of view, it is all too behaviouristic to be a test for can machines think like people. Therefore we can dinstinguish between *structural* equivalence and *functional* equivalence. The Turing test can help to identify functional equivalence but not structural equivalence.

6 References

[1] Tarek R. Besold. "Turing Revisited: A Cognitively-Inspired Decomposition. In: V.C. Mueller (ed.) *Philosophy and Theory of Artificial Intelligence* – SAPERE 5 (Studies in Applied Philosophy, Epistemology and Rational Ethics; Vol. 5, pp 121-132). Berlin: Springer. 2013.

[2] Margaret Boden. "Artificial intelligence and natural man". MIT Press. 1987.

[3] Nick Bostrom. "A history of transhumanist thought." *Journal of Evolution and Technology*, 14, (pp. 1-25). 2005.

[4] Mario Bunge. "Scientific research". Springer Verlag, 1967.

[5] Luciano Floridi., Taddeo, M., & Turilli, M. "Turing's Imitation Game: Still an Impossible Challenge for All Machines and Some Judges—An Evaluation of the 2008 Loebner Contest". *Minds and Machines*, 19, 145-150. 2009.

[6] Richard French. "Dusting off Turing test". *Science*, 336, (pp. 164-165), 2012.

[7] Adriaan de Groot. “Thought and choice in ch*ess”*. Mounton. 1978.

[8] Stevan Harnad, & Scherzer, P. “First, scale up to the robotic Turing test, then worry about feeling”. *Artificial Intelligence in Medicine*, 44, (pp. 83-89). 2008.

[9] Ray Monk. “Ludwig Wittgenstein. The Duty of Genius”. Jonathan Cape. 2009.

[10] Matthias Rauterberg. “Hypercomputation, uncounsciousness and entertainment technology”. In: Panos Markopoulos (Ed.), Fun and Games, LCNS 5294. (pp 11-20). Berlin: Springer Verlag. 2008.

[11] Pertti Saariluoma. “Foudational analysis. Presuppositions in experimental psychology. Routledge. 1997.

[12] Ayse Saygin, Cicekli & Varoll Akman. “Turing test – 50 years later”. *Minds and Machines*,10, (pp. 463-518). 2000.

[13] John Searle. “Minds, brains, and programs”. *Behavioral and Brain Sciences,* 3, (pp. 417-457). 1980.

[14] Alan Turing. “On computable numbers, with an application to the Entscheidungsproblem”, *Proceedings of the London Mathematical Society*, 42, (pp. 230-265). 1936.

[15] Alan Turing. “Computer machinery and intelligence”. *Mind*, 59, (pp. 433-460). 1950.

[16] Jeanette Wing. “Computational thinking”. *ACM Communications*, 49, (pp. 33-35). 2006.

[17] Ludwig Wittgenstein. “Tractatus logico-philosophicus”. Routledge, 1921/1974.

[18] Ludwig Wittgenstein. “Philosophical Iinvestigations”. Basil Blackwell. 1958.

[19] Ludwig Wittgenstein. “Blue and Brown Books”. Basil Blackwell, 1958.

[20] Jia You. “Beyond the Turing Test”. *Science*, 347, (pp. 116-116). 2015.

High-Density Pattern-Of-Life Modeling

Randolph M. Jones[1], J. T. Folsom-Kovarik[1], Pat McLaughlin[2], and Rich Frederiksen[1]
[1]Soar Technology, 3600 Green Court, Suite 600, Ann Arbor, MI, 48105 USA
[2]Scientific Systems Company, 24632 Monte Royale, Laguna Hills, CA, 92653 USA
rjones@soartech.com, jeremiah.folsom-kovarik@soartech.com, Pat.McLaughlin@ssci.com, rdf@soartech.com

Abstract - *Patterns of Life are general human behavioral patterns that emerge from individual behaviors and reflect group tendencies, especially sociocultural tendencies. Challenges for modeling patterns of life are somewhat different from challenges for traditional modeling of cognition, decision-making, and expertise. Modeling patterns of life requires solutions for background agents, which are not individually complex, but are individually unique and representative of rich sociocultural backgrounds; as well as solutions for foreground agents, which are unique and complex, but also blend in with the background behavior patterns. This paper presents a POL modeling architecture that addresses issues in authoring and executing such large populations of unique individuals.*

Keywords: Patterns of life, Large-scale behavior modeling, Automated scenario authoring

1 Patterns of life

We have pursued several projects that involve computational modeling of human *Patterns of Life* (POL). Schatz et al. [1] define POL as follows: “In the context of cultural training, patterns of life are the archetypal emergent properties of a complex sociocultural system.” In addition, “…the emergent properties…can be organized into categories that describe classes of patterns of life that share similar general features, and we expect these same classes will manifest (likely with different nuanced characteristics) across all societies and cultures.” Folsom-Kovarik et al. [2] proposed scalable POL models as a challenge problem for applied artificial intelligence. This paper describes an architecture and approach to respond to that challenge.

POL are properties of a society that determine how individuals function on a day-to-day basis. The US military uses POL for Stability, Security, Transition, and Reconstruction (SSTR) operations, to train US Forces who must operate in these communities, and to identify threats based on POL anomalies. The DARPA Insight program sponsored research into the structure of generative POL models to support a simulation and test environment for threat recognition algorithms. Typical applied and theoretical work in artificial intelligence focuses on high-level competence or intelligent decision-making of individuals. In contrast, POL modeling focuses on social, cultural, and everyday behavior. POL are themes that appear in a population’s everyday actions and interactions. The ability to understand patterns of life, and thus to represent and instantiate POL models, is a critical emerging military requirement.

Distribution Statement A: Approved for Public Release, Distribution Unlimited. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

This paper describes technical issues of POL modeling and details of the approach we have developed across many projects, beginning with the sophisticated Insight simulation. POL modeling requires solutions for breadth, efficiency, and authoring of agent behaviors. We use cultural representations that explicitly allow for natural human variation, and a computational infrastructure that generates a continuum of relevant socio-cultural behaviors in two classes: efficient schedule-based scripts that support high-density populations and knowledge-rich individual models with the competence to adapt decisions based on goals and situational understanding.

2 Application areas

The Department of Defense (DOD) recognizes that POL analysis is an excellent mechanism for threat identification and engaging with a population in appropriate socio-cultural ways [3]. Those who live in or analyze foreign areas learn to recognize anomalies in typical patterns and react accordingly. The training challenge is to prepare those who are not in the area or do not have time to slowly become familiar with these threats. Consider an example where a U.S. Marine Corps squad leader is operating near a town of about 10,000 people, with reports of insurgent activity in the area. On a given morning, the marketplace includes a small number of people milling around (perhaps a dozen or so), a few bicycles and one car. Some stalls in the marketplace are empty. Is this a clear indication of danger or normal for this particular time?

Understanding the local POL is essential for the squad leader to build an overall assessment of this situation, determine whether any observed elements are unusual or dangerous, and predict what is likely to happen next. The squad leader’s observations must cohere into high-level patterns that make sense. In this example, perhaps the traffic is typical for this time of the day and should increase dramatically later. This is just one example of how POL analysis informs modern warfighting and peacekeeping. Modeling POL can support scenarios with increasing levels of behavior fidelity depending on the simulation need.

Realistic background context in training. POL can provide a realistically rich background for focused training activities. This can improve training simply by increasing user presence or putting the training into a realistic context, even if the POL itself is not a key part of the training.

Training of anomaly identification. For other types of training, POL is a key focus. A trainee may be learning to distinguish significant, anomalous behavior from the normal patterns in an area. In this case, *background* POL models include routine behaviors, while *foreground* models generate patterns that are similar, but characteristic of significant anomalous threats. A trainee must pick out the anomalous behavior patterns from the background.

Socio-cultural sense making. For analysis and situational understanding, POL models can define high-level, latent behavior clusters that explain several low-level, observable behaviors. For example, a crowd meeting at a particular location every weekday morning might represent a high-level behavior of going to a bus stop to get to school. In such an application, an SSTR planner might use this information to decide where to locate schools or build roads. A logistics planner might use these patterns to decide when and how to move people and goods through the space with minimal risk.

3 Requirements for patterns of life

The problem of simulating POLs is to provide the simulated environment with realistic *foreground* and *background* representations of the human terrain. Realistic *foreground* models require groups of entities to be intelligent enough to appear realistic under scrutiny. Threatening insurgent behavior often comes from groups who blend in with normal patterns most of the time, but step out carefully as required to achieve goals. Foreground models must be aware of their surroundings, react to local dynamics and proximal entities, and blend in with the background POL models, so they cannot be simple scripts. Without sufficient fidelity, foreground agents stand out from the background population, making them easy to spot and useless for training or planning. Realistic *background* models must generate sufficiently broad behavior patterns that accurately reflect the agents' sociocultural milieu. The models must also be *efficient to execute* and *easy to author*. Individually authoring entity behaviors is intractable for large populations, but random generation produces unrealistic behaviors, too easy for a human learner to detect.

The overall POL of a population describes the actions and interactions of many individuals taken as a whole. The patterns emerge from aggregated individual actions. An observer can recognize general trends that characterize the *group*, even if they are not true for any one *individual*. Individual decisions and actions that comprise group behavior arise from underlying facts and narratives pertaining to the sociocultural situation. These provide a causal context that an observer can use to infer latent meanings from relatively few visible actions. Together, the emergent group behaviors and the background facts and narratives make up the POL for a particular region and population.

There is a unique collection of technical challenges to modeling POL. The major focus must be on routine behaviors rather than complex decision making. However, the behavior must be appropriate to situational context, so the agents must have some level of situation-understanding competence. Similarly, in order for behaviors to adapt appropriately to context, they cannot simply be scripts that break down if the assumptions behind the scripts become violated. The range of behaviors must be sufficiently broad and varied to provide a rich population-level dynamic in which a large variety of group-level patterns are observable and "normal", but that there are not just a large number of agents doing exactly the same things. The behaviors must also be appropriately configurable to represent sub-populations, such as different cultural groups, economic and social roles, occupation types, social strata, etc. The simulation must be computationally efficient enough to generate hundreds of thousands of background agents, as well as perhaps dozens of more sophisticated (and presumably more expensive) foreground agents. Finally, the technical solution cannot reasonably require specific programming of each background agent. There must be ways to configure the behaviors using population-level demographic parameters.

4 POL modeling architecture

We have designed and partially implemented a POL modeling architecture that specifically addresses these technical challenges. The architecture allows us to build models of the activities of many individuals in a scalable way. It includes a library of goals and behavior patterns that generate a wide variety of behaviors that instantiate group-level patterns. It also provides mechanisms to minimize user input for configuring large populations of agents. Figure 1 displays the components and functions of the POL architecture. The architecture addresses two primary modes of operation: on the left, a scenario author carries out *scenario generation*. On the right, an operator takes the generated scenario and carries out *scenario execution*. In many situations, the author and operator are a single user. Generation and execution both rely on a POL knowledge representation that supports background and foreground agents. Background agents are highly scheduled, with configuration mostly or entirely determined by population-level parameters. Foreground agents generally have minimal scheduling, make their own decisions during execution, and have more detailed and unique goals and activities, rather than population-level parameters. However, in our design, any

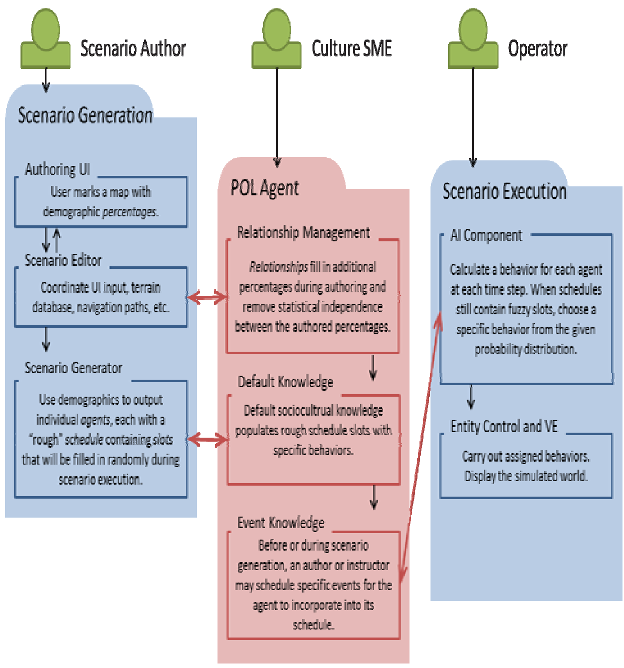


Figure 1. Schematic and functions of the POL architecture.

POL agent can be authored with a mix of population-level and detailed parameters, as well as more or less scheduling of specific activities. This enables rapid construction of the essential details and automated variation of the non-essential backgrounds. Because the background and foreground behaviors use the same sets of primitive actions and overlapping intermediate goals, the architecture supports the generation of foreground behaviors that blend appropriately with background behaviors.

5 POL knowledge representation

The knowledge representation includes primitive actions, goals, schedules, and configuration parameters. For this architecture, it is important to note that agent reasoning can take place during scenario authoring *and* execution. The output of the authoring process is a set of goals incorporated into a schedule. The agents know how to achieve the goals using the available primitive actions. The schedule may be complete or only partial. Background agent generation creates for each agent a rough schedule with some actions determined and some *slots*, where actions will be chosen from a reasonable range in order to create variation. Foreground agents have run-time knowledge for scheduling their own goals and actions. Configuration parameters influence the agents' choices of goals and the timing of the actions. In general, configuration parameters for background agents are used to generate choices during scenario authoring, while foreground agents may make some or most of their choices by reasoning about goals and parameters during scenario execution. The rest of this section provides more detail on each of these knowledge components.

Primitive actions. These define the interface between the agents and the simulation. Primitive actions generally provide the observable actions that a POL agent will generate, including movement, entering or exiting buildings and vehicles, using devices (e.g., to communicate), and interacting with other agents. Primitive actions can also be unobservable, such as a change in beliefs or achievement of a goal. In general, the primitive actions used by background agents are the same (or at least a large subset) of the primitive actions used by the foreground agents. This is because at a surface level there must be a possibility of confusing observable background and foreground behavior, so there should be nothing (or very little) observable that is unique to either type of agent.

Goal hierarchies. An important property of the POL architecture is that schedules are not merely timed scripts of primitive actions. Rather, they are composed of goals that can be decomposed into actions. The use of goals allows the agents to adapt their actions to changes in the scenario. As a simple example, if the schedules were composed only of actions, a schedule item might be something like "Starting at noon, drive Car X to Location Y". This is a specific and brittle command, which could fail if any assumptions behind the command become violated during scenario execution. For example, what happens if Car X becomes unavailable, or the agent is not able to get to Car X before noon? To address such possibilities, the POL schedules refer to goals, such as "As soon as possible before lunch time, drive to a nearby restaurant, with Car X being the preferred vehicle and Location Y being the preferred restaurant." This specification of schedules in terms of goals and preferences allows each agent to adapt its behavior when assumptions break down.

POL agents also organize their behavior around goals that can contain both scheduled and unscheduled actions. The primary difference between background and foreground agents is that the goals for background agents are generally at a level of abstraction closer to the primitive actions, while goals for foreground agents are more abstract. Both types of agents have goals centered on group activities that are fluid and contain unscheduled elements. Therefore, they can even carry out complex tasks such as planning a covert attack.

Schedules. Agents achieve goals based on schedules. Each agent has a schedule that tells it where and when to achieve its goals. A schedule covers a day, and agents can have more than one schedule to govern different types of days such as weekdays and weekends. Variation between daily schedules is accomplished with probability distributions over action alternatives. This is implemented via efficient fuzzy state machines, with extensions to improve expressivity, adaptivity, look-ahead reasoning, and behavior authoring. For example, a schedule might specify a 10% chance of going out to lunch and 90% chance of staying in the office. As another example, a *leave_for_work* goal's start time may vary about a 7:00AM center point. Each schedule items dictates a goal to achieve, with some indication of the time to start pursuing or to aim to achieve the goal (or some other temporal relationship). Depending on the situation and the parameters, there is room for variation in goal achievement. POL agents have the knowledge to refine goals into primitive actions. Simpler goals that generally populate the schedules of background agents have simple achievement actions. Additionally, the schedules include fuzzy adaptation parameters to reflect natural human variation. Without this, every employee of an office might stand up to leave at exactly the same time or use the same route to get to work. This approach enables us to simulate thousands of diverse background agents on a single machine.

Parameters. Depending on the type of goal and/or how the goal will be achieved, there can be additional parameters that need to be specified for each instance of a schedule goal. Example parameters include:

- Location of goal achievement
- Start time to pursue the goal
- End time to achieve the goal
- Type of goal
- Transportation plan to get to location, including
  - Sequence of waypoints to achieve
  - Transportation mode to use for waypoint sequences
- Equipment to be used to achieve the goal
- Target agent for communication or interaction goals

Foreground agents do not require values for all of their goal parameters. The scenario author and operator may leave the decision up to the agent to instantiate the parameters. Depending on the parameter type, the scenario, and the run-time situation, the agent may make the decision at authoring or execution time. In addition, the goals for foreground agents can be more abstract and complex, meaning they might be refined into other goals, rather than being refined directly to primitive actions.

In addition to the goal parameters specified above, each agent can have associated population and role parameters that are used at scenario authoring time to assist in automatically generating goals and schedules. For example, the architecture provides the ability to generate schedules for all agents in a region based on percentages the author assigns to that region. The author might specify that a certain percentage of the population in a region adhere to a particular religion, are employed, are married, own cars, etc. The generated schedules follow rules to keep them internally consistent. For example, agents are not scheduled to work far from their homes if they have no transportation to get them to work.

POL agents can also form groups and act together when all members of a group except one have schedules that specify they should follow the movements directed by one other agent. POL agents have social network links that specify relationships, and they use these links to form groups when appropriate. For example, when an agent goes to work on a weekday it goes alone, but when it takes the family to a park on the weekend, the entire family pursues the goal together. Configuration parameters are not only associated with goals and agents. They are also associated with the geography and regional infrastructure. This allows the scenario-authoring component to associate authoring choices together based on mixes of parameters. The next section presents examples of how parameters drive scenario-authoring decisions.

6 POL authoring

When discussing behavior models for thousands, or possibly hundreds of thousands, of agents, it is infeasible to take an approach that requires detailed specification of each individual agent. Thus, it is necessary for a POL architecture to support population-level specification of scenario authoring choices. In our architecture, scenario authoring consists of three basic activities:

- Geography and infrastructure. This involves setting up the simulated infrastructure of the scenario, including location and functions of buildings, as well as available objects (such as vehicles or devices).

- Background agent specification. This involves specifying population-level parameters that are used automatically to generate schedules for the individual background agents.

- Foreground agent specification. This is a more interactive process of creating more complex goals, schedules, and decision points for foreground agents. Still, the author is not required to script every action for any character. Also, foreground agent behaviors can be derived from templates for foreground agents that have similar goals.

Scenario authoring begins by creating a geographical terrain with specified building locations and functions. Buildings can be residential, commercial, industrial, religious, educational, or governmental. Next, the scenario author specifies demographic parameters for regions of the terrain. For example, an author can outline a neighborhood and enter parameters that tell how many of the people generated in that region are employed, how many own a car, how many belong to a particular religion, etc.

Once these regions have been defined, the authoring component automatically generates population individuals, characteristics, and relationships. For residential buildings, the demographic parameters are used to generate populations that occupy each building (using parameters such as average family size, together with sex and age distributions). Demographic parameters are also used to determine ownership of goods (such as vehicles and devices) for each agent. For non-residential buildings, the POL authoring component generates job slots for each building, based on building size and job-related parameters for the region. Using employment parameters, the authoring component stochastically assigns available job slots to "adult" members of the population. It is possible to bias employment assignments using factors such as employment rate, ethnicity, religion, residential location, vehicle ownership, etc.

Once the basic background population has been generated, organized into families (based on residence), and assigned jobs, additional demographic parameters are used to compute additional types of relationships between individuals. These relationships include:

- Non-immediate family connections
- Religious connections
- Political connections
- Miscellaneous "interest group" connections
- Social connections
- Educational connections
- Other economic connections

These simulated relationships and group memberships are then used to generate schedules for the background agents. Without the relations, the system would produce a background population that exhibits the specified population statistics at random. However, demographic facts in reality have dependencies between categories that make some characteristics predict others. In order to reflect these patterns, the POL system applies causal and correlative information to eliminate unwarranted independence and to introduce underlying causality. For example, if it is known that employment in an urban neighborhood is correlated with education level, the authoring component can use that knowledge to fill in blanks in the authored percentages and to impose additional patterns that mere percentages do not capture. The output of the authoring component is a background population bearing realistic simulated demographics that produce rational behavior patterns.

In addition, cultural baseline knowledge helps transform population demographics into agents with plans and goals. This process converts otherwise random assignments into population-level patterns. Agent generation creates rough schedules with some actions determined and some *slots*, where actions will be chosen from a reasonable range in order to allow variation. For the scenario author, using the POL architecture's baseline knowledge reduces or eliminates the need to hand-code any portion of the background agent schedules. This background knowledge imposes consistency and causality on the schedules. For example, in some cultures, it is rare for an individual to fail to attend religious ceremonies at least once a week. Without cultural knowledge, the choice to skip attendance would be random and carry no meaning for a human to understand. Cultural knowledge adds causation to the event. It changes the output of the random generator so the agent more closely follows the norm for its assigned culture, religion, ethnicity, or other relationship. Alternatively, it might add parameters labeling some population members as religiously unobservant, a fact that would lead to further constraints on agent schedules.

6.1 Foreground agent authoring

The basic characteristics of foreground agents can be defined in the same ways that background agents are configured. However, because foreground agents engage in more complex and specialized behaviors, the POL authoring component contains additional tools for building their more complicated schedules. Figure 2 depicts the general framework for representing foreground agent knowledge and generating foreground agent schedules. Foreground agents are implemented within the Soar cognitive architecture [4][5] to allow them to reason about their environment and adjust their goal-directed behavior accordingly. A comprehensive *long-term knowledge specification* contains the goals, tasks, conditional logic, and goal decompositions available to the foreground agents. An author runs the *scenario/behavior planning user interface* to configure foreground agents. This interface uses the long-term knowledge specification to assist in creating a behavior schedule for the foreground agents. For example, if the user specifies that the scenario should include agents with the goal "achieve-covert-attack", the user interface will automatically decompose that mission into a supporting set of subgoals, primitive actions, and parameter slots. The user interface will allow the user to examine the mission structure and fill parameters, including scheduling times to achieve goals. As the user fills parameters, the user interface performs further task decompositions and displays them to the user. This also identifies additional parameters that the user can fill. The user has the option of filling each parameter or allowing the agent to fill it during scenario execution. Ultimately, the output of the planning process is a partial schedule of goals for the foreground agent to achieve, with unspecified parameters left as choice points for the agent during execution. In the extreme, but atypical, case where the user selects values for *all* parameters in the schedule, the output is a scripted schedule similar to the background agents. However, the schedule includes a full goal hierarchy, which increases adaptivity and traceability in dynamic situations.

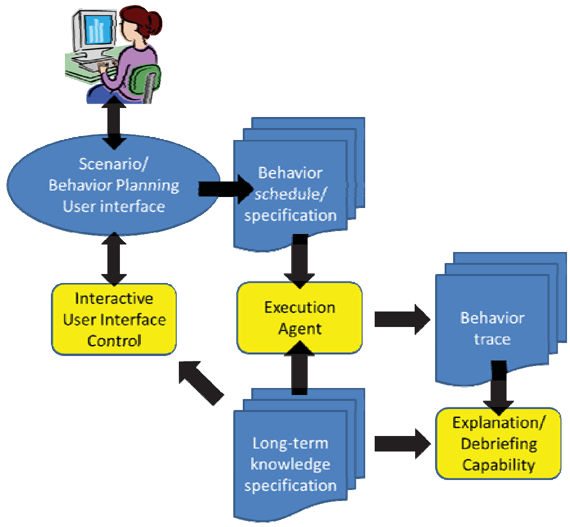


Figure 2. Multiple interpretations of HVI knowledge and behavior specifications.

7 POL agent execution

At run time, the *execution agent* follows the goals and actions dictated in each behavior schedule. For background agents, this typically executes a schedule of low-level goals. However, even low-level goals include adaptive constraints, such as pedestrians yielding to moving vehicles. These execute with high computational efficiency, because they do not involve any sophisticated sensing, understanding, or intelligent decision making. Rather, they require only efficient primitive functions for activities like route planning and schedule estimation. Foreground agents usually include schedule elements with unspecified parameter values or goal decompositions. The execution agent uses its decision logic and long-term knowledge specification to make choices, with the choices being sensitive to the current state of the execution environment. This allows foreground behavior that is not scripted, but is adaptive to the current situation. The execution agent fills in any details necessary to execute the mission. This type of adaptive, intelligent reasoning and situation understanding requires more computational resources than the simple schedules executed by background agents. Thus, we expect a scenario to include on the order of dozens of foreground agents.

After execution, an *explanation/debriefing* interface can examine population behavior patterns and apply long-term knowledge to generate explanations for the parameters, goals, and situational elements that contributed to decisions and actions generated by the execution agent. The interface uses the background knowledge base to reflect, after the fact, on the decisions that a foreground agent made during execution time [6]. This improves scenario creation and evaluation, by supporting traceability of the behavior to knowledge.

8 Current status

The POL architecture continues development under several DOD-funded projects, together with internal research and development efforts. We implemented the initial prototype under a DOD training program and DARPA Insight. Figure 3 shows a demonstration simulation environment running the initial POL prototype implementation. The prototype implemented background and foreground agents. The background engine interpreted schedules, translating goals into primitive actions using functions for path planning and action sequencing. It modeled several thousand agents per CPU core. The foreground engine used the Soar cognitive architecture [4] to implement situation understanding, hierarchical goal decomposition, and parameter-value selection at domain-specific choice points. We are extending this initial prototype through a collection of active projects, with specific recent focus on authoring.

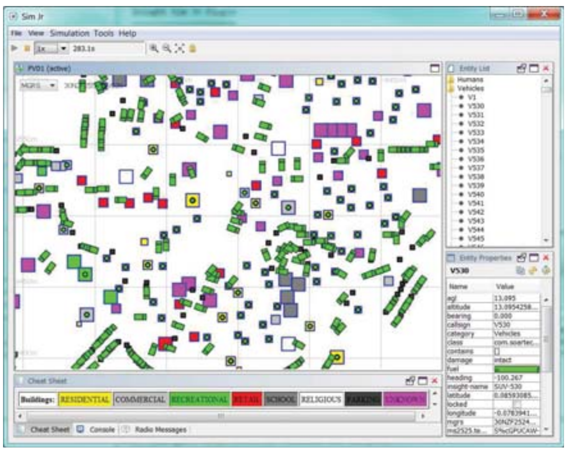


Figure 3. A sample pattern-of-life simulation.

Currently, users author the scenarios, with a user developing background and foreground schedules with population parameters used to generate some of the schedule goals, timings, and parameters. The full capabilities for relationship management and interactive behavior planning are still in the design phase. We have evaluated the initial prototype on a set of use cases in a schoolhouse environment, demonstrating the ability to deploy large numbers of background agents, together with foreground agents that perform team-based insurgent activities. Background and foreground agents blend in the simulation environment. We have demonstrated that demographic parameters allow us to configure up to 320,000 background agents. We have run example scenarios populated with 20,000 background agents per CPU core, plus small numbers of foreground agents that blend in.

9 Related work

The POL architecture we have presented addresses the unique collection of requirements and applications described at the beginning of this paper. The design and implementation of the architecture has benefited from lessons learned from other research efforts that address portions of the POL problem.

9.1 Crowd modeling

Crowd modeling portrays realistic movements for low-fidelity, identical individuals in relatively small spaces [7]. Typical crowd models implement thousands of individuals moving in areas up to the size of a football stadium. Academic models, for example, analyze the best way to design exits to quickly empty an auditorium. The individuals have concepts of personal space and other data that makes small-area movement realistic, but they are interchangeable in that they do not have individualized choices. All agents exhibit essentially the same behavior with small variations, such as attempting to use different doors or different loitering areas. Thus, looking carefully at individuals in these crowds reveals they are merely walking around random points and not following any latent patterns. They do not possess reasons for their choices or actions, and they do not implement demographically based constraints and decision logic.

9.2 Social network modeling

Social network modeling describes groups of individuals that differ in their properties and relationships. These models represent relationships between individuals as connection graphs [8][9]. The networks are often randomized to create a particular overall graph structure. These models, may have many different individuals, but are often valid only in the aggregate. For example, such models in the academic world have yielded results about what mechanisms for making and breaking links are typical in the network as a whole. These models focus on social dynamics, but typically ignore POL dynamics related to the physical world.

Random graph generation implies a lack of realism when observed closely. As an example, in a real-world social network, advertisers have recently begun to value having people "like" their products on Facebook. This has led to unscrupulous providers creating fake individuals and selling their "likes." However, random generation does not follow real patterns of life, so it is possible to detect these fake individuals – one product might be liked almost entirely by people in a single foreign city, and another might be liked by one person in each city, with no overlap. Both of these mistakes reflect a failure to model the POLs that should geographically cluster individuals with remote, second- and third-order similarities and interpersonal connections.

9.3 Abstract population models

Abstract population models simulate large groups with low detail. The population may include a few powerful individuals, or no individuals and only groups. For example, one such simulation models political, military, and other kinds of power as stocks and flows [10]. Some abstract population models include goals and beliefs associated with each stock that direct the flows to achieve goals, but most models do not. Abstract population models in general focus on group relationships and influences, and they do not model individual behaviors that would fit into an overall pattern of life.

10 Conclusions and future work

POL modeling differs from typical AI modeling in its focus on breadth and density of behavior, as well as the importance of agent behaviors organizing into appropriate high-level patterns. POL requires solutions that emphasize efficiency in behavior authoring as well as execution, and that model a continuum that blends background and foreground behaviors. Our POL architecture design addresses these requirement and currently implements components sufficiently to demonstrate efficient generation of high-density background populations that represent varied cultural demographics and provide a suitable context for sophisticated foreground agents to blend into.

An important next step is automatic generation of POL based on interaction with standard representations of a population. The vision is to ingest a geographical and statistical profile, available from open-source intelligence reports, and combine this with socio-cultural representations to generate background agents automatically. Once achieved, whether for training or for planning, the system will ingest just-in-time information and support high-fidelity simulations of scenarios for direct relevance to ongoing missions. A second consideration is modeling communications that affect POL. Communications have significant impact on behavior. Locals flee when they hear police are coming. Crowds form when they learn of a controversial rally through twitter. These drivers of POL deviations make a difference in practice, and we plan to implement authoring and run-time representations for these effects in the next iterations of our work.

11 References

[1] Schatz, S., Folsom-Kovarik, J. T., Bartlett, K., Wray, R. E., & Solina, D. 2012. Archteypal patterns of life for military training simulations. *Proceedings of I/ITSEC.* Orlando, FL.

[2] Folsom-Kovarik, J. T., Schatz, S., Jones, R. M., Bartlett, K., & Wray, R. E. (2014). AI challenge problem: Scalable models for patterns of life. *AI Magazine 35*(1), 10–14.

[3] Carpenter, L. W. 2012. Emerald Warrior 2012: Joint, coalition exercise offers special operations forces irregular warfare training. *Airman.*

[4] Laird, J. E. 2012. *The Soar cognitive architecture.* MIT Press.

[5] Wray, R. E., & Jones, R. M. 2005. An introduction to Soar as an agent architecture. In R. Sun (Ed.), *Cognition and multi-agent interaction: From cognitive modeling to social simulation,* 53–78. Cambridge, UK: Cambridge University Press.

[6] Taylor, G., Jones, R. M., Goldstein, M., Frederiksen, R., & Wray, R. E. 2002. VISTA: A generic toolkit for visualizing agent behavior. *Proceedings of the Eleventh Conference on Computer Generated Forces and Behavior Representation.* Orlando, FL.

[7] Helbing, D. and Molnar, P. 1995. Social force model for pedestrian dynamics. *Physical review E, 51*(5):4282. Hughes, R. 2002. A continuum theory for the flow of pedestrians. *Transportation Research Part B: Methodological, 36*(6):507-535.

[8] McPherson, M., Smith-Lovin, L., & Cook, J. M. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 415-444.

[9] Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P., & Bhattacharjee, B. 2007. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (pp. 29-42). ACM.

[10] Taylor, G., Bechtel, R., Knudsen, K., Waltz, E., & White, J. 2008. PSTK: A toolkit for modeling dynamic power structures. *Proceedings of the Seventeenth Conference on Behavior Representation in Modeling and Simulation. Providence*. Providence, RI.

Verification of the position estimation method of the smartphone by using visible light communication and its application to Intelligent Lighting System

Kohei Yamaguchi[2], Mitsunori Miki[1], Sho Kuwajima[2], Ryohei Jonan[2], and Hiroto Aida[1]
[1]Department of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *The Intelligent Lighting System controls individual lighting fixtures in accordance with illuminance information obtained from the illuminance sensor assigned to each worker and provides preferred illuminance to each worker. We established a method to realize the Intelligent Lighting System by using an illuminance sensor embedded in a smartphone instead of a standalone illuminance sensor. The Intelligent Lighting System turns on each lighting fixture in the optimal lighting pattern that minimizes power consumption by estimating the positional relationship between illuminance sensors and lighting fixtures. This study proposes a method for estimating the location of a smartphone by visible light communication. The proposed method enabled completing location estimation in 2 seconds regardless of the number of smartphones and obtaining the optimal lighting pattern in a shorter time than before.*

Keywords: Lighting Control, Illuminance, Smartphone, Position Estimation

1. Introduction

In recent years, as the improvement of energy efficiency has become a topic of broad discussion, there has been a drive toward energy saving in office buildings. Since lighting accounts for about 20% of the total power consumption in office buildings, improving the lighting environment can bring about a significant power saving and thus a great contribution to energy conservation. Against this backdrop, the authors proposed an Intelligent Lighting System which individually realizes illuminance levels required by individual workers while saving power consumption[1].An Intelligent Lighting System is a system to realize required illuminance at the position where an illuminance sensor is installed with minimum power consumption. We established a method to realize the Intelligent Lighting System by using an illuminance sensor embedded in a smartphone instead of a standalone illuminance sensor[2]. Since smartphones have a built-in illuminance sensor for screen brightness control, they may be utilized as illuminance sensors for Intelligent Lighting System. Not only that they may function as illuminance sensors, the use of commercial general purpose products will also increase the ease of maintenance while reducing initial system construction cost; they may also provide a touch-panel user interface.

When the Intelligent Lighting System controls individual lighting fixtures, it needs to determine the effect of each lighting fixture on each illuminance sensor (referred to as "illuminance/luminance influence factor") in order to realize an efficient lighting pattern. As a method to determine this illuminance/luminance influence factor, the Intelligent Lighting System using smartphones estimates the positional relationship between lighting fixtures and smartphones[2]. On the basis of the estimated positional relationship, the Intelligent Lighting System preferentially brightens lighting closer to smartphones, namely, those with a greater illuminance/luminance influence factor, in increasing lighting and preferentially dims lighting further from smartphones, namely, those with less illuminance/luminance influence factor in decreasing lighting. An efficient lighting pattern that curbs power consumption is realized by such lighting control. In our preceding study, we reported a method for estimating smartphone's position using binary search[2]. With the binary search method, however, the number of lighting control steps required for search increases as the number of smartphones searched increases. Therefore, it is considered that a large amount of time is required for search in an environment with many smartphones. Thus, this study proposes a position estimation method using visible light communication as one whereby search times do not depend on the number of smartphones and verifies the Intelligent Lighting System in which this method is implemented.

Many studies have reported on visible light communication[3][4]. Visible light communication refers to wireless data communication using light visible to human eyes (visible light) as transmission media. It is drawing attention not only as a conventional data communication method but also as a new position estimation method in an indoor environment where position estimation by GPS, etc. is difficult. On the other hand, since an illuminance sensor is built in a smartphone, visible light communication using a smartphone as a receiver is considered to be possible. The performance of smartphone built-in illuminance sensors differ, however, by smartphone models, which is considered to result in difference in the speed and range of visible

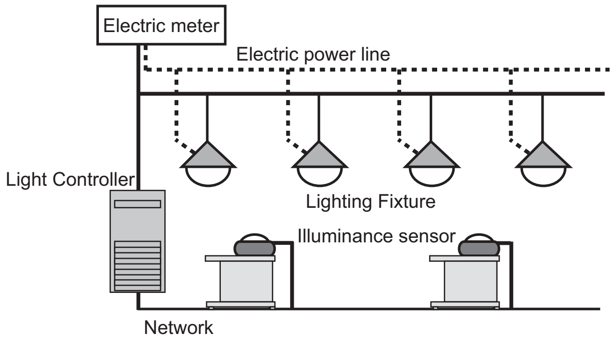


Figure 1: The construction of an Intelligent Lighting System

light communication by smartphones. This study thus examines, first of all, the response to luminance changes and communication range of the built-in illuminance sensor of different smartphone models. We then show the effectiveness of the proposed method by demonstrating that the convergence of illuminance to the illuminance preferred by a worker is possible by conducting position estimation by visible light communication.

2. Intelligent Lighting System

2.1 Configuration of Intelligent Lighting System

Figure 1 shows the construction of an Intelligent Lighting System. An Intelligent Lighting System is a system to realize required illuminance at the position where an illuminance sensor is installed with minimum power consumption[1]. As shown in Fig.1, it consists of lighting fixtures, control devices, illuminance sensors, a power meter and a network to connect them.

The control device installed on each lighting fixture changes luminance within the range undetected by a worker using an optimization method on the basis of illuminance and power consumption information. By repeating this, illuminance required by a worker is realized with power saving.

Using an algorithm based on simulated annealing, the Intelligent Lighting System solves, for each lighting fixture, an optimization problem with the luminance of each lighting fixture as the design variable, the target illuminance of each illuminance sensor as the constraint, and the power consumption under the total illuminance as the objective function in an autonomously distributed manner. Namely, the luminance of each lighting fixture is randomly changed within the range undetected by a worker for each search to search for the optimal lighting pattern. An illuminance/luminance influence factor is set for each lighting fixture in accordance with its positional relationship with each illuminance sensor, and a random change in luminance is made to have directionality in accordance with the degree of the effect.

One time search constitute a single step, and lighting is controlled by repeating the step every 2 seconds. The objective function of each function is represented by Formula (1).

$$f_i = P + \omega \times \sum_{j=1}^{n} g_{ij} \quad (1)$$

$$g_{ij} = \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ R_{ij} \times (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{cases}$$

$$R_{ij} = \begin{cases} r_{ij} & r_{ij} \geq T \\ 0 & r_{ij} < T \end{cases}$$

i: Number of lightings, j: Number of sensors
P: power consumption[W], ω: weight[W/lx]
Ic: current illuminance[lx], It:target illuminance[lx]
r: illuminance/luminance influence factor（regression coefficient）, T:threshold value

The objective function represented by Formula (1) is composed of power consumption P and illuminance constraint g and calculated for each lighting fixture. Penalty g whose constraint is the target illuminance of each illuminance sensor changes by the illuminance/luminance influence factor. The system functions in such a manner that only a lighting fixture with a large illuminance/luminance influence factor is significantly affected by penalty. In addition, by setting the threshold T to illuminance/luminance influence factor r, lighting fixtures affecting a given illuminance sensor can be narrowed down to those in its neighborhood. This causes lighting fixtures far from the illuminance sensor to be controlled to minimize power consumption.

2.2 Intelligent Lighting System using smartphones

It is conceivable to substitute illuminance sensors of the Intelligent Lighting Systems with smartphones, which have widely spread in recent years. Using smartphones as illuminance sensors has advantages such as reducing cost for installing dedicated sensors, improving serviceability by using generic parts, etc. An illuminance sensor for adjusting screen brightness is embedded in a smartphone, and it is possible to obtain illuminance by using it. Our previous study reported the effectiveness of the Intelligent Lighting System using smartphones as illuminance sensors by obtaining illuminance by means of smartphone built-in illuminance sensors[2].

3. Performance verification of smartphones

3.1 Response performance

Smartphones have an illuminance sensor built in for adjusting screen brightness, whose performance is considered

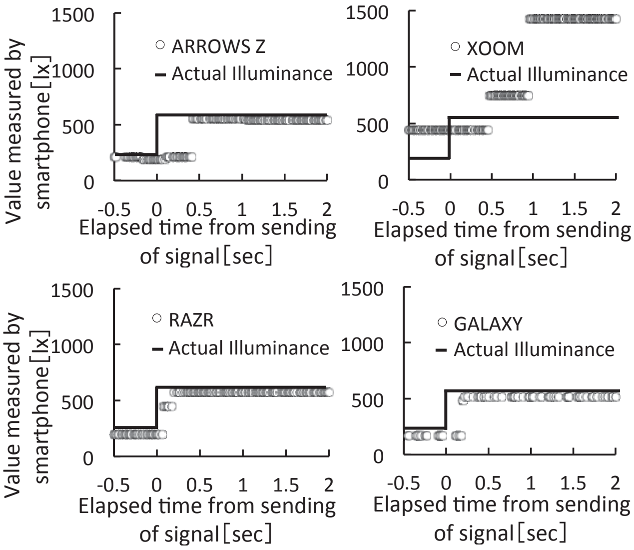


Figure 2: The history of illuminance given by smartphones

to vary by smartphone models.

We examined the response performance of smartphone built-in illuminance sensors in order to determine data communication speed in performing visible light communication using smartphones. Built-in illuminance sensors do not obtain a correct illuminance value immediately in response to a change in luminance of the lighting fixtures, and it takes some time for illuminance values to converge to the correct one. In visible light communication, time required for sending data of 1 bit is considered as time after a lighting fixture's luminance changes and by the time when a built-in illuminance sensor manages to measure the change correctly. Thus we conducted a verification experiment to measure time after a lighting fixture's luminance changes and by the time a built-in illuminance sensor measures the change correctly.

Smartphones ARROWS Z, made by Fujitsu, XOOM and RAZR, made by Company Motorola, and GALAXY, made by Company Samsung were used. The LED lighting system by Company Sharp was used, which can be controlled in 1,000 stages. The lighting fixture was turned on at constant luminance in advance, and the illuminance on the top of a desk directly under the lighting fixture which was 70 cm high above the floor was then measured with smartphones. The lighting luminance of the lighting fixture was subsequently increased by a certain amount. Then we measured time from the point when the lighting luminance rose to the time when illuminance values given by smartphones converged to a constant value. Luminance was set to 30% of the maximum lighting luminance before it was raised and 90% of the maximum after it was raised.

In order to measure a change in illuminance sensor precisely, an illuminance sensor NaPiCa made by Company Panasonic, capable of obtaining illuminance at an interval of about 1 ms, was placed adjacent to a smartphone to measure illuminance simultaneously.

Figure 2 shows the history of illuminance given by ARROWS Z, XOOM, RAZR, and GALAXY. The vertical axis of the graph indicates the illuminance value, and the horizontal axis, time from the transmission of a light control signal. The real change in illuminance measured by the illuminance sensor NaPiCa is shown in a solid black line. With every model in Figure 2, the illuminance value shown by the illuminance sensor NaPiCa rose almost simultaneously with signal transmission. It is thus found that the lighting fixture's luminance changed to a constant value immediately after signal transmission. It is also found from these graphs that with ARROWS Z, XOOM, RAZR, and GALAXY, it required 0.4 seconds, 1 second, 0.2 seconds, and 0.2 seconds respectively for the illuminance value to converge to a constant value after signal transmission. In this way, it was confirmed that time until convergence to a constant value differs by smartphones measuring illuminance.

Table 1 shows the result of measurement. The column Model in Table 1 shows models examined. The column Time shows the response time of built-in illuminance sensors. The longest response time recorded in 10 times of measurement is shown for each sensor. Values in this column can be regarded as transmission time per bit in visible light communication. Let RAZR, which yielded the shortest response time among smartphones shown in Table 1 shall be referred to as "Smartphone1," and GALAXY, which yielded the second shortest response time, as "Smartphone2."

Figure 3 shows the history of illuminance recorded when an information bit string (referred to as "Lighting ID") of "0101" is transmitted by actually changing a lighting fixture's lighting luminance, with the illuminance value obtained by Smartphone2 placed directly under the lighting fixture. The solid line shows the actual change in the lighting fixture's illuminance. The transmission speed in this instance was set to about 5 bps based on the result in Table 3. The lighting fixture's lighting luminance corresponding to a Lighting ID was set to 100% of the current luminance when a Lighting ID is 1 and 90% when a Lighting ID is 0. Figure 3 tells that Smartphone2 managed to obtain the change in illuminance corresponding to signal value transmitted.

3.2 Distance which can engage in visible light communication

When visible light communication is performed between a smartphone and a lighting fixture, it is necessary to change

Table 1: The response time

Model	Time [sec]
ARROWS Z	0.64
XOOM	1.22
RAZR(Smartphone1)	0.15
GALAXY(Smartphone2)	0.19

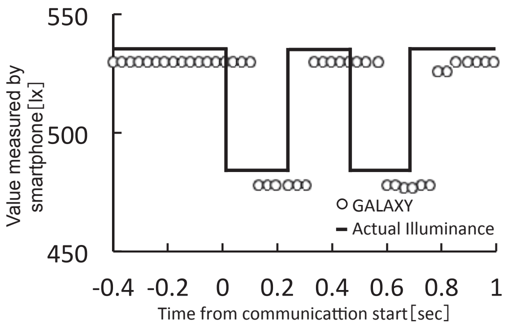


Figure 3: The history of illuminance when the Lighting ID of "0101" is transmitted

the lighting fixture's luminance within the extent that the change remains undetected by workers[5]. Since the range of an illuminance change which a VDT worker cannot detect is within 10% of the current illuminance, the luminance of a lighting fixture used in visible light communication in this study is changed by 10%. The distance between a lighting fixture and a smartphone that can detect the change in luminance of the lighting fixture is considered to differ by the smartphone's model. Thus, for each model, we measured the distance between a smartphone and a lighting fixture which can engage in visible light communication.

A lighting fixture was turned on at constant luminance and then the lighting luminance was raised by 10%. We measured the location of a smartphone which can detect the change in luminance was placed by measuring the horizontal distance from the point directly under the lighting fixture to the location of the smartphone. This distance can be regarded as the range within which visible light communication is possible. Table 2 shows the detection range of Smartphone1 and Smartphone2 when the lighting fixture was turned on at the maximum lighting luminance and minimum lighting luminance. The minimum lighting luminance was set to 30% of the maximum lighting luminance. Since lighting fixtures are generally placed at an interval of 1.8 m, based on Table 2, it is found that, even under minimum lighting luminance, where the possible range of visible light communication is the smallest, signals transmitted by multiple lighting fixtures interfere with each other.

Therefore, we conducted an experiment verifying the range in which a Lighting ID can be received in a case where multiple lighting fixtures transmit Lighting IDs which are different from one another. Different Lighting IDs were sent by two lighting fixtures A and B placed at an interval of 1.8 m, shown in Figure 4. "1010" was transmitted from the

Table 2: The detection range

Model	Distance(max luminance) [m]	Distance(min luminance) [m]
Smartphone1	1.3	1.1
Smartphone2	2.6	1.8

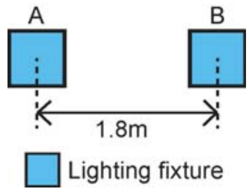


Figure 4: The preliminary experimental environment

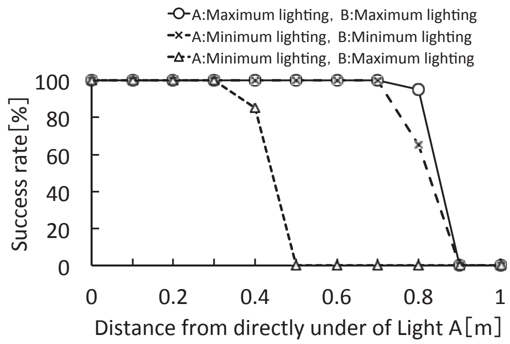


Figure 5: The rate of successful reception

lighting fixture A, and "0101," from the lighting fixture B. The lighting fixture's lighting luminance corresponding to a Lighting ID was set to 100% of the current luminance when a Lighting ID is 1 and 90% when a Lighting ID is 0. A smartphone was placed on the top of the desk along the line connecting the point directly under the lighting fixture A and the point directly under the lighting fixture B. The horizontal distance was measured between the point directly under the lighting fixture A and the point where the smartphone can detect the Lighting ID "1010." Figure 5 shows the rate of successful reception of the transmission of the Lighting ID in 20 attempts at each distance. The smartphone model used was Smartphone2, and the transmission speed was set to 5 bps. The result in Figure 5 tells that the distance at which the Lighting ID of the lighting fixture A can be received correctly at the probability of 100% is 0.7 m if both the lighting fixture A and the lighting fixture B are at the maximum or minimum lighting level and 0.3 m if the lighting fixture A is at the minimum lighting level with the lighting fixture B being at the maximum lighting level. Therefore, it was found that, even if the lighting luminance of a lighting fixture closest to a smartphone is minimal with the lighting luminance of other lighting fixtures being maximal, a correct Lighting ID can be received if a smartphone is placed within 0.3 m of the point directly under the nearest lighting fixture. The minimum lighting luminance was set to 30% of the maximum lighting luminance.

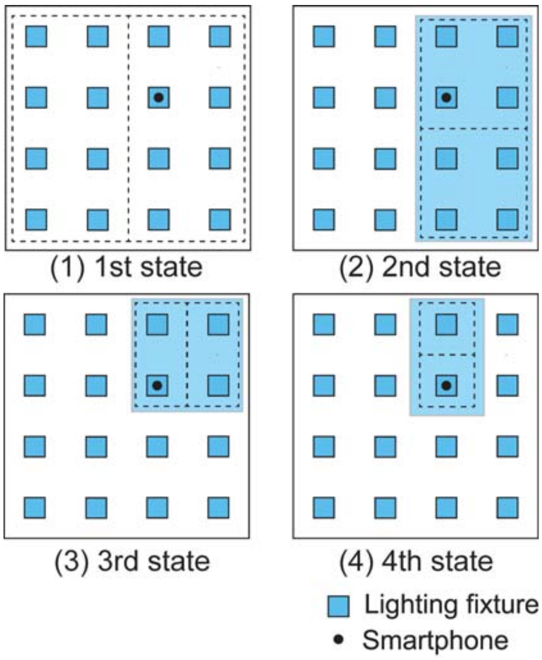


Figure 6: The concept of the binary seach method

4. A position estimation method of a smartphone using visible light communication

4.1 A conventional position estimation method

A position estimation method using binary search has conventionally been used in calculating an illuminance/luminance influence factor in the Intelligent Lighting System using smartphones. Conceptual diagrams for a binary search method are shown in Figure 6. Diagrams (1) through (4) in Figure 6 show an office environment in which 16 lighting fixtures were placed. First, lighting fixtures in the room are classified into 2 groups, as in (1) in Figure 6, and the luminance of lighting fixtures are changed uniformly in each group in such a range that workers do not perceive the change. Then, since a group closer to the smartphone causes a greater change in values obtained by a smartphone built-in illuminance sensor, it can be determined which group is closer to the smartphone by comparing the amount of change by groups. By recursively searching a group determined to be closer, as shown in (2), (3), and (4) in Figure 6, an approximate position of the smartphone can be narrowed down.

Next, an illuminance/luminance influence factor is determined based on the result of the position estimate. A preliminary experiment is conducted in advance using an illuminance meter for obtaining the precise illuminance/luminance influence factor at each location, and values thus obtained are assigned to lighting fixtures determined to be closer to the smartphone. An illuminance/luminance influence factor corresponding to the position of the smartphone can then be obtained.

In searching multiple smartphones, however, lighting fixtures need to be classified into groups for each smartphone under the binary search method. Consequently, as the number of smartphones to be searched increase, the required frequency of lighting control increases. Therefore, it is considered that a large amount of time is required for position estimation in an environment with many smartphones to be searched.

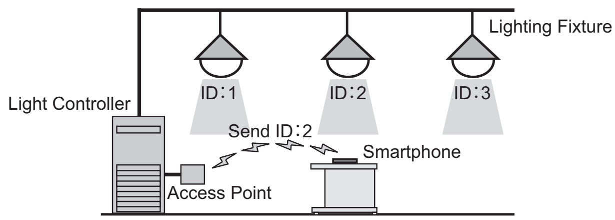


Figure 7: The position estimation method using visible light communication

4.2 A position estimation method using visible light communication

Figure 7 is a conceptual diagram for the position estimation method using visible light communication. In Figure 7, by changing luminance through repeated lighting in a pattern unique to each lighting fixture, a Lighting ID composed of 1 and 0 unique to each lighting fixture is transmitted. A built-in illuminance sensor of a smartphone measures changes in illuminance and receives a Lighting ID by analyzing illuminance values measured to determine it. The smartphone then transmits a Lighting ID received to a control device. The control device determines that the smartphone is located in the neighborhood of the point directly under the lighting fixture whose Lighting ID is transmitted to complete position estimation.

4.3 Proposal of the position estimation method using visible light communication

Visible light communication using smartphones can realize the communication speed of about 5 bps, if Smartphone1 or Smartphone2 is used, based on the result of performance verification in Section 3.1. As stated in Section 2.1, the lighting control algorithm of the Intelligent Lighting System repeats a control step every 2 seconds. Therefore, this study required completing position estimation in a single step of 2 seconds. In assigning a unique Lighting ID to each of about 100 lighting fixtures in a large office environment, the required length of a is 7 bits. In conducting visible light communication by using the algorithm shown below, 2 bits for a control signal are required in addition to the number of bits for a Lighting ID. Since the communication speed in

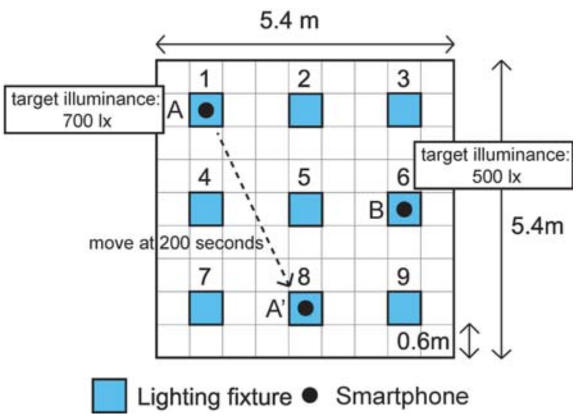


Figure 8: Experimental environment

transmitting the total of 9 bits in a single step of 2 seconds is 4 to 5 bps, position estimation can be completed in a single step by using Smartphone1 and Smartphone2, which can realize the speed of 5 bps.

The algorithm for position estimation by visible light communication is described below.

(1) Suspend luminance control for convergence to the target illuminance.
(2) Save the illuminance value of a smartphone under the current luminance as the standard value.
(3) Turn on all lighting fixtures at the lighting luminance of 90% of the current luminance and save the illuminance value of a smartphone as a standard value, as in (2).
(4) Turn on each lighting fixture at the lighting luminance of 100% or 90% of the luminance at the start of visible light communication in accordance with a Lighting ID uniquely assigned to each lighting fixture. Set lighting luminance to 100% if a Lighting ID is 1 and 90% if a Lighting ID is 0. At the same time, measure a change in illuminance by a smartphone. Set the median of standard values obtained in (2) and (3) as the threshold. Receive the Lighting ID "1" if an illuminance value is higher than the threshold and the Lighting ID "0" if it is lower than the threshold. Repeat this processing for the number of bits of a Lighting ID.
(5) Determine that the smartphone is placed directly under the lighting fixture corresponding to the Lighting ID received by it and finish position estimation.
(6) Determine the illuminance/luminance influence factor of each lighting fixture based on the result of position estimation by the method described in Section 4.1.
(7) After finishing position estimation, resume convergence to the target illuminance.

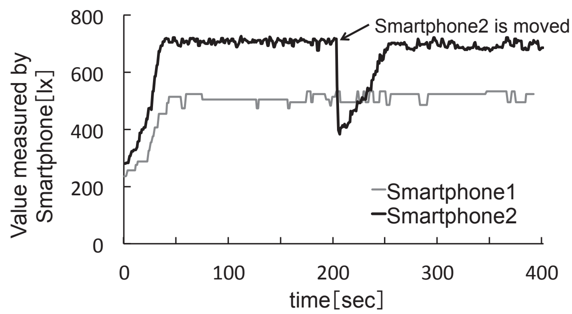


Figure 9: The history of illuminance data

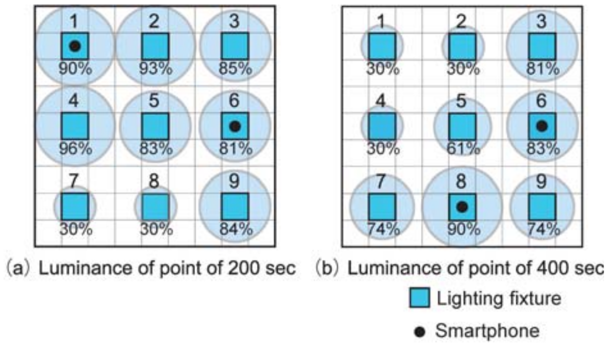


Figure 10: The status of lightings

5. Verification experiment

5.1 Experiment summary and conditions

An experiment verifying the effectiveness of this method was conducted by actually running the Intelligent Lighting System incorporating the proposed method. Base on the result in Table 1, one Smartphone1 and one Smartphone2 were used in this verification experiment.

In an experiment environment shown in Figure 8, Smartphone2 and Smartphone1 were placed at locations A and B, respectively, to estimate the positions of those smartphones. During visible light communication, the signal communication speed was set to 5 bps and the number of communication bits was set to 9.Based on the result in Figure 4 in Section 3.2, the range of visible light communication was set to within the horizontal distance of 0.3 m of the point directly under a lighting fixture. After completing position estimation, the Intelligent Lighting System was run to start illuminance convergence to the target illuminance. The target illuminance was set to 700 lx for the smartphone Smartphone2 at location A and 500 lx for Smartphone1 at location B. At 200 seconds after starting illuminance control, S2 placed at location A was moved to location A' indicated in Figure 8. The movement was made in 10 seconds. After moving the smartphones, their positions were estimated again. After the position estimation was completed, convergence to the target illuminance was resumed.

5.2 Experiment result

Figure 9 shows the transition of illuminance values obtained from illuminance sensors built in the smartphones. The horizontal axis of the graph indicates elapsed time, and the vertical axis, illuminance values from smartphones. The result of the experiment shows that the estimation of the position of the smartphone at location A was completed in about 2 seconds from the start of control and that processing started subsequently for realizing the target illuminance. Figure 9 tells that illuminance converged to target illuminance in about 50 seconds after the start of control. As the movement of the smartphones started at 200 seconds after the start of control, the measured value sharply decreased then. Figure 9 tells that, after the movement, illuminance converged to the target illuminance of 700 lx and 500 lx.

Next, Figure 10 shows the lighting state of lighting fixtures at 200 seconds and 400 seconds in Figure 9. The value indicated under each lighting fixture represents the lighting luminance of the lighting fixture with the maximum value as 100%, and a circle centered at each lighting fixture is intended to visualize its lighting luminance. Based on Figure 10, it was confirmed that the luminance of lighting fixtures near smartphones increased both before and after the movement and that the luminance of distant lighting fixtures was 30% of the minimum lighting luminance. As convergence to the target illuminance was realized in this lighting state, it was confirmed that the position estimation of smartphones was effective in controlling lighting fixtures.

5.3 Experiment conclusion

The proposed method was successful in estimating the positions of smartphones. In addition, the Intelligent Lighting System managed to realize an optimal lighting pattern even when a smartphone was moved, by obtaining the illuminance/luminance influence factor from the location estimated by the proposed method and increasing the luminance of lighting fixtures near the smartphone. It was also shown that, in visible light communication, it was possible to realize communication speed of 5 bps and estimate the positions of smartphones in about 2 seconds even with 100 lighting fixtures if a smartphone model with a faster response speed to an illuminance change was used.

In the position estimation by the proposed method, however, it is difficult to receive a Lighting ID precisely if a smartphone is placed under a point between multiple lighting fixtures as Lighting IDs of those lighting fixtures interfere with each other. A visible light communication method using a camera built in a smartphone can be cited as a solution to this problem. If a smartphone is placed on the top of a desk, two or more lighting fixtures can always be in its camera's viewing angle if they are installed at a common interval of 1.8 m. By sending a unique Lighting ID from each lighting fixture, a smartphone can receive multiple Lighting IDs at the same time. It is considered possible to estimate even the position of the smartphone placed under a point between multiple lighting fixtures by having the smartphone distinguish lighting fixtures by Lighting IDs it receives and identify the positional relationship among lighting fixtures by image processing.

6. Conclusion

This study examined the response performance of illuminance sensors built in smartphones and managed to perform visible light communication at the speed of 5 bps by using smartphones with a built-in illuminance sensor with short response time to an illuminance change. It then verified the method for estimating the position of a smartphone using visible light communication in the Intelligent Lighting System using smartphones as illuminance sensors and showed that a smartphone can be used in the Intelligent Lighting System only if it was placed near the point directly under a lighting fixture.A position estimation method for cases where a smartphone is placed under the point between multiple lighting fixtures needs to be examined in the future by further working on visible light communication using a camera built in a smartphone.

References

[1] M.Miki, "An Intelligent Lighting System and the Consortium for Smart Office Environment" Transaction of the Japanese Society for Artificial Intelligence, vol.22, no.3, pp.399-410, 2007

[2] M.Miki, S.Kuwajima, H.Ikegami, Y.Azuma, H.Nakabayashi, K.Machida, H.Aida "Verification of the smartphone's availability for illuminance sensor in an intelligent lighting system" The Science and Engineering Review of Doshisha University, vol.55, no.1, pp.65-71, 2014

[3] T.Yamazato, "Application of Visible Light Communication to LED Traffic Light," The Illuminating Engineering Institute of Japan, vol.98, no.1, pp.17-20, 2014

[4] H,Ito, M.Miki, M.Yoshimi, "Improvement in Convergence of Illuminance in Intelligent Lighting System through the Introduction of Visible Light Communication Technology," The Illuminating Engineering Institute of Japan, vol.95, no.3, pp.559-569, 2012

[5] T.Shikakura, H.Morikawa, Y.Nakamure, "Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment," The Illuminating Engineering Institute of Japan, vol.85, no.5, pp.346-351, 2001

An Intelligent System Framework for Measuring Attention Levels of Students in Online Course Environments

Omer Useche and Eman El-Sheikh
Department of Computer Science, University of West Florida, Pensacola, Florida, USA

Abstract – *With the growing popularity of online education, new challenges arise. One of those challenges, which results from the loss of face-to-face interaction, is the inability to track students' attentional response to instructional content and resources included in online courses. This paper describes an intelligent system that attempts to address this challenge, by monitoring student attention while participating in a Massive Open Online Course (MOOC). Educators and instructional designers can use the feedback gathered from the system to evaluate both the individual attentional needs of students and the effectiveness of certain instructional content and resources used in an online course, and adapt the course accordingly.*

Keywords: intelligent system, student modeling, attention measurement, computer vision, online education, MOOC.

1 Introduction

Significant recent advances in educational technology often come along with statements along the lines of "education is going to drastically change." These statements were often inspired from the advent of some new technology such as radio, film, television, personal computers and more recently, the Internet. Nevertheless, while these technologies were successful in changing other aspects of society, education remained largely unchanged. However, a growing number of online educational content and e-learning based courses have begun to popularize online instruction. This is evident in a survey conducted in 2013, which showed that at least 32% of students were taking at least one online course at the time [1]. Not only are universities offering fully online courses, many are also publishing Massive Open Online Courses (MOOCs), e-learning courses that anyone with a computer and access to the Internet can enroll in. These courses often make use of a varied number of instructional resources such as content modules, quizzes, exercises, videos, and online forums where instructors and students can collaborate and ask questions to provide a complete learning package. Moreover, many MOOCs provide relatively easy to use platforms for anyone to create their own online courses. But even with these advances, however, concerns regarding the effectiveness of online courses are still being raised.

Education has for years been interpreted as an activity that takes place in a classroom, with rows of students sitting on uncomfortable desks while directing their attention to an instructor standing in front of a whiteboard. However, while traditional methods and styles of education may seem outdated to many, there is at least one element that MOOCs have still not been able to address: the feedback that instructors get from their students based on everyday, face-to-face interaction. In fact, many of those who still oppose the use of MOOCs as an effective method of instruction stress "the lack of face-to-face tutoring as one of the main weaknesses of online courses" [2].

It could be easily argued that the job of an instructor is to facilitate the learning process. This process is driven by a complex system made up of a series of components and subsystems that determine what is to be learned and how effectively we learn. Information that must be learned by a student is interpreted as relevant or irrelevant by an executive subsystem called the decider, which makes decisions as to what is worth directing attention to based on received information, and on what needs to be encoded in the brain as knowledge [3]. Attention is an important part of this process so it follows that facilitating the learning process would require capturing students' attention.

Frequent face-to-face interactions in classroom settings inform educators of the educational needs and learning preferences of their students, and more importantly, the degree to which their instructional methods are effective based on the perceived levels of attention and motivation of their students. This type of feedback is important, as attention is necessary for acquiring new information [4]. Attention, in particular, is largely considered by cognitive psychologists, and more recently by a branch of neuroscience concerned with understanding the learning process in the brain, as a key to learning. Attention can be understood as a "mechanism that can flexibly control the flow of information from the environment to the organism and through the organism's various stages of neural processing" [5]. As such, its importance to education cannot be undermined. Indeed, the educational psychologist Robert Gagne, one of the most cited authors in instructional design and education literature, considered attention to be the first step of a series of nine instructional events that are essential in the leaning process

[6]. Sylwester and Choo advise that “teachers should adapt their instruction to the built-in biases and limitations of their students' stable attentional mechanisms” and “use imaginative teaching and management strategies to enhance the development of their students' adaptable attention processes” [7]. Yet, attention is often subjectively measured by observing students’ behavior in face-to-face settings, and recommendations and strategies for gaining students’ attention are often devised with the classroom in mind.

In the context of online educational content, how can instructors and instructional designers assure that they are grabbing students’ attention in order to assure effective learning? Moreover, how can attention in an online learning environment be measured when, more often than not, attention seems to be a subjective interpretation of a student’s behavior in class?

In an effort to provide a potential solution to this problem, we propose an intelligent system that uses computer vision technology to track and measure students’ attention levels while engaging in instructional content through a MOOC. The information gathered by the system can then be used as an indication of student engagement and as feedback that can be used to improve online educational content. This system can be used by educators and instructional designers who wish to explore solutions that could bridge the gaps that exist between online and face-to-face education and enhance online learning.

2 Related work

The study of attention can be traced back to the beginnings of experimental psychology in the middle of the 19th century, when psychologists and scholars attempted to gain an understanding of attention through means of observation and cognitive analysis in order to further understand human behavior. At the beginning of the 20th century, Geisler reviewed a series of methods that were considered for measuring attention through: 1) changes of *peripheral vision*, which becomes more narrowed when a subject concentrates on a particular image, 2) changes in *muscular strength*, by correlating muscular tension with attention, 3) *liminal and differential sensitivity*, by asking subjects to rate how much they were able to notice different types of stimuli, 4) *reaction time,* which inversely correlates attention to retardation of attention, 5) *accuracy of work*, which correlates quality or quantity of work to the degree of attention that is directed to a particular task, and 6) a semantic attempt at measuring attention by using different *graded distractors* or varied degrees of stimuli to manipulate the perceived levels of attention on a subject [8].

As technology has continued to be refined, so have the methods used for tracking, identifying and/or measuring attention in different settings. Of particular interest for the development of our proposed system are a number of methods that measure attention based on the position of a student’s head while he or she is performing a particular task, using technology to capture head pose lean as an indication of eye gaze. For instance, Ba and Ordobez attempted to recognize the Visual Focus of Attention (VFOA) in the context of a meeting by tracking head pose as an indication of visual focus through the use of a geometric model that allows their system to correlate head pose to visual gaze [9]. By doing this they could determine what participants were directing their attention to during meetings. Similarly, Ishii et al. studied the relation between eye gaze and attentional focus during conversations using eye gaze duration, eye movement and pupil size as effective variables to track for this purpose [10].

Even more relevant to our methods for measuring attention, Doshi and Trivedi found head pose to be a clear indication of directed attention by asking participants to describe where their attention was going to be directed at a particular moment, and by stimulating unconscious attention [11]. In this process, the researchers found that head movements often preceded eye gaze when shifting attention between different objects and stimuli. Moreover, Asteriadis, Karpouzis and Kollias conducted a study that highlights the importance of taking into consideration both head pose and eye gaze when tracking attention by exploring the ability of intelligent systems to replicate a human’s perception of attention [12]. In their paper they studied a series of annotations from the University of Boston dataset, which was gathered from a series of participants on perceived levels of attention from a number of pictures of subjects who were engaged in a particular task. Based on their exploration of this subject, they concluded that both eye gaze and head pose play an important role in determining attention. For instance, large head pose and small eye gaze were associated with low levels of attention, as opposed to large head pose and large eye gaze, a combination that was associated by participants as an indication of high levels of attention.

While these studies were not conducted in the context of an online course taken individually by a student, they provide relevant information that can be used for the design of our system. For instance, Stanly conducted a study in which he attempted to predict user attention by using the Microsoft Kinect to capture a wide array of variables, including body lean, head pose, position of hands and audio. For the head pose variable, he used yaw, pitch and roll angles to demine eye gaze and what the users were looking at while performing a test [13]. These variables were found to be effective in determining whether users were attentive or inattentive after comparing test results to perceived focus of attention.

3 Research Tools and Methods

The proposed system will be an extension of Open edX, the open source platform behind edX. EdX is a non-profit online site created by Harvard and MIT that offers online courses and MOOCs from some of the world’s most renowned universities and institutions [14]. Because Open edX is open sourced, developers can extend the source code and add new features to any course for use by instructors and educators who use edX's free platform for building their own interactive online courses.

For tracking head pose and estimating eye gaze, we will use the Intel RealSense camera and developer kit. The Intel RealSense camera is able to capture x, y and z coordinates, and the development kit provides a set of libraries that can be used to track a number of inputs, which includes head pose, as illustrated in Figure 1. Another important reason for using the Intel RealSense camera is the fact that Intel plans to incorporate the same technology in laptops and mobile devices in the future, which would allow systems such as the one proposed in this paper to be commonly used to enhance the online learning experience in MOOCs in the future.

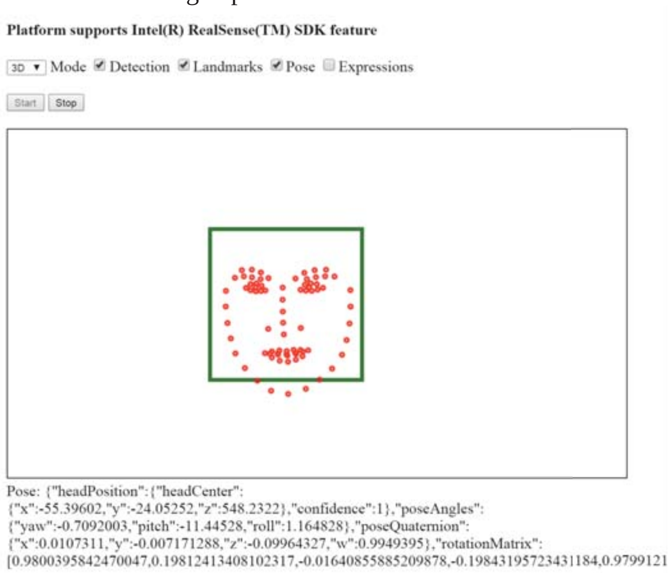


Figure 1. Intel RealSense Image Processing Example

Gathering of head pose data will be accomplished using JavaScript and JQuery frameworks provided by Intel for image processing. The data gathered from the camera will be sent to a virtual machine server that contains all back end code used by Open edX, which uses Python and Django to process the data in order to generate reports on attention levels for each student.

It is important to note that, while Open edX provides instructors and course creators with a wide array of tools that can be used to deliver instruction in several formats, for this system, we will focus on gathering attention data only when students watch instructional videos that are part of an Open edX course developed for this study. This will help instructors judge the quality and value of the instructional videos they decide to create and/or use for their own courses.

Each report generated by the attention tracking system will contain the following information:

1. URL and title of the instructional video that is played while data on attention levels is gathered.
2. Perceived levels of attention, based on perceived eye gaze using head pose data gathered from the code running in the browser. Data visualization techniques will be used in order to provide a historic, graphical representation of attention for individual students enrolled in a particular class. This information will inform instructors of individual attention levels for each student, which could help identify students who may have difficulties understanding certain content.
3. Group attention data, also represented using data visualizations tools and techniques. This will allow instructors to compare the effectiveness of different instructional content based on the data gathered from all students enrolled in a course.

Attention will be calculated based on whether it is perceived

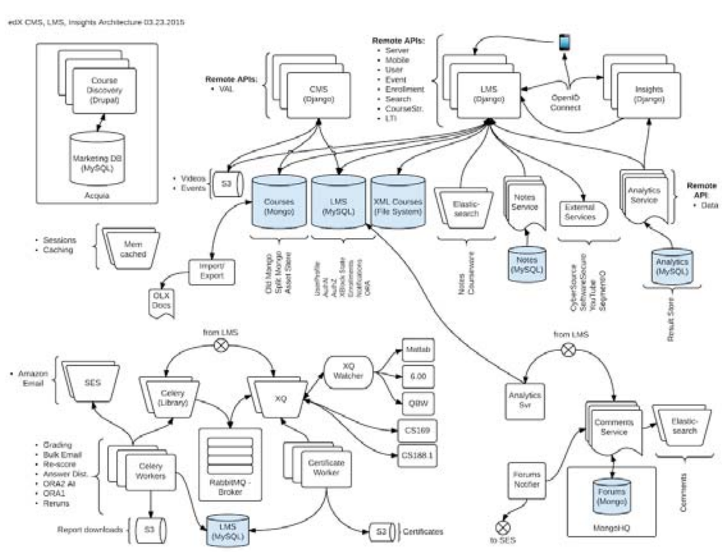


Figure 2. edX CMS, LMS, Insights Architecture [15]

that a student is directing her/his eye gaze towards an instructional video. Because the values for yaw, pitch and roll detected by the Intel RealSense camera all equal 0.0 when users are staring directly at the camera, attention will be calculated based on how these values deviate from 0.0 during the duration of instructional videos. Additional calculations may be required during development.

Lastly, a custom Open edX course is currently in development, which will be designed to work with and used to evaluate our system. The course will contain a series of instructional videos, each followed by a short test for evaluating the content taught in each of those videos.

4 System Architecture

Open edX is composed of multiple components, each serving a different purpose for the creation, storage and delivery of online courses as shown in Figure 2. Our goal is to build our system as an extension to these components in the Learning Management System (LMS) as depicted in Figure 3. The system will be modeled to fit the existing Open edX architecture, which uses a server-client architecture to deliver course content.

The architecture will rely on the following:

1) Browser modules that make use of Intel's RealSense JavaScript libraries for capturing head pose data throughout the duration of instructional videos. The data captured will then be sent in JavaScript Object Notation (JSON) format to the Django server for processing. Other modules in the browser will be responsible for visually processing attention data from the server (as charts) in an instructor module containing attention reports.
2) An xBlock module that processes the head pose data sent from the browser. The data is processed in the Django LMS server and stored as part of a student's performance history in a MySQL database.
3) Head pose data and attention levels are associated with videos watched while data was gathered in a MySQL database. The server extracts attention and head pose data from this database every time a client requests an attention report.
4) All course data is extracted from a Mongo database.

There will be two client modules responsible for preparing head pose data for processing. One module will primarily make use of the RealSense camera to capture raw head pose data. This module will then pass data to a filter module that will determine whether head pose values should be recorded or not based on variance values that will be established in the early stages of development. A calibration video, which will ask students to go through a series of steps (such as directing their gaze in different directions) will also be developed and used to establish what changes in head pose values should be considered relevant and how often head pose data should be saved for processing at the server level. This is done to limit the amount of data that will need to be sent to the server. We are exploring options for measuring attention data at the client (browser) or server level based on criteria including impact on performance and what happens when a user changes the page in the middle of a video. A possible machine learning module could also adjust head pose values as needed based on test scores and history data. Once data is passed to the server, the system will estimate attention levels and record results in a database and return an adjusted standard deviation score. A separate data visualization module then displays attention reports to instructors.

Because the LMS is built with Django, our system will rely on the Model View Controller (MVC) pattern to extend the functionality of edX as a unit called xBlock, a term used by edX to describe Python classes that are used to build small edX web applications.

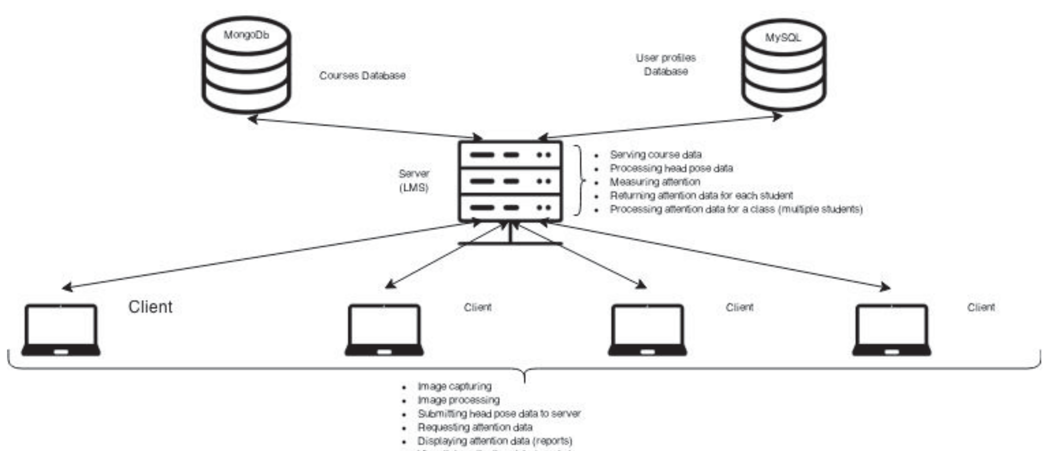


Figure 3. High-level System Architecture

5 Additional Features

The following features will be added to the system as time permits:

- Attention could be tracked while students use other course content or resources, such as when taking quizzes or going though practice exercises for subjects like math and computer science.
- Add a learning agent that further individualizes results by enhancing the accuracy of perceived attention levels based on historical data for each student and comparing perceived levels of attention with grades for individual modules.

6 Discussion

Capturing attention data with respect to specific online educational content allows instructors to further understand the needs of their online students. This could in turn improve the online education experience for teachers and students, in particular for students whose needs prevent them from attending a physical classroom. For instance, a report regarding attention levels gathered for an entire class may show that a majority of students seem to be more than 75% attentive for a particular video. In this case, it is to be expected that the overall scores for the evaluation associated with that video will be high. However, if only a small number of students show less than 50% of attention and low scores for the video in question, the instructor may determine that the content in that video may not be useful for those particular students. In this situation the instructor may then recommend that those students review additional supporting materials or use different instructional content that could potentially engage students more effectively.

Furthermore, we expect that the information that our system provides will be used as an indication of the quality and appropriateness of certain educational content. For instance, if an instructor sees that the overall attention levels for a particular video are below 50%, and the scores for the associated evaluation are low as well, the instructor may deem the video ineffective or not appropriate for the content that is being evaluated. In this case, the instructor may decide to create a new video that will be able to capture students' attention more effectively.

The above scenarios will of course require the judgment of the instructor when determining whether the instructional content is a cause for low attention levels, if some students require additional help to understand and follow that content, or if the attention data is not reflective of the quality of instruction at all. Critical judgment will require consideration of attention data and evaluation scores for individual students and for a class a whole. However, it would be helpful to add a machine learning component that makes this type of determination for the instructor. The component could advise the instructor on whether a video or instructional resource is ineffective or if one or more students need further help to understand specific content. Such a machine learning component is a potential direction for future work.

7 References

[1] Allen, I. E., & Seaman, J. (2013). *Changing Course: Ten Years of Tracking Online Education in the United States*. Sloan Consortium. PO Box 1238, Newburyport, MA 01950.

[2] Muzafarova, T., & Kaya, E. (2015). Survey of Awareness of Massive Open Online Courses (MOOC)–a Case of International Black Sea University Students, Georgia. *Journal of Education*, *3*(2), 15-19.

[3] Hays, R. T. (2006). *The science of learning: A systems theory approach.* Boca Raton, FL: BrownWalker Press.

[4] Halpern, D. F., & Riggio, H. R. (2003). *Thought and knowledge: An introduction to critical thinking*. Mahwah, NJ: Lawrence Erlbaum Associates.

[5] Geisler, W. S., & Cormack, L. K. (2011). Models of overt attention. *The Oxford Handbook of Eye Movements*. Oxford, NY: Oxford University Press.

[6] Tuckman, B., & Monetti, D. (2012). *Educational Psychology with Virtual Psychology Labs.* Belmont, CA: Cengage Learning.

[7] Sylwester, R., & Cho, J. Y. (1993). What Brain Research Says about Paying Attention. *Educational Leadership*, *50*(4), 71-75.

[8] Geissler, L. R. (1909). The measurement of attention. *The American Journal of Psychology*, 473-529.

[9] Ba, S. O., & Odobez, J. M. (2009). Recognizing visual focus of attention from head pose in natural meetings. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *39*(1), 16-33.

[10] Ishii, R., Shinohara, Y., Nakano, T., & Nishida, T. (2011). Combining multiple types of eye-gaze information to predict user's conversational engagement. In*2nd workshop on eye gaze on intelligent human machine interaction*.

[11] Doshi, A., & Trivedi, M. M. (2009, June). Head and gaze dynamics in visual attention and context learning. In *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on* (pp. 77-84). IEEE.

[12] Asteriadis, S., Karpouzis, K., & Kollias, S. (2011, May). The importance of Eye Gaze and Head Pose to estimating levels of Attention. In *Games and Virtual Worlds for Serious*

Applications (VS-GAMES), 2011 Third International Conference on (pp. 186-191). IEEE.

[13] Stanley, D. (2013). Measuring attention using Microsoft Kinect. *Rochester Institute of Technology*. Retrieved from http://scholarworks.rit.edu/cgi/viewcontent.cgi?article=5772&context=theses.

[14] About Open edX. Retrieved April 14, 2015, from https://open.edx.org/about-open-edx.

[15] edX CMS, LMS, insights architecture [Online image]. (2015). Retrieved April 12, 2015, from https://open.edX.org/contributing-to-edX/architecture.

Complex DNA and Good Genes for Snakes

Md. Shahnawaz Khan[1] **and Walter D. Potter**[2]
[1,2]Institute of Artificial Intelligence, University of Georgia, Athens, Ga, United States

Abstract - *The Snake in the Box problem deals with finding the longest snakes in an n-dimensional hypercube. The snake is supposed to follow a specific distance constraint, described by the term "spread". It is an NP-Hard problem and searching the entire search space is not a feasible option as the search space grows exponentially with increasing dimensions. In the previous paper [1], a generic pattern among the longest snakes for the Snake in the Box problem was discussed. This generic pattern was termed the DNA because of the structural and functional similarity with the DNA of living cells. It is fundamentally different for each of the four combinations of odd and even dimension and spread. In the previous paper, we discussed the simplest pattern, found in the odd dimensions with odd spread. This paper illustrates one of the complex DNA patterns that is found in the other three types of odd and even combinations of dimension and spread. It also discusses several possible combinations of transition sequences in a simple DNA pattern (similar to gene combinations in the DNA of living cells) and their effect on the length of the longest snakes that can be grown from it.*

Keywords: Snake-in-the-box, Generalization, DNA of snake, Complex DNA, Good genes

1 Introduction

Snake in the box problem is an NP-Hard combinatorial problem which has been pursued by both computer scientists and mathematicians for several decades [8]. It aims to find the longest maximal snakes in an n-dimensional hypercube. A snake is a special type of path in a graph (an n-dimensional hypercube) that does not violate certain distance constraints, described using the concept of "spread". A snake represents a path in an n-dimensional hypercube, such that if the distance between any two nodes along the path is less than or equal to the spread then the shortest distance (the Hamming distance) between them is equal to this distance along the path. If the distance between any two nodes along the path is greater than the spread then the shortest distance through the graph between these two nodes should be greater than or equal to the spread. Spread is nothing but a positive integer which represents this distance constraint and generally starts with 2. Several works have been published on the longest snakes for spread 2 and higher in several dimensions [2] [3] [7] [9] [10].

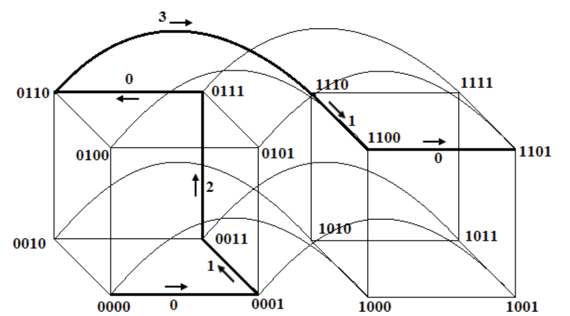


Figure 1: Transition sequences (0-based) in a 4-dimensional hypercube

The longest maximal snake for a particular dimension-spread refers to the longest snake that can be found in a dimension-spread and cannot be grown further. Snakes can be represented in several ways. Among various representations of snakes, the transition sequence is a simple and parsimonious representation. For a 0-based transition sequence representation, a non-negative integer describes the transition of nodes (the position of change of the bit between the previous node and current node when the nodes are represented in a binary code) to build a snake (see Figure 1). A canonical snake, in a transition sequence representation, is a snake transition sequence such that the first occurrence of any transition precedes the first occurrence of any other transition that is bigger than it. For example 0, 1, 2, 3 is a canonical snake in dimensions greater than or equal to 4 while 0, 1, 3, 2 is not a canonical snake as the first occurrence of transition "2" does not precede the first occurrence of transition "3" in the second case. Given the current computing resources, it is not possible to search the complete search space of the graph to find the longest maximal paths in dimensions greater than 7 [5]. Often several heuristics are applied to hunt for these snakes in the hypercube [7]. This paper discusses three things in a broad sense. First of all it briefly explains the fundamentals relating core subsequences to DNA [1]. It distinguishes the two types of DNA based on the mapping of complementary pairs and terms them as simple and complex. Later it explains the more complex DNA and how it is used in the snake. Finally, the last topic covered is about how to build simple DNA that would grow to long snakes.

2 DNA Overview

In [1], a generalized underlying structure was explained which was found to be common among the longest snakes in several dimension-spreads known so far. It also discussed the similarity they draw with the DNA of living cells. These underlying structures form the basic foundation in the potential construction of the longest snakes. Using these structures three new lower bounds of snakes in three separate dimension-spreads were found. This structure termed as the DNA was defined as "*DNA of a valid snake is the smallest portion of the snake (approximately at the center of the snake) that contains all the possible transition sequences for the snake and has one or more points of symmetry. It also defines the complementary pairs of the transitions that should be used in the remaining parts of the snake.*" One of the important characteristics of the DNA was that they have all the possible transitions that can participate in building the snake present in them. They also define the pairing of transitions that should follow in the remaining part of the snake. A concept of shadows was used to explain the underlying behavior while using the DNA structure in building snakes.

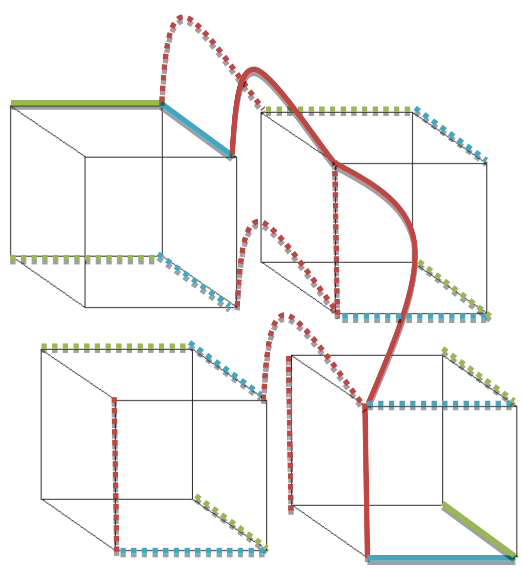


Figure 2: The longest snake and its shadows in dimension-spread (5, 3)

Also in [1], the broad differences in the DNA structure of odd and even dimensions were discussed. The discussion later was restricted to the odd spread as even spreads had multiple symmetric points, leading to multiple mappings of complementary pairs. For odd spreads, which have a single symmetric point, the even dimensions were also excluded since the remaining number of possible transitions in such dimensions, apart from the odd number of distinct transitions required to form the core of the DNA, were odd and could never have a unique mapping. In short, odd dimensions with odd spreads were the only point of discussion. Here we categorize the DNA into two, based on their unique or multiple mapping of complementary pairs. They are termed:

1. Simple DNA
2. Complex DNA

Simple DNA (DNA in odd dimensions with odd spreads) was the primary focus of the DNA discussion in [1]. In this paper we will try to explain our complex DNA. We will discuss the possible structures and will highlight some of the structures containing good genes (the transition pair combination that leads to longer snakes).

3 Decoding Complex DNA

For snakes whose spread is even or for even dimensions with odd spread, there are many reasons for non-unique mapping of complementary transition pairs. One such reason could be that there is more than one symmetric point in the DNA for even spread snakes. Also for cases where odd numbers of transitions are left to be used for defining the complementary transition pairs (after forming the core of the DNA), these odd number of transitions lead to a non-unique mapping. For these types, the DNA is more complex than the one for odd dimensions with odd spread. In this section we will try to explain some of these complex DNAs with examples. For our first illustration, let us take the first example as the longest maximal snake in dimension 7 with spread 2, which is of length 50 and is shown below.

Snake(7, 2) : 0, 1, 2, 0, 3, 1, 0, 4, 2, 1, 0, 3, 5, 0, 1, 2, 4, 0, 6, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 4, 0, 3, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 0, 6, 1, 0, 4, 2, 1, 0

This is the longest maximal snake of dimension 7. The DNA of this snake is shown in the shaded grey region while at its core the two transitions (transitions "3" and "4") are shown in red color (since spread = 2). Let us carefully examine the remaining part of the snake. We start with the complementary transition pair mapping, the mapping of the transition pairs equidistant on the left and right side of the symmetric point(s), which is defined in the DNA for this snake. For the snake shown above, when {3, 4} are together considered as the symmetric point, the complementary pairs of transition "0" are "5", "4" and "0" and are shown below with superscripts.

$6, 5^{c1}, 0^{c2}, 4, 2, 0^{c3}, 3, 4, 0^{c3}, 1, 2, 4^{c2}, 0^{c1}, 3$
("3" and"4" as the symmetric point)

The distance of complementary pair c1 is 5 from the symmetric points while the distance of complementary pair c2 is 4. The distance of complementary pair c3 is 1. The other two complementary pairs that appear in the snake are "0" with "6" and "0" with "3". These complementary transition pairs are defined when transitions “3” and “4” are considered as the symmetric point individually, as shown.

6^{c4}, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 4, 0^{c4}, 3
(“3” as the symmetric point)
6, 5, 0, 4, 2, 0, 3^{c5}, 4, 0^{c5}, 1, 2, 4, 0, 3
(“4” as the symmetric point)

In short, in the DNA we observe that “0” forms the complementary pairs with “5”, “4”, “0”, “0”, “4”, “6”, “3”, “4” and “0” on the left and right sides of the DNA. All the other complementary transition pairs that have been used in this snake are shown below with their names as superscripts. The transitions forming the pair share the same complementary pair name on the left and right side of the symmetric point (such as “c6”).

6, 5, 0, 4^{c6}, 2^{c7}, 0, 3^{c9}, 4^{c9}, 0, 1^{c7}, 2^{c6}, 4, 0, 3
(“3” and”4” as the symmetric point)
6, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 4, 0, 3
(“3” as the symmetric point)
6, 5^{c8}, 0, 4, 2^{c10}, 0, 3, 4, 0, 1, 2^{c10}, 4, 0, 3^{c8}
(“4” as the symmetric point)

The only complementary pair that cannot be defined using this DNA is that at three places “1” is paired with itself. One of the possible explanations that can be accommodated for this anomaly is that while adding these complementary pairs, since the symmetric point also changes based on the complementary pair we are choosing so the DNA includes one neighboring transition to its left or right to keep the DNA always symmetric about its symmetric point. So say if we are using complementary pair c’ where the symmetric point is “4” and the pairs are “0” and “1” as shown:

6, 5, 0, 4, 2, $0^{c'}$, 3, 4, 0, $1^{c'}$, 2, 4, 0, 3, 5

As “4” is the new symmetric point transition, so it should include “5” on its right in the DNA to make it symmetric about it (7 transitions to its left and 7 transitions to its right). Also, after including transition “5”, the pairing would look like:

0, 3, 5, 0, 1, 2, 4, 0, 6, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 4, 0, 3, 5, 0, 4, 2, 0, 3, 4, 0, 1

This inclusion would explain the apparent anomaly. Later, to shift the symmetric point again to {3, 4} the transition “1” is added to the left. Let us take an example of another longest maximal snake in dimension-spread (8, 4) which is of length 19.

Snake (8, 4): 0, 1, 2, 3, 4^{c1}, 5^{c2}, 0^{c3}, 1, 6, 3, 7, 5, 1^{c3}, 2^{c2}, 3^{c1}, 4, 5, 0, 1 (Possible core 1)
Snake (8, 4): 0, 1, 2, 3, 4^{c1}, 5^{c2}, 0^{c3}, 1, 6, 3, 7, 5, 1^{c3}, 2^{c2}, 3^{c1}, 4, 5, 0, 1 (Possible core 2)

The symmetric point in the above snake is transition “3” which is shown in blue color. Since the spread is an even number, the possible two cores encoded in red are listed as its two variants. It also shows all the complementary pairs denoted using the superscripts and are used in the remaining part of the snake. The only complexity the DNA of this snake carries is that there is more than one mapping of the complementary transition pair (e.g. transition “5” is paired with transitions “1” and “2”).

4 Simple DNA with Good Genes

In simple DNA, while defining the unique pairing, there are several possible combinations of transition sequences that can be used. These different combinations (similar to gene combinations in the DNA of living cells) decide the length of the longest snake that can be found using the DNA. This important characteristic of being a factor in deciding the length of the longest possible snake, similar to genotype mapping with phenotype in the DNA of living cells, is discussed in this section. Some of these transition combinations that contribute to the longest snakes have been identified.

Consider a snake (11, 5). The core 5 transitions that are used initially have to be different, and can be written as:
3, 1, 0, 2, 4
In this dimension (n = 11), the remaining number of possible transitions to be used in the DNA is even (it is 6). All the longest snakes that were found in [1] belonged to this category of odd dimension and odd spread. After initially placing the first k transitions in a spread-k snake (here k is equal to 5), the next two transitions (“6” and “5”) can occur in one of the following ways as shown below:

6, 5, 3, 1, 0, 2, 4, 5, 6 or 6, 5, 3, 1, 0, 2, 4, 6, 5

Interestingly, both of these types can contribute to the longest snakes depending on the dimension and spread. In the first type the two transitions outside the core are placed at equal distance from the symmetric point (in the above example, transition “5” is at a distance of three from transition “0” to the left and right side while transition “6” is at a distance of four from transition “0” to the left and right side). This type of DNA contributes to the longest maximal snake in (7, 3).
Snake (7, 3):
0, 1, 2, 3, 0, 4, 5, 1, 0, 3, 6, 4, 0, 1, 2, 3, 0, 4, 5, 1, 0

The second one in which the next two transitions form complementary pairs with each other by switching sides on the left and right side of the core, is found in (11, 5). The DNA is shown below:
DNA of Snake (11, 5):
9, 7, 0, 10, 8, 1, 4, 5, 7, 6, 10, 8, 3, 4, 2

As shown in the above example transition “8” and transition “10” form the complementary pairs. This second type of DNA was also used in the recently found longest snakes in (13, 5), (15, 7) as well as in (17, 7). The first type of DNA, when it was used for (15, 7), grew to be the longest snake of length 57 after an exhaustive search of placing the next complementary pairs. The second type also grew to be the longest snake of length 57. The exhaustive search here refers to the addition of complementary pairs to the left and right side of the DNA, in an exhaustive way as discussed in [1]. The exhaustive search was not complete for (17, 7) and 103 is the length of the longest snake found so far. For the smaller dimension-spread sometimes there are not enough transition options to form these two structures as is the case with snakes in (7, 5). Also for (9, 3) the longest known snake of length 63 was found with the second type of DNA, while the first one could only grow to a snake of length 55. The search was exhaustive for placing the next complementary pairs using both types of DNA in (9, 3). The other good gene combination in the DNA, found in the dimensions searched so far, is to add the complementary transition pairs, nearest to the symmetric point, immediately when it can be added (transition “7” and transition “4” at the second and fourteenth positions respectively in the example blow)

9, 7, 0, 10, 8, 1, 4, 5, 7, 6, 10, 8, 3, 4, 2

This is very common in bigger dimensions. For bigger dimensions, there are large numbers of transition pairs that need to be placed on the left and right side of the core structure. Laying all of them simply on alternate sides does not make good DNA for growing long snakes.

9, 0, 10, 8, 1, 4, 5, 7, 6, 10, 8, 3, 2

The DNA shown above is one such example for (11, 5). In this DNA, transition “7” and transition “4” are not placed at the second and fourteenth positions respectively as was done in the previous DNA. This type of DNA does not grow to be the longest maximal snake in (11, 5). The longest snake this DNA can grow is of length 35 (again found using the exhaustive search of placing the complementary pairs), while the longest maximal snake is of length 39.

5 Results and Discussions

While building snakes using the transition sequence and validating these snakes in the transition sequence, it was possible to find some of the longest snakes known so far in dimension 8 through 12 for spread 3, 4 and 5. The results are summarized in Table 1. The values in parentheses indicate the best known results. In Table 1, we see that the exhaustive search was not efficient enough to find the longest snakes in bigger dimensions like dimension 9, 10, 11 and 12 with spread 3. The maximally longest snakes that were found in other dimension-spreads using the exhaustive search were used for DNA analysis and replicating them in other dimension-spreads.

Dimension-Spread	Spread 3	Spread 4	Spread 5
Dimension 8	35(35*)	19(19*)	11(11*)
Dimension 9	58(63)	28(28*)	19(19*)
Dimension 10	-	47(47*)	25(25*)
Dimension 11	-	68(68)	39(39*)
Dimension 12	-	-	55(56)

Table 1: Canonical Longest Snakes found in dimension-spread

* indicates the length of the longest maximal snake

For snake (9, 3), a simple DNA analogous to the simple DNA of known maximal snakes of previous dimension-spreads was used to build the known longest snake. Since dimension-spread (9, 3) is an odd dimension with odd spread a simple DNA with unique complementary pair mapping was possible to build. The search space of paired complementary transitions was exhaustively searched for the simple DNA. One of the possible implications, if the DNA approach finds the longest maximal snake, is that the length of the longest maximal snake in (9, 3) is of length 63 since the search was complete using this approach. On the other hand, for snake (11, 3) which also happens to be an odd dimension-odd spread the search space of paired transitions using the DNA was not completed. The best found so far was of length 153. As reported in [1], the other three new longest snakes were also built using this DNA approach. For (15, 7), the search space was exhaustively searched for DNA pairing and the longest found was 57. Since the search was completed, it could also be the longest maximal snake in (15, 7). For (17, 7) and (13, 5), the search was not complete and longer snakes are possible using the complementary pairs. These results are summarized in Table 2.

Table 2: Results from the Best Gene Combinations in DNA

Dimension-spread	Good genes
(15, 7)	57^{c} (55)
(9, 3)	63^{c} (63)
(13, 5)	85 (79)
(17, 7)	103 (98)
(11, 3)	153 (157)

c - Complete search for the given structure

In the previous section, we discussed the various possibilities of placing the transitions inside the DNA and their implications due to varying mapping of complementary pairs. In Table 3, we summarize the results obtained using various gene combinations in the DNA for several dimension-spreads. The second column describes the DNA that was used for searching the snake and the third column describes the longest snake that was found using the DNA. For some of the dimension-spreads the search was complete (length is marked by a superscript c) while for others longer snakes are possible. The values in parentheses in the third column are the length of the previously known longest snakes in the dimension-spread. An asterisk * means that it is the length of the optimal longest maximal snake in the dimension-spread i.e., there is no other maximal snake that can be longer than this in the given dimension-spread. While others without an asterisk mean there is a possibility of finding a longer maximal snake in the dimension-spread.

In this paper we demonstrated the possibility of building snakes in dimension and spread combinations other than the odd dimension odd spread. The DNA for these remaining combinations is complex and uses multiple mapping. The usefulness of this type of DNA is not much appreciated because there was no new records established using this type of DNA. This complex DNA is able to explain various longest snakes while for few others, it cannot completely explain using a single hypothesis. For the Simple DNA, the results obtained were astonishing and we were able to break three new records while tying with another. Simple DNA which is found in odd dimension with odd spread shows promising results and can be used to explore further. One of the important area for exploration could be to predict the next complementary pairs outside the DNA that could help in growing the longest snake.

6 References

[1] Khan, Md. S. and Potter, W. D. (2015). "The DNA of Snakes", (under review).

[2] Meyerson, S., Whiteside, W., Drapela, T., and Potter, W.D. (2014). "Finding Longest Paths in Hypercubes: Snakes and Coils," *in Proceedings of the IEEE Symposium on Computational Intelligence for Engineering Solutions, IEEE SSCI' 2014, Orlando, FL* (to appear)

[3] Meyerson, S., Whiteside, W., Drapela, T. and Potter, W. D. (2014). "Finding Longest Paths in Hypercubes, 11 New Lower Bounds: Snake, Coils, and Symmetric Coils," (under review).

[4] G. Zémor. (1997). "An upper bound on the size of the snake-in-the-box", *Combinatorica*, 17, 287–298

[5] David Kinny. (2012). "A New Approach to the Snake-In-The-Box Problem", *Proc. 20th European Conference Artificial Intelligence*, 462–467

[6] Kochut, K. J. (1996). "Snake-In-The-Box Codes for Dimension 7", *Journal of Combinatorial Mathematics and Combinatorial Computations* 20:175-185

[7] S. Hood, J. Sawada, C.H. Wong, (2010). "Generalized Snakes and Coils in the Box"

[8] W.H. Kautz. (1958). "Unit-distance error-checking codes", *IRE Trans. Electronic Computers*, 179–180

[9] D. A. Casella and W. D. Potter. (2004). "New lower bounds for the snake-in-the-box problem: Using evolutionary techniques to hunt for snakes", *In Proceedings of the Florida Artificial Intelligence Research Society Conference*, pages 264-268

[10] S. Hood, D. Recoskie, J. Sawada, D. Wong. (2011). Snakes, coils, and single-track circuit codes with spread k, *Journal of Combinatorial Optimization*, 1–21

Table 3: Longest Snakes Built Using Various Gene Combinations in the DNA

Dimension-spread	DNA	Length
(11, 5)	**9, 2, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 1, 10**	**39^c(39*)**
(11, 5)	9, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 10	35^c(39*)
(11, 5)	9, 2, 7, 6, 5, 3, 1, 0, 2, 4, 5, 6, 8, 1, 10	33^c(39*)
(9, 3)	**7, 5, 4, 3, 1, 0, 2, 4, 3, 6, 8**	**63^c(63)**
(9, 3)	7, 2, 5, 4, 3, 1, 0, 2, 4, 3, 6, 1, 8	57^c(63)
(9, 3)	7, 2, 5, 4, 3, 1, 0, 2, 3, 4, 6, 1, 8	57^c(63)
(9, 3)	7, 5, 4, 3, 1, 0, 2, 3, 4, 6, 8	55^c(63)
(15, 7)	**13, 11, 2, 9, 8, 7, 5, 3, 1, 0, 2, 4, 6, 8, 7, 10, 1, 12, 14**	**57^c(55)**
(15, 7)	**13, 4, 11, 2, 9, 8, 7, 5, 3, 1, 0, 2, 4, 6, 8, 7, 10, 1, 12, 3, 14**	**57^c(55)**
(15, 7)	**13, 11, 2, 9, 8, 7, 5, 3, 1, 0, 2, 4, 6, 7, 8, 10, 1, 12, 14**	**57^c(55)**
(15, 7)	13, 11, 9, 8, 7, 5, 3, 1, 0, 2, 4, 6, 8, 7, 10, 12, 14	53^c(55)
(15, 7)	13, 11, 9, 8, 7, 5, 3, 1, 0, 2, 4, 6, 7, 8, 10, 12, 14	51^c(55)
(15, 7)	13, 11, 9, 7, 5, 3, 1, 0, 2, 4, 6, 8, 10, 12, 14	45^c(55)
(13, 5)	**11, 9, 2, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 1, 10, 12**	**85(79)**
(13, 5)	**11, 9, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 10, 12**	**85(79)**
(17, 7)	**15, 13, 11, 2, 9, 8, 7, 5, 3, 1, 0, 2, 4, 6, 8, 7, 10, 1, 12, 14, 16**	**103(98)**
(17, 7)	15, 13, 4, 11, 2, 9, 8, 7, 5, 3, 1,0,2, 4, 6, 8, 7, 10, 1, 12, 3, 14, 16	93(98)
(11, 3)	9, 7, 5, 4, 3, 1, 0, 2, 4, 3, 6, 8, 10	153(157)

c - Complete search for the given DNA   * indicates the length of the longest maximal snake

Value in parenthesis indicates the length of previously known longest maximal snake

Parametric and Nonparametric Mixture Models Based on Interval Regression

Roberta A. de A. Fagundes[1], Bruno A. Pimentel[2], Renata M.C.R. de Souza[2] and Francisco José A. Cysneiros[3]
[1] **Universidade de Pernambuco, Campus Gov. Miguel Arraes de Alencar, Polo Comercial, BR 104, Km 62 Caruaru (PE)- Brazil**
[2] **Universidade Federal de Pernambuco, Centro de Informática, Av. Jornalista Anibal Fernandes s/n, Cidade Universitária, CEP 50.740-560, Recife (PE)-Brazil**
[3] **Universidade Federal de Pernambuco - Depto. de Estatística, Av. Prof. Luiz Freire, s/n, Cidade Universitária,CEP 50740-540-Recife (PE) - Brazil**

Abstract— *It is increasingly common to use tools of Symbolic Data Analysis to reduce data sets and create new data ones, called symbolic data sets, without losing much information. These new data sets can be obtained for preserving the privacy of individuals when their information are present in the original data sets. In this work, we propose prediction models based on regression mixtures for interval symbolic data. The advantage of these mixtures is that they allow to assume a nonparametric function for center (midpoint) and a parametric function for range of the intervals or a nonparametric function for range and a parametric function for center. The proposed models are applied to a scientific production interval data set of institutions from Brazil. Here, this interval data set was built in order to reduce data and preserve the the information privacy. The quality of the interval prediction obtained by the models is assessed by a mean magnitude of relative error.*

Keywords: symbolic data analysis; regression mixture; interval data; interval regression

1. Introduction

The statistical treatment of interval data has been considered in the context of *Symbolic Data Analysis* (SDA) [1] which is a knowledge discovery and data management field related to multivariate analysis, pattern recognition and artificial intelligence. An extensive coverage of earlier symbolic data analysis methods can be found in [2]. *SDA* focuses on the analysis of data sets where individuals are described by variables that can represent internal variation and/or structure. Symbolic data values can be intervals, histograms, distributions, lists of values, taxonomies, etc. The term symbolic is used to stress the fact that the values are of a different nature.

Some data sets naturally consist of symbolic interval data as for example, the data set of minimum and maximum temperatures naturally represented by intervals, while many other interval symbolic data sets result from the aggregation of large classical data sets. For example, regarding scientific production data, the interest is in describing the behavior of some group of researchers rather than each scientific production by itself. By aggregating the scientific production data through institution and area of knowledge categorical variables it is obtained the information of interest; here the observed variability within each group is of utmost importance.

Regression analysis is one of the most widely used techniques in engineering, management and many other fields. In the framework of regression models for symbolic interval data, several models have been introduced. Most of these models consider parametric functions. The purpose of this work is to investigate the use of regression mixture models for interval-valued data. Four interval models are adopted and each one uses parametric and nonparametric functions. For each model a function fits midpoint data and another function fits range data of the intervals. Here, linear and robust regressions are considered as options for parametric functions and kernel regression as option for nonparametric regression.

In previous work [3], we proposed two mixture models for intervals data based on kernel and linear regressions. However, it is well known in the literature that linear regression fits the parameters using least squares approach that is sensitive to outliers. Thus, this paper generalizes the nonparametric and parametric mixture model for interval data based on the use of kernel and robust regressions for midpoint and range of the intervals. The linear regression is a particular case of the robust regression when the weights for items of the data set are identical to 1.0.

In Fagundes *et.al* [4] interval outliers are defined in the context of linear regression based on midpoint and range. The advantages of mixture models based on robust and kernel regressions are: kernel regression provides a versatile method of exploring a general relationship between variables and gives good predictions of observations yet to be made without reference to a fixed parametric model; and robust regression model is not sensitive in the presence of outliers.

The proposed models are applied to a scientific production interval data set of institutions from Brazil. An educational data analysis is a domain of application that has not yet been explored in the *SDA* framework. Concerning social services domain, Neto and De Carvalho (2002) [5] showed an application about administrative management of Brazilian cities (in Pernambuco state) using interval-valued variables. Da Silva, Lechevallier, de Carvalho, and Trousse (2006) [6] made experiments using information of web users whose aim is to cluster users with the same web usage behavior together, for this, a dynamic clustering method for interval-valued variable was used. Zuccolotto (2006) [7] presented the use of Symbolic Data Analysis in a database about job satisfaction of Italian workers through the principal component analysis method.

Here, the scientific production interval data set was built taking into account the following advantages:

1) Summarize data: Initially, the data set contains more than 140000 individuals described by points in the R^{33}. These data can be aggregated using one or more categorical variables and a new data set smaller than the old one without losing much information can be obtained;
2) Ensure the privacy of individuals: The original data contains information that explicitly identify the individuals. The generalization process allows to ensure confidentiality of original data;
3) Use higher-level category: The original data set represents scientific production of researchers whereas the aggregated data set is able to represent profiles of scientific production taking into account variability intrinsic to each profile. The aggregated of this paper is to study Brazilian production in the particular scientific area of Computer Science.

The rest of this paper is organized as follows, section 2 describes regression mixture models for interval data proposed in this paper. Section 3 describes the scientific production data considered in this paper and highlights the aggregation process adopted to obtain symbolic data. Section 4 presents a performance analysis of these models using on the scientific production data. Finally, Section 5 gives the concluding remarks.

2. Regression mixture models for interval data

Let $\Omega = 1, \ldots, n$ be a data set of n objects described by the response interval-valued variable Y and p predictor interval-valued variables $\mathbf{X} = (X_1, \ldots, X_p)$. Each object i of Ω is represented as an interval feature vector $\mathbf{z}_i = (\mathbf{x}_i, y_i)$, $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$ where $x_{ij} = [a_{ij}, b_{ij}] \in \Im = \{[a, b] : a, b \in \Re, a \le b\}$ $(j = 1, \ldots, p)$ and $y_i = [\alpha_i, \lambda_i] \in \Im$.

This method aims to find a smooth and nonlinear relationship between the interval response variable Y and the vector de interval predictor variables $\mathbf{X} = (X_1, \ldots, X_p)^T$ using the information of center (midpoint) and range of the intervals as:

2.1 Representing intervals

The interval response $[\alpha_i, \lambda_i]$ can be rewritten by:

$$y_i = [\alpha_i, \lambda_i] = [y^c - y^r/2, y^c + y^r/2]$$

Assuming the result above, the interval response variable is represented by a pair of quantitative variables (Y_c, Y_r) that describes the center and range of the intervals, respectively. Consider also that each predictor interval variable X_j is represented by a pair of quantitative variables (X_j^c, X_j^r) that describes the center and range of this interval variable.

Let $\mathbf{x}_i^c = (x_{i1}^c, \ldots, x_{ip}^c)^T$ where $x_{ij}^c = [a_{ij} + b_{ij}]/2$ and $\mathbf{x}_i^r = (x_{i1}^r, \ldots, x_{ip}^r)^T$ where $x_{ij}^r = b_{ij} - a_{ij}$. Consider $y_i^c = [\alpha_i + \lambda_i]/2$ and $y_i^r = \lambda_i - \alpha_i$. Thus, X is represented by (X^c, X^r).In this regression method, to explore Y by X is equivalent to explore Y^c by $\mathbf{X}^c$ and Y^r by $\mathbf{X}^r$, separately.

2.2 Modeling the relationship between intervals

A relationship between Y and X is give as:

$$E(Y/\mathbf{X}) = [E(Y^c/\mathbf{X}^c) - E(Y^r/\mathbf{X}^r), E(Y^c/\mathbf{X}^c) + E(Y^r/\mathbf{X}^r)],$$

$$E(Y/\mathbf{X}) = \left[m^c(\mathbf{X}^c) - \frac{1}{2}m^r(\mathbf{X}^r), m^c(\mathbf{X}^c) + \frac{1}{2}m^r(\mathbf{X}^r)\right].$$

where m^c and m^r are parametric and nonparametric functions. Examples of m^c and m^r are described in Table 1.

Table 1: Mixture Models

Models	$m^c(\mathbf{X}^c)$	$m^r(\mathbf{X}^r)$
1	$\sum_{i=1}^n \omega_i^c y_i^c$ with $\omega_i^c = \frac{K(d(\mathbf{x}^c, \mathbf{x}_i^c))}{\sum_{i=1}^n K(d(\mathbf{x}^c, \mathbf{x}_i^c))}$	$(\mathbf{x}^r)^T \widehat{\beta}^r$ with $\widehat{\beta}^r = (\mathbf{X}^{r^T} \mathbf{X}^r)^{-1} \mathbf{X}^{r^T} \mathbf{y}^r$
2	$(\mathbf{x}^c)^T \widehat{\beta}^c$ with $\widehat{\beta}^c = (\mathbf{X}^{c^T} \mathbf{X}^c)^{-1} \mathbf{X}^{c^T} \mathbf{y}^c$	$\sum_{i=1}^n \omega_i^r y_i^r$ with $\omega_i^r = \frac{K(d(\mathbf{x}^r, \mathbf{x}_i^r))}{\sum_{i=1}^n K(d(\mathbf{x}^r, \mathbf{x}_i^r))}$
3	$\sum_{i=1}^n \omega_i^c y_i^c$ with $\omega_i^c = \frac{K(d(\mathbf{x}^c, \mathbf{x}_i^c))}{\sum_{i=1}^n K(d(\mathbf{x}^r, \mathbf{x}_i^r))}$	$(\mathbf{x}^r)^T \widehat{\beta}^r$ with $\widehat{\beta}^r = (\mathbf{X}^{r^T} \mathbf{W}^r \mathbf{X}^r)^{-1} \mathbf{X}^{r^T} \mathbf{W}^r \mathbf{y}^r$
4	$(\mathbf{x}^c)^T \widehat{\beta}^c$ with $\widehat{\beta}^c = (\mathbf{X}^{c^T} \mathbf{W}^c \mathbf{X}^c)^{-1} \mathbf{X}^{c^T} \mathbf{W}^c \mathbf{y}^c$	$\sum_{i=1}^n \omega_i^r y_i^r$ with $\omega_i^r = \frac{K(d(\mathbf{x}^r, \mathbf{x}_i^r))}{\sum_{i=1}^n K(d(\mathbf{x}^r, \mathbf{x}_i^r))}$

where $\mathbf{W}^c$ and $\mathbf{W}^r$ are weight matrices for center and range data, respectively;$\mathbf{X}^c$ and $\mathbf{X}^r$ input data matrices for center and range data, respectively; $\mathbf{y}^c$ and $\mathbf{y}^r$ are response data vectors for center and range data, respectively.

In this paper four parametric and nonparametric regression mixture models are investigated:

1) The model 1 (here called $MM : CK+RL$) combines kernel regression for center data and linear (multiple) regression for range data.
2) The model 2 (here called $MM : CL+RK$) combines linear (multiple) regression for center data and kernel regression for range data.
3) The model 3 (her called $MM : CK+RR$) combines kernel regression for center data and robust regression for range data.
4) The model 4 (here called $MM : CR+RK$) combines robust regression for center data and kernel regression for range data.

The regression mixture models 1 and 2 assume that the data set does not include interval outliers. The parameter are estimated from data using the least squares approach. The regression mixture models 3 and 4 consider interval outliers. Here, these outliers can be identified by investigating if there are point outliers on midpoint or range data of the intervals. The Fisher scoring method [8] can be easily applied to get $\hat{\boldsymbol{\beta}}^c$ and $\hat{\boldsymbol{\beta}}^r$ where the process for $\hat{\boldsymbol{\beta}}^c$ and $\hat{\boldsymbol{\beta}}^r$ can be interpreted as a modified least square.

There is a number of popular robust criterion function ρ. The least square is a particular case when the weight given to each residual is equal to 1.0. So, the robust regression method can be classified by the their ψ function that controls the weight given to each residual (Montegomery *et. al* [8]).

These regression mixture models use Guassian kernel functions and squared Euclidean distance applied to center and range of the intervals data. In these kernels functions, the bandwidth h is the standard deviation for a normal distributions centered on $\mathbf{x}_i^c$ or $\mathbf{x}_i^r$.

3. Scientific Production Data

The data were extracted from the National Council for Scientific and Technological Development (`http://www.cnpq.br`) that is an agency of the Ministry of Science, Technology and Innovation in order to promote scientific and technological research and the training of human resources for research in the country. Other important Brazilian agency is the Coordination for the Improvement of Higher Level Personnel (`http://www.capes.gov.br`) whose main activity is to evaluate the Brazilian research institutes. This agency evaluates the Brazilian post-graduate courses based on the scientific production of the researchers.

The scientific production of each researcher is described by a set of 33 continuous numerical and 3 categorical variables. The continuous variables are averages of production values computed in three years (2006, 2007 and 2008) for each researcher. They are: 1. National journal, 2. International journal, 3. Presentation of papers, 4. Books , 5. Chapter of book, 6. Other publications, 7. Summary of journal, 8. Summary of annals, 9. Publication, 10. PhD guidelines finished, 11. Master guidelines finished, 12. Specialization guidelines finished, 13. Graduate guidelines finished, 14. UR guidelines finished, 15. PhD guidelines unfinished, 16. Master guidelines unfinished, 17. Specialization guidelines unfinished, 18. Graduate guidelines unfinished, 19. UR guidelines unfinished, 20. Guidelines finished, 21. Guidelines unfinished, 22. Other intellectual productions, 23. Other types of production, 24. Registered software, 25. Unregistered software, 26. Unregistered product, 27. Registered techniques, 28. Unregistered techniques, 29. Technique works, 30. Technique presentations, 31. Other production-related techniques, 32. Techniques and 33. Artworks. The categorical variables are: institute, area of knowledge and sub-area of knowledge.

The data base considers 141260 researchers from 410 institutions such as federal, state, municipal and private universities, colleges integrated, colleges, institutes, schools, technical education centers that have at least one course of masters or doctorate degree recognized by Coordination for the Improvement of Higher Level Personnel, public institutes of scientific research, public technological institutes and federal centers of technological education or research laboratories and development of state enterprisers. Each institution is organized into several areas of knowledge such as Biological Sciences, Exact Science, Engineering, Agricultural Science, Health Sciences, Applied Social Sciences, Humanities and Linguistics-Literature-Arts. Each area of knowledge is divided in 76 sub-areas of knowledge. Each researcher is related to only one sub-area of knowledge.

Let Ω be a data set of researches indexed by $i(i = 1, ..., 141260)$. Each researcher is described by a vector of 33 continuous numerical and 3 categorial values $v_i = (v_i^1, \ldots, v_i^3 3, c_i^1, c_i^2, c_i^3)$ where $v_i^j \in \Re$ $(j = 1, \ldots, 33)$ and c_i^1, c_i^2, c_i^3 are the institute, area and sub-area of knowledge of the researcher i. Tools of SDA are applied on this researcher data base in order to build new units. These new units are modeled by interval symbolic data. Here, three reasons are considered by using SDA: 1) to reduce the size of the original base since SDA starts as mining data process applied on a large data base; 2) to ensure the privacy of researches and 3) to use higher-level objects described by the variables that allow to take into account variability and/or uncertainty.

This work analyzes the interval scientific production data of Brazilian institutes. In this context, the interval data proposed for [9] and [10] available at `http://www.cin.ufpe.br/~bap/ScientificProduction` are aggregated the sub-area of knowledge categorical variables Computer Science generating a new data base of size 166. These data represent new concepts of scientific production are describe in Table 2. In order to apply regression models to this interval data set, predict variables are choice using an *a priori* knowledge of software estimation experts and they are: $NPhd$(PhD guidelines finished), $NMaster$ (Master guidelines finished) and $NScientific$ (UR (scientific) guidelines finished). These predictor variables explain the number of publications of the researchers from science computer response variable ($NPub$ - Publications) as it is showed interval graph in Figure 1, 2 and 3. In these Figures illustrates the interval-valued data set containing very large rectangles in a data set can mean the presence of atypical intervals. Note in these figures that, there are rectangle *outliers* that are remote in the X and Y coordinate.

Table 2 presents part of the interval scientific production data set from proposed in this work. Each row of this table corresponds to a number of publications of the researchers from science computer described by the variables $NPhd$, $NMaster$, $NScientific$ and $NPub$ with an interval description the minimum and maximum values of these variables . The rows of this table describe concepts of scientific production. These concepts model the information taking into account variability. There is remote intervals on the response variable can be easily observed in Figure 1, 2 and 3 in both coordinates(X and Y) and confirmed by the values described in Table 2.

Table 2: Concepts of scientific production data described by interval data.

	$NPub$	PhD guidelines ($NPhd$)	Master guidelines ($NMaster$)	UR guidelines ($NScientific$)
1	[0,14.25]	[0,6.5]	[0,1]	[0,2]
2	[0.5,16.25]	[0,0]	[0,0]	[0,5.75]
3	[0,28.5]	[0,817]	[0,3.75]	[0,10]
...	...	...	...	...
162	[0,39.75]	[0,0.25]	[0,4.75]	[0,26.6]
163	[0,9.25]	[0,0.25]	[0,0]	[0,2.75]
164	[0,8.75]	[0,1.75]	[0,1.25]	[0,0.75]

4. Performance analysis

The four regression mixture models ($MM : CK + RL$, $MM : CL+RK$, $MM : CK+RR$ and $MM : CR+RK$) proposed in this paper are applied to scientific production interval data set and a performance analysis is carried out. Moreover, in order to make a comparative study with other distribution-free regression methods of the *SDA* literature, the interval robust regression model based on center and range information [4] (here called IRR), the interval linear regression model based on center and range information [11] (here called ILR) and the interval kernel regression model based on center and range information[3] (here called IKR) are also to applied to this data set.

The prediction accuracy of the models are measured by the mean magnitude of relative error ($MMRE$) that is estimated by the hold-out method in the framework of a Monte Carlo simulation with 1000 replications. The test and learning sets are randomly selected from each input data set. The learning set corresponds to 75% of the data and the test data set corresponds to 25%. The experiments are performed using the Language R `{http://www.r-project.org/}`. The $MMRE$ is given as

$$MMRE = \sum_{i=1}^{n} \frac{1}{2n} \left\{ \left| \frac{\alpha_i - \hat{\alpha}_i}{\alpha_i} \right| + \left| \frac{\lambda_i - \hat{\lambda}_i}{\lambda_i} \right| \right\}. \quad (1)$$

The problem of an automatic choice of the bandwidth (h) is important in kernel regression. An appropriate bandwidth can be defined by studying the $MMRE$ behavior regarding different bandwidths [12]. Here, the bandwidth is chosen based on the lowest value of the $MMRE$ that in this simulation study is 0.01.

The comparison between the regression methods of the SDA literature and regression mixture methods are achieved by applying the statistical *Wilcoxon* test for not paired samples at a significance level of 5%. Let μ_1 and μ_2 be the average of the $MMRE$ for quantitative data and interval-valued data, respectively. The null(H_0) and alternative (H_1) hypotheses are:

$$\begin{cases} H_0 : \mu_1 = \mu_2 \\ H_1 : \mu_1 < \mu_2 \end{cases}$$

Tables 4, 6 and 8 present the comparison between regression methods based on the *p-value* of the statistics tests.

4.1 Results considering $NPhd$ and $NMaster$ explanatory variables

Table 3 presents the average and standard deviation of the $MMRE$ for IRR, IKR, ILR, $MM : CK + RL$, $MM : CL+RK$, $MM : CK+RR$ and $MM : CR+RK$ models. The predict variables $NPhd$(PhD guidelines) and $NMaster$ (Master guidelines) explain the response variable, that is, number the publications of the researches of computer science ($NPub$). The comparison between each two methods is achieved based on the wilcox-test for a difference in mean of the $MMRE$ with independent samples at the significance 5%. Table 4 shows the wilcox-test statistics computed in this study. From the values in Tables 3 and 4 some remarks are extracted.

- The IRR and $MM : CR+RK$ regression models have the best prediction performances due to the presence of *outliers* in the center of intervals as it is showed in Figure 1. Thus, the use robust regression in the center of the intervals is indicated. However, it can be observed that the $MM : CR+RK$ should be preferred because the mathematical coherence for intervals;
- The regression mixture models based on nonparametric functions to model center data $MM : CK + RR$ and $MM : CK + RL$ have the worst performances. The ILR and IKR methods have similar performances;
- The p-values in Table 4 support the previous remarks.

Table 3: $MMRE$ for interval data set.

Models	$Average \pm St.Deviation$
ILR	4.47943 $\pm$ 1.675311
IRR	**2.749475 $\pm$ 1.163158**
IKR	4.282684 $\pm$ 4.660366
MM:CK+RL	7.460052 $\pm$ 7.410414
MM:CL+RK	3.262939 $\pm$ 1.251656
MM:CK+RR	7.530917 $\pm$ 7.719749
MM:CR+RK	**2.998639 $\pm$ 0.874532**

Table 4: Comparison between regression methods.

Comparison	*p-value*
$\mu(MM:CRRK) \times \mu(IRR)$	6.9508 $\times 10^{-08}$
$\mu(MM:CRRK) \times \mu(MM:CKRL)$	6.0250 $\times 10^{-69}$
$\mu(MM:CRRK) \times \mu(MM:CLRK)$	2.5151 $\times 10^{-08}$
$\mu(MM:CRRK) \times \mu(MM:CKRR)$	3.8637 $\times 10^{-66}$
$\mu(MM:CRRK) \times \mu(ILR)$	2.8836 $\times 10^{-114}$
$\mu(MM:CRRK) \times \mu(IKR)$	1.8853 $\times 10^{-17}$

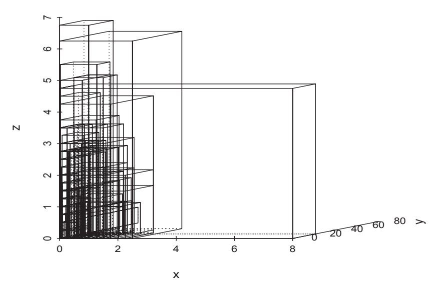


Fig. 1: Interval plot: $NPhd$ (X), $NMaster$ (Z) and $NPub$ (Y).

4.2 Results considering $NPhd$ and $NScientific$ explanatory variables

Table 5 shows the average and the standard deviation of the $MMRE$ for IRR, IKR, ILR, $MM : CK + RL$, $MM : CL + RK$, $MM : CK + RR$ and $MM : CR + RK$ models. The predict variables $NPhd$ (Phd guidelines finished) and $NScientific$(UR guidelines finished) explain the response variable, that is, number the publications of the researches of science computer ($NPub$). The comparison between each two methods is achieved based on the wilcox-test for a difference in mean of the $MMRE$ with independent samples at the significance 5%. Table 6 shows the wilcox-test statistics computed in this study. From the values in Tables 5 and 6 some remarks are extracted.

- The $MM : CR + RK$ regression mixture model proposed in this paper exhibited the best values of performance. As expected, this model consider a parametric function for the center that is sensitive to outliers and a nonparametric function for the range guaranteed the mathematical coherence for intervals;
- The ILR, IKR and IRR regression models and $MM : CL + RK$ have similar performances in terms of $MMRE$. However, it can be observed that the $MM : CL + RK$ regression mixture model should be preferred because the mathematical coherence for intervals.
- The $MM : CK+RL$ and $MM : CK+RR$ regression mixture models have the worst performance among all the regression models because there are remote rectangles as it is displayed in Figure 2;
- The p-values in Table 6 support the previous remarks.

Table 5: $MMRE$ for interval data set.

Models	$Average \pm St.Deviation$
ILR	5.327373 $\pm$ 1.919245
IKR	4.901354 $\pm$ 5.154982
IRR	3.739869 $\pm$ 1.425767
MM:CK+RL	8.225326 $\pm$ 6.470068
MM:CL+RK	5.019499 $\pm$ 1.996001
MM:CK+RR	8.368377 $\pm$ 7.055955
MM:CR+RK	**2.619647 $\pm$ 1.513572**

Table 6: Comparison between regression methods.

Comparison	*p-value*
$\mu(MM:CRRK) \times \mu(IRR)$	6.9508 $\times 10^{-08}$
$\mu(MM:CRRK) \times \mu(MM:CKRL)$	6.0250 $\times 10^{-69}$
$\mu(MM:CRRK) \times \mu(MM:CLRK)$	2.5151 $\times 10^{-08}$
$\mu(MM:CRRK) \times \mu(MM:CKRR)$	3.8637 $\times 10^{-66}$
$\mu(MM:CRRK) \times \mu(ILR)$	2.8836 $\times 10^{-114}$
$\mu(MM:CRRK) \times \mu(IKR)$	1.8853 $\times 10^{-17}$

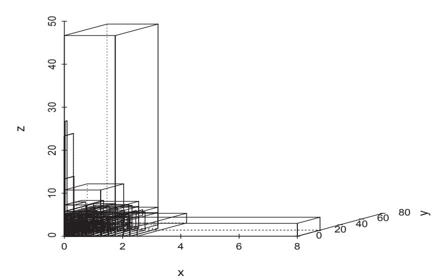


Fig. 2: Interval plot: $NPhd$ (X), $NScientific$ (Z) and $NPub$ (Y).

4.3 Results considering $NMaster$ and $NScientific$ explanatory variables

Table 7 presents the average and the standard deviation of the $MMRE$ for IRR, IKR, ILR, $MM : CK + RL$,$MM : CL+RK$, $MM : CK+RR$ and $MM : CR+RK$ models. The predict variables $NMaster$ (Master guidelines) and $NScientific$(UR guidelines) explain the response variable, that is, number the publications of the researches of computer science ($NPub$). The comparison between each two methods is achieved based on the wilcox-test for a difference in mean of the $MMRE$ with independent samples at the significance 5%. Table 8 shows the wilcox-test statistics computed in this study. From the values in Tables 7 and 8 some remarks are extracted.

- The IKR, $MM : CL + RK$ and $MM : CR + RK$ regression mixture models are similar in terms of $MMRE$. However, it can be observed that the $MM : CR + RK$ and $MM : CL + RK$ regression mixture models should be preferred because the mathematical coherence for intervals. As expect, $MM : CL + RK$ is the best option in terms $MMRE$;
- These results show that the IRR and ILR models in the SDA literature have similar performance;
- The $MM : CK + RR$ and $MM : CK + RL$ regression mixture models have worst performance in terms $MMRE$ because the parametric form exists in the center of the intervals illustrated in the Figure 3 highlights the presence of a parametric form between the explanatory and response variables.

Table 7: $MMRE$ for interval data set.

Models	$Average \pm St.Deviation$
ILR	3.995003 $\pm$ 1.395748
IKR	4.878269 $\pm$ 6.794949
IRR	3.38593 $\pm$ 1.434064
MM:CK+RL	8.967796 $\pm$ 9.944816
MM:CL+RK	**2.643926 $\pm$ 1.068585**
MM:CK+RR	8.645091 $\pm$ 10.34362
MM:CR+RK	3.219962 $\pm$ 1.279143

Table 8: Comparison between regression methods.

Comparison	*p-value*
$\mu(MM : CLRK) \times \mu(IRR)$	5.7689 $\times 10^{-38}$
$\mu(MM : CLRK) \times \mu(MM : CRRK)$	2.5336 $\times 10^{-27}$
$\mu(MM : CLRK) \times \mu(MM : CKRL)$	1.2787 $\times 10^{-75}$
$\mu(MM : CLRK) \times \mu(MM : CKRR)$	6.4910 $\times 10^{-65}$
$\mu(MM : CLRK) \times \mu(ILR)$	6.2081 $\times 10^{-114}$
$\mu(MM : CLRK) \times \mu(IKR)$	5.9507 $\times 10^{-24}$

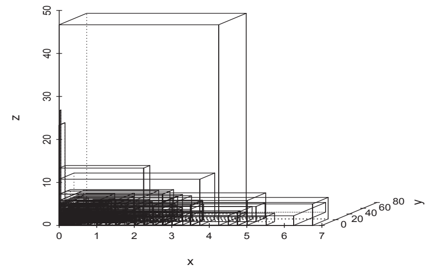


Fig. 3: Interval plot: $NMaster$ (X), $NScientificr$ (Z) and $NPub$ (Y).

4.4 Evaluating predicted interval

Figure 4 illustrates the predicted intervals versus real intervals. The predicted intervals are obtained by the $MM : CR+RK$ method from a test data set (scientific production) based on $NPhd$ and $NMaster$ explanatory variables. Figure 5 presents the predicted intervals based on $NPhd$ and $NScientific$ explanatory variables and $MM : CR + RK$ method. Figure 6 exhibits the predicted intervals based on $NMaster$ and $NScientific$ explanatory variables and $MM : CL + RK$ method.

As expect, the regression mixture methods have good linear fittings between predicted and real intervals. This means that the $MM : CR + RK$ method with $NPhd$ and $NMaster$ explanatory variables (Figure 4) and $MM : CL+ RK$ method with $NMaster$ and $NScientific$ explanatory variables (Figure 6) are adequacy to estimating scientific production data set.

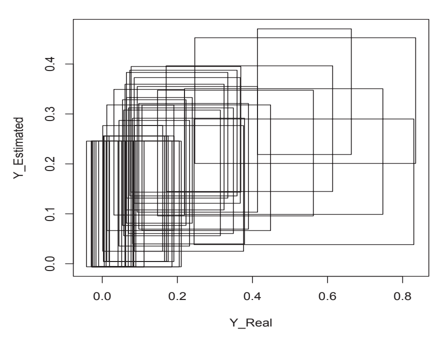


Fig. 4: Interval plot: Estimated Y versus real Y based on the $MM : CR + RK$.

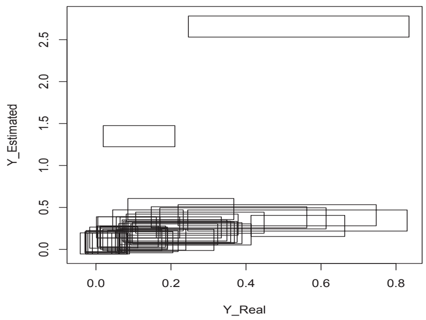


Fig. 5: Interval plot: Estimated Y versus real Y based on the $MM : CR + RK$.

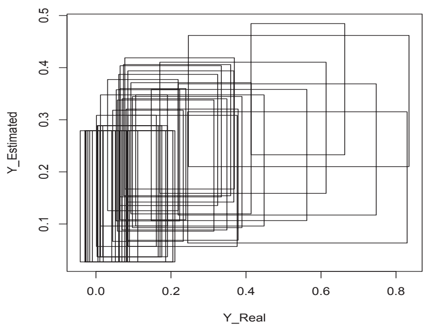


Fig. 6: Interval plot: Estimated Y versus real Y based on the $MM : CL + RK$.

5. Conclusion

This work presented a study of Brazilian scientific production based on tools of the Symbolic Data Analysis (SDA) and regression mixture models. The data set is originally formed by researchers from different centers of research. Tools of Symbolic Data Analysis are applied in order to model the information regarding variables that take into account variability. So, new units are obtained and they are described by interval data. Each unit represents aggregated data under the same institute and subject of research. The aggregation process provided the following advantages: reduction of the size of the base, assurance of the privacy of individuals.

The regression mixture models use kernel functions in center or range providing a versatile method of exploring a general relationship between interval variables and gives good predictions of interval observations yet to be made without reference to a fixed parametric model. Furthermore, the regression mixture models utilize robust regression in center or range as it is an alternative to least squares estimation in the presence of outliers.

References

[1] Billard L. and Diday E. *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, West Sussex, England, 2006.

[2] Diday E. and Noirhomme-Fraiture M. *Symbolic Data Analysis and the SODAS Software*. John Wesley & Sons, ltd, 2008.

[3] Fagundes, R.A.A., Souza, R.M.C.R. and Cysneiros, F.J.A. Interval Kernel Regression, vol. 128,371-388, 2014.

[4] Fagundes, R.A.A., Souza, R.M.C.R. and Cysneiros, F.J.A. Robust Regression with Application to Symbolic Interval Data, vol. 26,564-573, 2013.

[5] Neto, E.A.L., De Carvalho, F.A.T.: Symbolic Approach to Analyzing Administrative Management. The Electronic Journal of Symbolic Data Analysis 1(1), 1âĂŞ13, 2002.

[6] Silva, A.D., Lechevallier, Y., De Carvalho, F.A.T., Trousse, B.: Mining web usage data for discovering navigation clusters. In: IEEE Symposium on Computers and Communications,910âĂŞ915, 2006.

[7] Zuccolotto, P.: Principal components of sample estimates: an approach through symbolic data analysis. Applied and Metallurgical Statistics 16, 173âĂŞ192, 2006.

[8] Montgomery D.C., Peck E.A. and Vining G. G. *Introduction to Linear Regression Analysis*, Wisley-Interscince. 2006.

[9] Pimentel,B.A., Nóbrega, J.P. & Souza,R. M.C.R. Using Weighted Clustering and Symbolic Data to Evaluate InstitutesâĂŹs Scientific Production, 435-442, 2012.

[10] Pimentel,B.A. & Souza,R. M.C.R. A weighted multivariate Fuzzy C-Means method in interval-valued scientific production data, Expert Systems with Applications, 41, 3223-3236, 2014.

[11] Lima Neto E.A. and De Carvalho F.A.T. Centre and Range method for fitting a linear regression model to symbolic interval data. *Computational Statistics and Data Analysis*, 52:1500-1515, 2008.

[12] Sheather, S.J. and Jones, M.C., A reliable data-based bandwidth selection method for kernel density estimation, *Journal of the Royal Statistical Society*, Series B, 53:683-690, 1991.

Conceptual Design of a Smart Classroom Based on Multiagent Systems

Jose Aguilar[1,2], **Priscila Valdiviezo**[2], **Jorge Cordero**[2] **and Manuel Sánchez**[3]
[1] CEMISID, Universidad de Los Andes, Mérida, Mérida, Venezuela
[2]DCCE, Universidad Técnica Particular de Loja, Loja, Loja, Ecuador
[3]Universidad Nacional Experimental del Táchira, San Cristóbal, Táchira, Venezuela

Abstract - *The smart environments have been used in different domains: home, educational and health centers, etc. Particularly, a smart environment in education must integrate different aspects linked to virtual and presencial education, the profile of the students, to the pedagogical paradigm used, etc., in real time. In this paper we characterize a smart classroom considering these aspects, using the multiagent systems paradigm. Particularly, we define the different components of a smart classroom with their properties. Based on that, we describe these components like agents using MASINA, a methodology to specify multiagent systems. We define two frameworks of agents which describe the different types of components in a smart classroom (of software and of hardware), and give examples of applications of these two frameworks in a device and a software of a smart classroom. Finally, we show an example of conversation in a smart classroom based on our multiagents approach, specifically in a work session.*

Keywords: Smart Classroom, Multiagent System, AmI, Middleware

1 Introduction

The new advances in information technology, mainly in ubiquitous and pervasive computing, allow us interact with all computation devices as a whole, and express our tasks in a high abstraction level very natural. The research domain in Ambient Intelligence (AmI) is motivated by this idea. In [1] has been defined an AmI for the education as any space where ubiquitous technology helps the learning process in an unobtrusive manner.

Smart Classroom for the education is one of the challengers in the area of AmI. AmI brings new ideas and approaches into educational process at every level of education. Redefine a classroom is an inevitable trend, where is integrating sensor technology, communication technology, artificial intelligence, etc. into classroom. The idea is to exploit the smart environments in an educational process, considering the specific aspects on the educational domain (presencial and virtual education, self-formation, etc.) and the advances in ubiquitous computing, augmented reality, mobile computing, etc.

A previous work has developed a middleware based on multiagent systems to support smart educational environments [2]. This middleware proposes different levels, one for the management of the multiagents community, other to manage the access to services, applications, etc., and the last one to characterize the different components (of software and hardware) of these environments. This specific level can be customized for one particular educational environment.

In this paper we define a Smart Classroom based on the multiagents paradigm, called SaCI (Salón de Clase Inteligente, for its acronym in Spanish). We define the different components in a Smart Classroom and propose two types of agents (frameworks), one to characterize the software components and the other to define the hardware components on this environment. Additionally, we test the capabilities of description of this frameworks in one component of software and of hardware. Finally, we modeled a conversation in the Smart Classroom.

Our Smart Classroom is supported by the middleware to smart environments based on multiagent systems developed in [2]. This middleware is customized, in order to describe the different components of the Smart Classroom. The description of the agents and of the conversations are made using MASINA [3], a methodology to specify multiagent systems.

This work is organized like follow: the next part present some concepts linked to a Smart Classroom, the next section presents the state of the art about Smart Classrooms. Section four presents the middleware, and the next section presents the different components of a Smart Classroom, the frameworks based on agent to describe them, and one example of application of these frameworks. Finally, we present one example of conversation in a Smart Classroom based on multiagent systems.

2 State of the Art

There are a lot of works in the domain of AmI, and in the domains of e-learning, cloud learning, etc. that is part of the technology used for education. Additionally, there are a lot of products of software in the education domain (Intelligent Tutorial Systems (ITS), Computer-Supported Collaborative Learning Systems (CSCL), Content Management Systems, (CMS), Learning Management Systems, (LMS), Virtual Learning Environments (VLE), Learning Content Management Systems (LCMS), etc.). For our case, we are interesting in the works that propose AmI for educational purpose. In this section we present some of these works.

A first work is the state of the art presented in [4], in the area of ubiquitous and context-aware learning in a special type of smart environment: smart workplaces. They introduce the concept of smart learning which include the areas of ubiquitous and context-aware learning. They present several works in the domain of smart offices [4-6] and smart learning (ubiquitous and context aware learning) [4, 7].

Other interesting project is the Smart Classroom project [8]. It proposes a real-time interactive classroom by bringing pervasive computing technologies into traditional distance learning. The goal of this project is to integrate the teacher's experience in tele-education and in the traditional classroom education. They propose a 3D user in an augmented classroom, so that the teacher can interact with the remote students like interacts with the local students.

The AARTIC project [9] proposes an intelligent environment that assists software engineering students in their assignments. The system helps the student to understand concepts using. They propose two agents: the first supervises students' activities in the system, and the second allows the teacher to evaluate the class as a whole and each student.

In [10] is proposed an ubiquitous learning environment in order to support students doing learning activities. It can detect the physical objects on the environment and provides a recommendation based on it. The students can contact, interact, and collaborate with other students, in order to reach learning goals.

In [1] proposes two ubiquitous devices for an intelligent learning environment: The Experience Recorder which is an embedded system that records the paths followed by a student in a classroom, and an iBand which is a wearable bracelet-like device that has information about the students.

Other smart environment is ISABEL [11], which is a multi-agent e-learning system, where the idea is to divide the students in groups with similar profiles, where each group is managed by a tutor agent. Additionally, there a teacher agent associated to each e-learning site, which provides recommendations to the student agents, active in that site in a given moment.

[12] presents an augmented reality environment to help students in the learning process. The environment, called SESIL, recognizes book pages and specific elements of interest within a page, as well as perceives interaction with pens/pencils, etc. The environment gives an assistance unobtrusive, context - aware of the student. In this way, the learning process can be improved with the presentation of related material.

The SMART Classroom project has like goal to combine the latest technological teaching tools, collaborative teaching and pedagogy, to create an education service environment [13]. They propose three concepts to describe a Smart Classroom: the "Smart Pedagogy", the "Smart Learning Space" and the "Smart Teaching Solutions". The "Smart Pedagogy" is based on the different pedagogical methodologies, because a Smart Classroom must allow the application of different pedagogical methodologies to provide more interactive alternatives that a traditional teaching environment. The "Smart Learning Space" covers both physical and virtual teaching spaces, defining the essential furniture and technical specifications of ideal physical classrooms, and the most important features of virtual learning platforms. This space must allow flexibility and the ubiquity of the learning process. Finally, the "Smart Teaching Solutions" are the different tools and materials for the learning process like the learning objects, games, simulators, etc.

3 Middleware used by SaCI

In this work we are going to use the middleware for Intelligent Learning Environments proposed in [2] This middleware propose six level (see Figure 1). The physical level has the different software components to interconnect the different elements of the environment (software or hardware), like APIs, etc. It is connected and worked with the Operating System, for the different activities to be carried out in the environment.

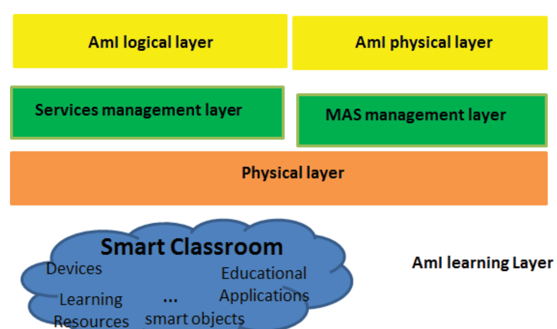


Figure 1. Middleware for Intelligent Learning Environments instanced in a smart classroom.

The multi agent system management level is composed by a multi-agent community to support the execution of multi-agents applications. This level follows the FIPA standard and has been defined in previous papers [14]. The services management layer has the responsibility of finding, searching, etc. services required in a given moment by the applications in the platform. The AmI physical layer represents the different devices present on the environment, represented like agents. The AmI logical layer represents the different software components that are used in the educational platform. This layer describes the different software components of SaCI like agents, but additionally it has two agents: a profile agent to represent each student in SaCI (its capabilities, learning style, etc.), and the tutor agent to represent the professor in SaCI. Finally, the AmI learning layer is where are deployed the different devices and software of SaCI.

In this paper we are going to define the agents that compose the AmI physical and logical layers. For that, we are going to define a general type of physical agent (framework) to represent the different devices in the educational platform (intelligent or not). Similarly, we are going to define a general logical agent (framework) to represent the different educational softwares, or the individuals present on the environment.

4 Agents Specification for SaCI

In this work we are going to use the middleware for Intelligent Learning Environments

In this section we specify the agents of SaCI, basically they define the physical and logical layers of the previous Middleware. At the beginning we define the physical layer of SaCI, starting with the list of the different devices in SaCI, then we define the framework to represent the devices like agents, and finally we give an example of utilization of the framework in a device of SaCI. Next, we define the physical layer of SaCI following a similar procedure.

4.1 Agents of the AmI physical layer

4.1.1 Characterization of the devices of SaCI

In this section we present a list of the possible devices in a smart classroom like SaCI. These devices can be or not in a given moment in the environment. Table 1 presents this list, and for each device we define their capabilities, tasks in the learning process, their components, and their interactions with other components of SaCI.

4.1.2 Agent Model for the AmI physical layer

In this section we propose a general model (framework) to describe the different devices of SaCI (intelligent or not) like agents. This general model has general information of each device, and information about its tasks and intelligence (The framework is based on MASINA models [3], specifically the models of agent, task and intelligence). The framework is composed by the next templates, one initial template with the basic information of a device (see Table 2).

Because the middleware is based on the idea of services to be provided between the agents, the list of services given by the agents is very important. Now, we need to describe how the services are given by a device. For that, template in Table 3 describes the different tasks that must be executed in a device (physical agent) in order to give the set of services previously defined. For that, it is necessary fill out one task template for each task.

Device	Tasks	Components	Intelligent Capabilities	Interaction with other components of SaCI
Student Board	Represent the virtual student in the AmI	Video camera, screen, processor, wifi, bluetooth	Detect the presence of a student. Recognize and adapt to the mood of the student	Smartboard LVE
Smart board	Display learning contents Allow students - educational contents interaction	Touch support, Display high-resolution processor, wifi, bluetooth	Adapt educational contents to context	LVE Student Board Smart cameras Interactive desk
Smart camera	Show who speaks in the physical environment Follow the activities in the environment	Processor, wifi, bluetooth,	Follow active objects in the environment (moving, talking, etc.)	Smartboard Student Board
Interact desk	Show educational multimedia contents (photos, videos, etc.) Enrich contents	Processor, Display high-resolution wifi, bluetooth, stereo speakers, microphone, touch support	Adapt educational contents to users	Smartboard LVE
Social Robot	Recognize emotional states in the environment Motivate and arouse interest in activities to learn	Vision System; Recognition System of Face, Gesture and Speech; Music Player; Object Tracking; Obstacle Detection; etc.	Adapt to the context surrounding it	Smartboard LVE
Augment Reality System	Enrich contents	Processor, Video Beam	Adapt educational contents to users	Smartboard LVE

Table 1. List of the possible devices in SaCI

Finally, some of the devices can be intelligent. In this case, we must define an intelligent template (see Table 4) with the basic information about that.

ID device: the physical identification of the device, **Type device:** the type of device. **Name Agent:** the name of the device in the multiagent system. **Description:** describes the general characteristic of the devices. **Components**: a device can be composed by other devices. **Goal:** what is it the main function of this device in the educational environment. **Services:** list of services given by this device in the educational environment.

Table 2. Basic Information of a Device

Task Name: name of the task. **Objective:** goal of the task. **Description:** describes the task.

Table 3. Tasks Template

Learning mechanisms: if the device has learning algorithms. **Reasoning mechanisms:** if the device has reasoning algorithms. **Ontologies:** list of ontologies (domains, contextual) used by the device. **Historical Information:** if the device keep historical information. **Source:** where is stored the data used by the device (database, etc.).

Table 4. Intelligent Template

4.1.3 Example of Specification of a device of SaCI like Agent

In this section we show how to specify the SmartBoard like agent using our framework (See Tables 5, 6, 7 y 8).

ID device: SB001, **Type device:** Smart Board, **Name Agent:** Smart Board, **Description:** Board with intelligent capabilities to interact with users, etc. The Smart Board operates as part of a system that includes a interactive whiteboard, a computer, a projector and collaborative learning software for education. **Components**: whiteboard, a computer, a projector, touch system **Goal:** Students are able to work more collaboratively on a single workspace where the contents are shown. **Services:** display learning contents, allow the interaction students-contents

Table 5. Basic Information of a Smart Board

Task Name: Display learning contents **Objective:** allow to see the contents. **Description:** this task is for allow to show the educational contents and interact with them

Table 6.Task Template for the task "Display learning contents"

Task Name: Allow students - educational contents interaction **Objective:** allow the interaction among the students and the contents . **Description:** The dives need an interface (like a touch system) in order to allow the interaction amongo the contents showed and the students.

Table 7.Task Template for the task "Allow students - educational contents interaction"

Learning mechanisms: the device has learning algorithms in order to learn the behavior of the users. **Reasoning mechanisms:** the device can use the knowledge to reason about the . **Ontologies:** this devices need ontologies about the environment (contextual). The information about the users (profile, subject studied, etc.) are asked to other agents **Historical Information:** not apply. **Source:** the device has internal databases.

Table 8.Intelligent Template to Adapt educational contents to context

4.2 Agent of the AmI logical layer

4.2.1 Characterization of the software components of SaCI

In this section we present a list of some of the possible softwares in SaCI. These softwares can be or not in a given moment in the environment. Table 9 presents this list, and its characteristics.

Software	Tasks	Components	Intelligent Capabilities	Interaction with other components of SaCI
Virtual Learning Environment (VLE)	Manage educational contents Store information about the user's profile Enable tutors and students to exchange messages with other participants, teamwork, download educational resources, consult the teacher, etc.	Courses Educational resources Interactive tools	Adapting to the user profile Learning the user's learning style and adapt instruction to it.	Smartboard Smart cameras Interactive desks

Repository of learning objects	Manage educational resources in different formats.	Database Ontology to describe the learning objects Resources web server	Discovery new learning objects	VLE Smartboard
Recommender system of educational resources	Suggest educational resources to students according to their learning style Provide personalized answers to the user	Educational Resources Knowledge base	Provide resources adapted to the learning style	Repository of learning objects Smartboard VLE
Academic System	Register students performance	Curricula Student Records	Not apply	VLE Recommender system of educational resources

Table 9. List of the possibles softwares in SaCI

4.2.2 Agent Model of the AmI logical layer

Like the previous phase, we propose a framework to describe the different software of SaCI like agents. This framework has general information of each software, and its tasks and intelligent behavior, and like the previous framework is based on MASINA models [3]. The framework is composed by the next templates, one initial template with the basic information of a software:

Name software: the name of the software,
Software Type: the type of educational software (VLE), management, etc.,
Framework: if the software is based in a framework, etc.,
Name Agent: the name of the software in the multiagent system,
Description: describes the general characteristic of the software that is represented
Components: maybe this software can be composed in other subsystems
Goal: what is it the main function of this software in the educational environment.
Services: list of the services given by the software in the educational environment.

Table 10. Basic Information of a Software

Now, for each service we describe the tasks that must execute each software. For that, we use the same task template defined in the previous section (table 3). Finally, some softwares can be intelligent systems. The basic information about this aspect is defined in the same intelligent template defined in the previous section (table 4)

4.2.3 Example of Specification of a software of SaCI like Agent

In this section we show how to specify the VLE like agent using our framework.

Name software: Virtual Learning Environment (VLE)
Software Type: Educational management
Framework: NA
Name Agent: VLE,
Description: This system allows access to courses in which the student is enrolled and supports the teaching-learning process by providing educational and resources tools that are presented to the user based on their profile.

Components: Courses, Educational resources, Interaction tools, Video collaboration software, and other own interaction of VLE such as: internal mail, chat, forum, course management module, etc.

Goal: Facilitate student interaction with the main actors of the learning process: teachers, educational resources, and media.

Services: Manage educational contents, Store information about the student's profile, Enable tutors and students interaction.

Table 11. Basic Information of VLE

Now, we only specified its task to manage educational contents.

Task Name: Manage educational contents
Objective: it manages the sequences of presentation of the contents according to the curricula. Additionally, it uploads and updates documents in different formats.
Description: Each course has a sequences of educational contents to be used. This system controls the advances in the presentation of these contents. Additionally, this resources can be updated.

Table 12. Task Template to manage educational contents of VLE

Learning mechanisms: This software doesn't use learning algorithms
Reasoning mechanisms: This software doesn't has reasoning algorithms.
Ontologies: not apply
Historical Information: The VLE stores historical information on each course, such as, actions on the tools (forums, chat, messaging), reading and downloading documents, student score, consultations to the teacher, etc.
Source: Databases, academic system

Table 13.Intelligent Template of VL

5 Conversations in SaCI

A main aspect in a multiagent system is the capability of interaction between the agents. During the execution of a multiagent system there are several occasions where the agents need interact to reach one specific goal. Each set of interactions to reach a given goal in the system is called in MASINA conversation. We keep the same idea to describe the interactions in SaCI.

In this section we describe one of the conversations in SaCI, and how is specified using the conversation model and the interactions diagram proposed in MASINA. We will analyse the conversation linked to the online tutoring process in SaCI. We start defining the conversation model (see table 14).

Conversation Name: online tutoring process
Goal: help the students during their processes of learning
Agents: VLE, a smart board, a recommender system of learning resources, an academic system, a tutor, and several students.
Beginner: tutor and students
Precondition A new subject to be covered in the curriculum by the students.
End condition When the student finish the interaction with the resources of learning planned by the VLE and tutor for this session
Description This conversation describes the different activities on the AmI to support the online support to the students. These activities are carried out by different agents according to their roles. For example, VLE plans the learning resources to be used during the session, and monitors its utilization during it. Also, it demands to search the learning resources according to the students´ profile and the subject of the session.

Table 14. Example of conversation in SaCI

In the conversation are involved different types of agents of SaCI: one device type, and the rest of software type: VLE, students, etc. Figure 2 shows the interaction diagram of this conversation.

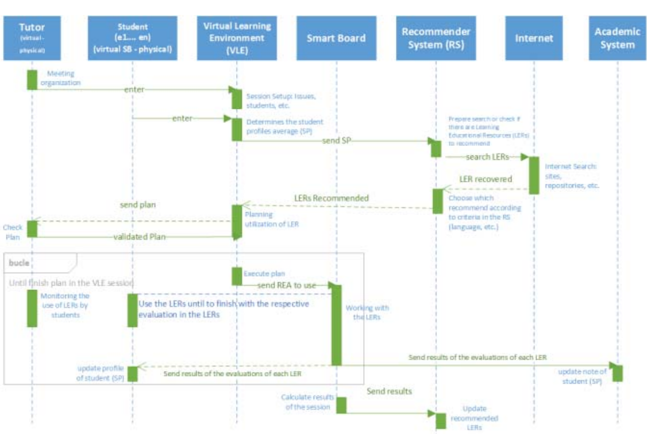


Figure 2. Conversation “online tutoring process in SaCI”

In Figure 1 the process of tutoring in SaCI is represented like a conversation between SaCI agents. SaCI adapts the online tutoring process to the requirements of a specific session. For that, VLE determines the subjects to learn and the students´ profile, and immediately asks an intelligent search (that is carried out by the recommender system) of learning resources, which are shown into the environment by the smart board according to the planning defined by VLE. Then, the students interact with these learning resources via the smart board, and VLE monitors the work of the students (normally, the learning resources have an evaluation phase before to finish with their utilization, that VLE must guarantee the students carry out). This is a cyclical process that is done in each tutoring session. At the end, VLE establishes a student’s score (evaluation), and updates the learning profile of the students in function of these results (learn) for the next session.

6 Conclusions

A flexible way to model AmIs is very important, in order to extend its utilization in different domains: home, classroom, museums, etc. This modelization must consider the main aspects on these environments: the intelligent capabilities of the components and the interactions between them. The multiagent paradigm supports very well these requirements of modelization.

But in order to model a smart classroom using agents, it is necessary define a middleware to support the community of agents that describe the AmI. The work defined in [2] solves this problem. In this way, the main aspect is to describe the different components of the AmI, and like they are represented on the middleware. That is the goal reached in this work for the case of a smart classroom.

Particularly, we see as SaCI can be described very easy based in our multiagent framework. Additionally, the dynamical behavior (described by the conversations) are also described very well. In this way, the components and the conversations in SACI are describe without problem.

In general, the scalability and flexibility properties of an AmI can be studied in our model, as well as more complex properties like the emergence and the self-organization will be able be analyzed in next studies using this modelization.

Our templates give the main information of the agents of software and hardware, but if more details are required in order to be implemented (for example, about the language of knowledge representation, the type of coordination in a conversation (for example, auction), etc.) the template can be extended using the models defined in MASINA.

Next work must analyse the emergence and the self-organization of SaCI using this model, and the specific problems of implementation (multiagent platform, communication protocols, etc.). The middleware defined in [2] is a good base to start this implementation.

Acknowledgments

Dr Aguilar has been partially supported by the Prometeo Project of the Ministry of Higher Education, Science, Technology and Innovation of the Republic of Ecuador. This work has been partially supported by the UTPL Project entitled: "Medios de Gestión de Servicios (Middleware) Inteligentes para Entornos de Aprendizaje Virtual".

7 References

[1] N. Winters, K. Walker et G. Roussos, «Facilitating learning in an intelligent environment,» chez *IEEE International Workshop on Intelligent Environments*, Colchester, UK, June 2005.

[2] M. Sánchez et J. Aguilar, «AmICL - Middleware Reflexivo para Ambientes Inteligentes de Aprendizaje en la Nube,» Mérida, Venezuela, June 2014.

[3] J. Aguilar, I. Besembel, M. Cerrada, F. Hidrobo et F. Narciso, «Una metodología para el Modelado de Sistemas de Ingeniería Orientado a Agentes,» *Inteligencia Artificial: revista iberoamericana de inteligencia artificial,* vol. 12, n° %138, p. 39–60, 2008.

[4] P. Mikulecký, «Smart environments for smart learning,» chez *Proceedings of the 9th International Scientific Conference on Distance Learning in Applied Informatics*, Sturovo, Slovakia, May 2012.

[5] D. Cook et D. Sajal, Smart Environments: Technology, Protocols and Applications, New Jersey: John Wiley & Sons, 2005.

[6] C. Ramos, G. Marreiros, R. Santos et C. F. Freitas, «Smart offices and intelligent decision rooms,» chez *Handbook of Ambient Intelligence and Smart Environments*, Springer US, 2010, p. 851–880.

[7] S. Yang, T. Okamoto et S.-S. Tseng, «Context-Aware and Ubiquitous Learning,» *Educational Technology & Society,* vol. 11, n° %12, pp. 1-2, April 2008.

[8] Y. Shi, W. Qin, Y. Suo et X. Xiao, «Smart classroom: Bringing pervasive computing into distance learning,» chez *Handbook of Ambient Intelligence and Smart Environments*, Springer, 2010, p. 881–910.

[9] F. Mhiri et S. Ratté, «AARTIC: development of an intelligent environment for human learning,» chez *Proceedings of the 14th annual ACM SIGCSE conference on Innovation and technology in computer science education (ITiCSE '09)*, Paris, France, July 2009.

[10] M. El-Bishouty, H. Ogata, S. Rahman et Y. Yano, «Social Knowledge Awareness Map for Computer Supported Ubiquitous Learning Environment,» *Educational Technology & Society,* vol. 13, n° %14, p. 27–37, 2010.

[11] S. Garruzzo, D. Rosaci et G. Sarné, «Isabel: A multi agent e-learning system that supports multiple devices,» chez *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, November 2007.

[12] X. Z. George Margetis, P. Koutlemanis, M. Antona et C. Stephanidis, «Augmented interaction with physical books in an Ambient Intelligence learning environment,» *Multimedia tools and applications,* vol. 67, n° %12, p. 473–495, 2013.

[13] S. Stenvall-Virtanen et K. Nordell, «Smart Environments: Technology, Protocols and Applications,» Finland, May 2014.

[14] J. Aguilar, A. Ríos, F. Hidrobo et M. Cerrada, Sistemas Multiagentes y sus aplicaciones en Automatización Industrial, 2nd éd., Mérida: Talleres Gráficos, Universidad de Los Andes, 2013.

Approaches and Strategies to Extract Relevant Terms: How are they being applied?

J. Valaski, S. Reinehr, and A. Malucelli

joselaine.valaski@pucpr.br,sheila.reinehr@pucpr.br, malu@ppgia.pucpr.br

PPGIa, Pontifícia Universidade Católica do Paraná, Curitiba, Paraná, Brazil

Abstract - *One of the goals of the term extraction is to identify and structure relevant information form texts. Despite advances in recent years, there are still several challenges for the development of efficient tools and methods related to this activity. Term extraction in specific domain knowledge is even a more complex issue, because sometimes you cannot rely on an initial vocabulary to support the extraction. Motivated by this question, an exploratory research to identify the approaches and strategies that are being applied to term extraction has been conducted. The main goal was to identify the most significant one. The research began with 174 proposals and finished with 25 that achieved the filter criteria. This research presents the main data related to these selected proposals.*

Keywords: Relevant Term, Term Extraction, Exploratory Research, Domain Knowledge.

1 INTRODUCTION

The recent years have witnessed a proliferation in unstructured text data [1]. In a world where the amount of digital data grows over more than 50% per year, any means to structure this data becomes increasingly relevant [2]. Difficulties arise when knowledge is contained in a textual format and no support is available. In these cases, techniques for automatic processing of a textual content are required [3].

In Natural Language Processing (NLP), several techniques exist for extracting terms from large collections of general and specific texts [1].

In general, these techniques are applied to extract relevant terms from texts. With these results it is possible to generate structured and formalized information. The goals of these techniques are diverse, such as, creating concept maps to assist in the learning process [4], building domain ontology [5], creating semantic model domain [6-7], extracting skills in job advertisements [8], improving search on the Web [9] and extracting important topics in a blog [10].

For the identification of relevant terms, distinct approaches and strategies can be applied, such as, linguistic, statistics and hybrid. Despite several proposals of tools and methods for the relevant term extractions, most extant term extraction algorithms are inadequate to address the challenges posed by domain-specific texts. A major challenge is the sparse nature of these texts, which do not offer reliable statistical evidence, and severely compromise the algorithms' performance [1].

The term extraction can be supported by an initial vocabulary or not. However, when you need to identify terms from a specific domain, in most cases, you cannot count on the support of this initial vocabulary. This restriction makes the development of algorithms with a good performance even more challenging.

In this context, it is considered relevant to obtain an overview of the proposals presented in this area. It is believed that with these results, it is possible to identify approaches and strategies that have been applied with greater relevance. These parameters can be used as criteria on the selection of tools and/or methods. Moreover, its parameters may support the improvement of existing tools and/or methods.

As it has not been found on scientific bases a work that presented these results in a systematic way, this research aimed to conduct this exploratory type of research method. The research was conducted through the analysis of scientific papers available at Scholar search site.

This research is structured as follows: Section 2 presents a brief theoretical framework about the most common approaches and strategies applied to the relevant terms' extraction. Section 3 describes in detail how the exploratory research was conducted. Section 4 presents and discusses the main results of this research. Section 5 concludes the research and presents the future work.

2 THEORICAL BACKGROUND

Several methods are available to extract terms from a set of documents. These methods can be broadly categorized into three different approaches: linguistic approaches, statistical approaches, and hybrid approaches [1][11].

2.1 Linguistic approach

Linguistic approach uses NLP for term extraction. Linguistic methods are often implemented as Part-of-Speech

(POS) filters, which accepts, as terms, any noun sequences containing optional adjectives and/or prepositions. A POS-tagger labels the part-of-speech of terms (e.g., adjective, noun, verb, etc.) appearing in a text. In general, there are three main techniques on this approach: syntactic, morphological and semantic analysis.

- Syntactic analysis: identifying the syntactic function of the word, such as noun, adjective and verb. POS-tagger tools are applied on this identification;
- Morphological analysis: derivation a term's form, e.g., whether a terms used in singular or plural form. The procedure lemmatization is used on this analysis. The lemmatization allows to group together in a single attribute the multiple morphological forms of words which have a common semantics; and
- Semantic analysis: identifying the meaning of words, normally obtained by means of an external base, for example, WordNet.

When extracting terms for a certain domain, they however do not consider the relevance of a term for that domain. Since linguistic methods rely on the syntactic structure, they identify terms according to the unithood property [5].

2.2 Statistical approach

Differently than the linguistic approaches, statistical methods do not use the linguistic characteristics of terms, but rely solely on statistical measures to extract terms [5]. These statistical methods are applied to acquire the relevance of a term for a domain.

Statistical approach concerning the termhood, it is statistically determined based on the observation that the highly frequent expressions in a domain specific corpus are likely to denote relevant terms. Another termhood estimation technique is that of corpus comparison, in which a domain-specific corpus is compared against a collection of general texts. Expressions that are more likely in the domain-specific corpus are then treated as domain-specific terms.

In general, the strategies on statistical approach are based on frequency with some variation: absolute frequency, frequency with comparison, frequency with weight and co-occurrence:

- Absolute frequency: count the absolute frequency of a term in a document.
- Frequency with comparison: count the frequency of a term in a document considering the frequency of a term in another document. An example is to use a domain-specific corpus to compare against a collection of general texts;
- Frequency with weight: count the frequency of a term in a document and apply distinct weight. As an example, a term that appears on the document title can have a bigger weight than a term that appears on the document body.
- Co-occurrence: count the frequency of two or more terms together. In this strategy the compound terms are considered more relevant than simple terms.

3 METHOD

An exploratory research was conducted to identify general and specific information applied to relevant terms extraction. This identification aimed to map the most relevant approaches and strategies.

The search was performed using the search engine Scholar. The Scholar was used because it enables the identification of diversified publications, such as, dissertations, theses, technical reports, articles, among others. The search on only renowned scientific bases, such as, Springer Link, Science Direct and IEEE, could limit the number of publications. This conclusion was obtained by a preliminary test.

The period between 2004 and 2014 has been defined for the selection of the publications. This criterion was established considering that this research does not aim to explore the evolution of all proposals for the extraction of relevant terms. The main goal was to identify the latest tools and methods related to this activity. It is believed that the period of 10 years is enough for this scenario.

The search string used was: "term extraction" and "tool" and "relevant term." The survey was conducted on 05/31/2014 and 174 publications were returned. The research was also performed without a date filter. In this scenario 212 publications were returned. This showed that the largest number of publications related to this topic, actually happened in the last 10 years. Thus, the filter was kept in date (2004-2014) and 174 publications were evaluated.

Among these 174 publications, patents, citations, books or files without access were identified. Thus, the publications without access were excluded, resulting in 96 publications with documents available for analysis. Among the 96 publications available, 52 were selected by reading the abstract. By reading these abstracts, it was feasible to identify the possibility of these proposals to be related to some tool or method applied to the extraction of relevant terms.

These 52 publications were read in full to ensure the application of some tool or method for the extraction of relevant terms. Furthermore, studies that depended on an initial vocabulary for term extraction were excluded. The goal of this research was to identify proposals that did not depend on an initial vocabulary to extract relevant terms. This restriction is important especially in specific domains where normally there is not an initial vocabulary to support this activity.

As a final result, 25 proposals were obtained, and the following information was extracted: year of publication, language of the processed text and tools applied on extraction approach. The results are shown below.

4 RESULTS AND DISCUSSION

The results are presented in two perspectives: general data and tools and the applied approaches and strategies. In the first perspective, information related to proposals origin and tools used are presented. In the second perspective, the approaches and strategies applied to relevant term identification are presented. The symbol UI (Unidentified Information) was used to indicate situations where information could be identified.

4.1 General data and tools

Table 1 presents the general data, such as: publication year, origin country of publication and idiom of the processed text.

Table 1. General data

Reference	Year	Country	Text idiom
[1]	2013	Netherlands	English
[3]	2011	Italy	Italian
[4]	2009	Brazil	Portuguese
[5]	2012	Italy	English
[6]	2010	Netherlands	English
[9]	2008	China	English
[10]	2012	South Korea	English
[11]	2014	Netherlands	English
[12]	2013	Iran	Farsi
[13]	2013	Brazil	Portuguese
[14]	2010	China	English
[15]	2008	Germany	English
[16]	2012	India	English
[17]	2012	Brazil	Portuguese
[18]	2005	Germany	English
[19]	2012	Spain	English Spanish
[20]	2013	Slovak Republic	**UI**
[21]	2004	China	Chinese
[22]	2013	Sweden	Swedish
[23]	2009	Netherlands Spain Italy	English Dutch Spanish
[24]	2012	Mexico	**UI**
[25]	2010	Brazil	Portuguese
[26]	2011	Tunisia	English
[27]	2013	Germany	English
[28]	2007	Austria	English

In order to obtain a better analysis, the data presented on Table 1 was summarized and is presented in Table 2. The data summarization allows some important information, which are discussed as follow.

From 2012 a bigger number of proposals related to the relevant term identification was observed. This could support the idea that the importance of this activity has increased in the last few years. It is important to emphasize that 2014 is a special period, because this research was performed in the first semester. Thus, it is expected that the number of publications is lower this year.

There is a big concentration of researches on European countries (Netherlands, Germany, Italy, Austria, Spain and Sweden). Among the analyzed proposals, 14 had European origin. In the American continent, Brazil is the highlight country and in the Asian continent, China is the highlight country.

Table 2. General data summary

Year	Qty	Country	Qty	Idiom	Qty
2004	1	Netherlands	4	English	15
2005	1	Brazil	4	Portuguese	4
2007	1	China	3	Spanish	2
2008	2	Germany	3	Chinese	1
2009	2	Italy	3	Swedish	1
2010	3	Spain	2	Farsi	1
2011	2	Austria	1	Italian	1
2012	6	South Korea	1	Dutch	1
2013	6	India	1	**UI**	2
2014	1	Iran	1		
		Tunisia	1		
		Mexico	1		
		Slovak Republic	1		
		Sweden	1		

English is the main language used to process texts. Among 25 analyzed proposals, 15 used a text written in English in order to evaluate the extraction instruments of relevant terms. The bigger the amount of applied works in English language is, it results in more significant advances in tools and methods applied to this language, than to any other.

Table 3 presents data related to applied tools. In this table, for each proposal the tools applied are presented in order to: perform term extraction, linguistic annotation (e.g., parser, tagger, dependency relationship) and other support (e.g., documentation indexing, n-gram extraction, summarization and semantic indexing).

Also in order to make a better analysis possible, the data showed on Table 3 was summarized and are presented in Table 4. The data summarization allows some important information, which is discussed as follows.

Nine different proposals of tools applied to relevant term extraction were found. However, only “EXATO LP” was found in more than one distinct proposal. However, it is important to emphasize that both proposals that used “EXATO LP,” belong to the same research group. This result shows several tools but each one of them with its own approach. This can be seen more clearly in the following results.

It is important to emphasize that 15 proposals did not present a tool in order to perform relevant term extraction. Only an algorithm was applied, as it can be observed in the next results as well.

Table 3. Applied tools

Reference	Tools		
	Term extraction	**Linguistic Annotation**	**Other**
[1]	ExtTerm	Stanford	**UI**
[3]	**UI**		
[4]	**UI**		
[5]	Extractor (SAOD)	**UI**	Lucene
[6]	**UI**	FreeLing2 Alpino	**UI**
[9]	**UI**	Qtag	**UI**
[10]	**UI**		
[11]	ATCT	Stanford	**UI**
[12]	**UI**	Bijankhan	**UI**
[13]	**UI**		
[14]	**UI**	Survey parser	**UI**
[15]	SProUT	MINIPAR	Lucene
[16]	**UI**		
[17]	EXATO LP	PALAVRAS LX-center	**UI**
[18]	**UI**	Genia YamCha-Chunker	**UI**
[19]	**UI**	**UI**	Ngram statistics package
[20]	**UI**		
[21]	**UI**		
[22]	IPhraxtor	Connexor machinese syntax	**UI**
[23]	Tybot	**UI**	**UI**
[24]	**UI**		
[25]	EXATO LP	PALAVRAS	NSP tool
[26]	**UI**	**UI**	TextTiling LSI
[27]	ATExTA	**UI**	**UI**
[28]	Protégé plugin	**UI**	**UI**

These results may confirm the idea of the recent evolution of relevant term extraction. Many algorithms have been proposed but there not enough tools able to integrate the best algorithms results.

It has been also observed that several proposals used linguistic annotation to support the relevant term extraction. Twelve distinct tools have been identified. Only “PALAVRAS” and “Stanford” tools have been applied in more than two proposals. Moreover, documentation indexing, summarization and n-gram extraction tools to support the relevant term extraction have also been identified.

The main results obtained by these data were as follows: there is relevant application of linguistic annotation tools to identify features that enable the relevant terms identification; there are several proposals without a supportive tool, and among the proposals that use a tool, the approaches are very diversified.

Table 4. Tools summary

Term extraction	Qty	Linguistic Annotation	Qty	Other	Qty
EXATO LP	2	PALAVRAS	2	Lucene	2
ATCT	1	Stanford	2	Ngram statistics package	1
ATExTA	1	YamCha-Chunker	1	NSP tool	1
SProUT	1	Connexor machinese syntax	1	TextTiling	1
Protégé plugin	1	FreeLing	1	LSI	1
IPhraxtor	1	Genia	1		
Tybot	1	LX-center	1		
ExtTerm	1	MINIPAR	1		
Extractor	1	Survey parser	1		
UI	15	Bijankhan	1		
		Alpino	1		
		Qtag	1		

Due to the approach diversification, it was considered important to identify the ones that are being applied with more emphasis. In order to obtain this result, for each proposal, approaches and strategies applied were identified. The results are presented as follows.

4.2 Applied approaches and strategies

For each analyzed proposal, with the use of tools or not, applied approaches and strategies to relevant term extraction were identified. Table 5 presents these results. The approach and strategies presented in Section 2 to classify the proposals were used.

Linguistic approach

Among the 25 analyzed proposals, 17 distinct proposals that applied linguistic approach to relevant term extraction were identified. Considering the linguistic approach, the syntactic analysis was the most used one. Using the syntactic analysis strategies, 11 proposals [1][9][11-12][14-15][17-18][22-23][25] suggest the noun term identification as a main feature to relevant term extraction. Some of these proposals also suggest the adjective [1][12][15], verb [1] and preposition [1][12] term identification.

The proposals that used morphological analysis, in a general manner, had the goal to prepare the term to syntactic analysis. Applying the lemmatization process to reduce the word to its root, can be cited as an example. Only two proposals were identified considering the semantic analysis [6][23].

Statistical approach

All the 25 proposals applied strategies from statistical approach. The co-occurrence [1][6][9][11-12][16][18-21][24-25] and frequency with comparison [1][3][5][9-11][13][15][17][20][28] were the most used strategies from the statistical approach. The most highlighted co-occurrence strategy were: techniques to identify n-grams [9][19][24-25], MI (Mutual Information) metric [1][6][21] and C-Value metric [12].

In the frequency with comparison strategy, the highlighted metrics were as follows: TF-IDF (Term Frequency - Inverse Document Frequency) [3][9-10][20][28], TF-DCF (Term Frequency - Disjoint Corpora Frequency) [13][17], KF-IDF [15] and IG (Information Gain) [5].

Despite being a simple strategy, the absolute frequency was also applied in some proposals [4][22-23][26-27]. The frequency with weight was applied as follow: type of structure where the term appears, such as title [11]; typed of syntactic structure where the term appears [14], type of annotation performed by the user [20], and the position of a candidate term in the hierarch, the hypernym relations between candidate terms [23].

Hybrid approach

Among the 25 analyzed proposals, 17 applied strategies combined with linguistic and statistical approach. These results show that the relevant term extraction is more efficient if distinct strategies are applied together.

Considering the results obtained from the exploratory research, the following strategies were identified as the most relevant:

- Using of syntactic analysis to consider a noun;
- Applying the TF-IDF metric; and
- Identification of co-occurrence to consider a compound term.

All of these 3 strategies can be used together to rank the terms. The position that the term appears can be used to measure the relevance.

Among all proposals analyzed, only 3 [1][9][11] applied the 3 strategies identified as most relevant. In [1], a framework called ExtTerm oriented to term extraction is proposed. In [9] the strategies to improve the search on the Web are proposed. Finally in [11] a framework called ATCT to automatic build domain taxonomy from texts is proposed. In all proposals there is not availability of a tool to public use.

Table 5. Applied approaches and strategies

Reference	Approaches/Strategies						
	Linguistic			Statistical			
	Morphological	Syntactic	Semantic	Absolute frequency	Frequency with comparison	Frequency with weight	Co-occurrence
[1]		X			X		X
[3]	X				X		
[4]				X			
[5]					X		
[6]	X		X				X
[9]		X			X		X
[10]					X		
[11]		X			X	X	X
[12]		X					X
[13]					X		
[14]		X				X	
[15]		X			X		
[16]							X
[17]	X	X			X		
[18]		X					X
[19]							X
[20]	X				X	X	X
[21]							X
[22]		X		X			
[23]		X	X	X		X	
[24]							X
[25]		X					X
[26]	X			X			
[27]	X			X			
[28]	X				X		
Total	**7**	**11**	**2**	**5**	**11**	**4**	**12**

The obtained results give guide on the features to be considered to choose a tool to relevant term extraction. Through broader search, using the search engine Google, it was possible to identify free tools available to relevant term extraction. Most of these tools are not mentioned in the proposals analyzed in this exploratory research.

An evolution of this research could be a survey of the tools that meet the strategies indicated as the most relevant.

5 CONCLUSION AND FUTURE WORK

The application of techniques to relevant term extraction is a crucial activity to transform non-structured texts in structured and formalized information. In order to support this activity, there are several tool, methods and approaches.

Despite the diversity of proposals, we have not found a systematic map presenting how these instruments are being applied. We believe that this overview is important to support the decision on what tools and methods to choose. Moreover, this overview can suggest the improvement of existing tools and methods. This research had the goal of performing this systematic map.

The main obtained results were: the hybrid approach applications that bonds syntactic and statistical approach were the most relevant ones. The strategies that privilege the noun identification, compound term and use TF-IDF metrics are the most significant.

This survey was used to define the most efficient instruments to relevant term extraction. These results are being applied on an environment to support the semiautomatic build of ontological conceptual model. The next step is to integrate selected tools in order to obtain relevant term extraction on specific domain in this environment.

REFERENCES

[1] Ittoo, A., and Bouma, G. Term extraction from sparse, ungrammatical domain-specific documents. Expert Systems with Applications, v, 40, n. 7, p. 2530–2540, 2013.

[2] IDC, The digital universe decade — are you ready?, from http://www.emc.com/ collateral/demos/microsites/idc-digital-universe/iview.htm 2010.

[3] Amato, F., Mazzeo, A., and Scippacercol, S. A method for the evaluation of the peculiar lexicon significance. Electronic Journal of Applied Statistical Analysis: Decision Support Systems and Services Evaluation, v. 2, n. 1, p. 54–64, 2011.

[4] Dalmolin, L. C. D., Nassar, S. M., Bastos, R. C., and Matheus G. P. A Concept Map Extractor Tool for Teaching and Learning. In *9th IEEE International Conference on Advanced Learning Technologies*, p.18–20, 2009.

[5] Eynard, D., Matteucci, M., and Marfia. A modular framework to learn seed ontologies from text. Semi-Automatic Ontology Development: Processes and Resources, 2012.

[6] Bosma, W., and Vossen, P. Bootstrapping Language Neutral Term Extraction. In 7th Language Resources and Evaluation Conference (LREC), 2010.

[7] Valaski, J., Reinehr S., and Malucelli, A. Environment for Requirements Elicitation Supported by Ontology-Based Conceptual Models: A Proposal. In Proceedings of the 2014 International Conference on Software Engineering Research and Practice (SERP'14), ISBN 1-60132-286-0, Las Vegas, USA, p. 144-150, 2014

[8] Reichhold, M., Kerschbaumer, J., Fliedl, G., and Winkler, C. Automatic Generation of User Role Profiles for Optimizing Enterprise Search. In ICSSEA, v. 43, p. 1-8, 2012.

[9] Chen, T., Chang, P., and Teng, W. Supporting Informational Web Search with Interactive Explorations. In IEEE International Conference on Signal Image Technology and Internet Based Systems, p.153–60, 2008.

[10] Park, J., Lee, S., Jung, H., and Lee, J. Topic word selection for blogs by topic richness using web search result clustering. In Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication - ICUIMC, 2012.

[11] Meijer, K., Frasincar, F., and Hogenboom, F. A semantic approach for extracting domain taxonomies from text. Decision Support Systems, v. 62, p. 78–93, 2014.

[12] Behin-Faraz, H., Passban, P., and Shokrollahi-Far, M. A reliable linguistic filter for Farsi term extraction. The 5th Conference on Information and Knowledge Technology, p. 328–31, 2013.

[13] Fernandes, P., Furquim, L. O. C, and Lopes, L. A Supervised Method to Enhance Vocabulary with the Creation of Domain Specific Lexica. In IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), p. 139–42, 2013.

[14] Zhang, X., and Fang A. C. An ATE system based on probabilistic relations between terms and syntactic functions. In 10th International Conference on Statistical Analysis, 2010.

[15] XU, F. Bootstrapping Relation Extraction from Semantic Seeds. Thesi, 2008.

[16] David, M. R., and Samuel, S. Clustering of PubMed abstracts using nearer terms of the domain. Bioinformation, v. 8, n 1, p. 20–25, 2012.

[17] Lopes, L., Vieira R., Fernandes, P., and Couto, G. Exatolp: extraction of language resources from Portuguese corpora. In International Conference on Computational Processing of the Portuguese Language – PROPOR, p. 45–47, 2012.

[18] Wermter, J., and Hahn, U. Finding new terminology in very large corpora. In K-CAP '05 Proceedings of the 3rd international conference on Knowledge capture, p. 137–44, 2005.

[19] Vàzquez, M., and Oliver, A. Improving Term Candidate Validation Using Ranking Metrics. In 3rd World Conference on Information Technology, v. 3, p. 1348–1359, 2013.

[20] Harinek, J., and Simkom M. Improving term extraction by utilizing user annotations. Proceedings of the 2013 ACM symposium on Document engineering - DocEng, 2013.

[21] Pu, H., and Chien, L. Integrating log-based and text-based methods towards automatic web thesaurus construction. In Proceedings of the American Society for Information Science and Technology, v. 41, n. 1, p. 463–471, 2005.

[22] Merkel, M., Foo, J., and Ahrenberg, L. IPhraxtor - A linguistically informed system for extraction of term

candidates. In Proceedings of the 19th Nordic Conference on Computational Linguistics, p. 121–132, 2013.

[23] Bosma, W., Vossen, P., and Soroa, A. KAF: a generic semantic annotation format. In Proceedings of the GL2009 Workshop on Semantic Annotation, 2009.

[24] Vieyra, S., Suárez-Figueroa, M. C., Estrada, H., and Martinez, A. knOWLearn: a reuse-based approach for building ontologies in a semi-automatic way. 2012.

[25] Lopes, L., Oliveira, L. H., and Vieira, R. Portuguese term extraction methods: Comparing linguistic and statistical approaches. In Proceedings of the 9th PROPOR, p. 1–6, 2010.

[26] Chabi, A. H., Kboubi, F., and Ahmed, M. B. Thematic analysis and visualization of textual corpus. International Journal of Computer Science & Engineering Survey, v. 2, n. 40, 2011.

[27] Houy, C., Sainbuyan, K., Fettke, P., and Loos, P. Towards automated analysis of fads and trends in information systems research: Concept, implementation and exemplary application in the context of business process management research." In IEEE 7th International Conference on Research Challenges in Information Science (RCIS), p. 1–11, 2013.

[28] Pammer, V. Two Protégé plug-ins for supporting document-based ontology engineering and ontological annotation at document-level. In 10th International Protégé, p. 1–3, 2007.

Construction of the EEG Emotion Judgment System Using Concept Base of EEG Features

Mayo Morimoto[1]**, Misako Imono**[2]**, Seiji Tsuchiya**[2]**, and Hirokazu Watabe**[2]
[1]Department of Information and Computer Science, Graduate School of Science and Engineering, Doshisha University, Kyo-Tanabe City, Kyoto, Japan
[2]Department of Intelligent Information Engineering and Science, Faculty of Science and Engineering, Doshisha University, Kyo-Tanabe City, Kyoto, Japan

Abstract - *For a robot converse naturally with a human, it must be able to accurately gauge the emotional state of the person. Techniques for estimating emotions of a person from facial expressions, intonation and speech content have been proposed. This paper presents a technique for judging the emotion of a person using EEGs. The system of judging the emotion from EEGs is called EEG Emotion Judgment System, and constructed Concept Base for reducing noise in this paper. Accuracy of emotion judgment using EEG features of all subjects was 57.6% and using leave-one-out cross validation was 30.8%. Although performance accuracy remains low, continued development is required through further development of methods for both reducing different variety of noise mixed in with EEGs.*

Keywords: EEG, judging emotion, concept base

1 Introduction

For a robot to converse naturally with a human, it must be able to accurately gauge the emotional state of the person. Techniques for estimating emotions of a person from facial expressions intonation and speech content have been proposed. Currently, EEGs are attracted attention for the tool of estimating emotion. EEGs are the electrical signals from brain, and control people's expression. It has advantage for possibility of reading without direct processing.

This paper presents a technique for judging the emotion of a person by differing between EEGs. The system of judging the emotion from EEGs is called EEG Emotion Judgment System. The noises are easily included in EEGs when judging emotions and are difficult to remove out. This system considered for noise by constructed Concept Base to reduce the influence of noises.

2 Overview of Proposed Technique

The objective of this technique was to read the emotions of a conversation partner from EEGs. Figure 1 shows outline of the proposed technique of emotion judgment from EEGs which is called EEG Emotion Judgment System.

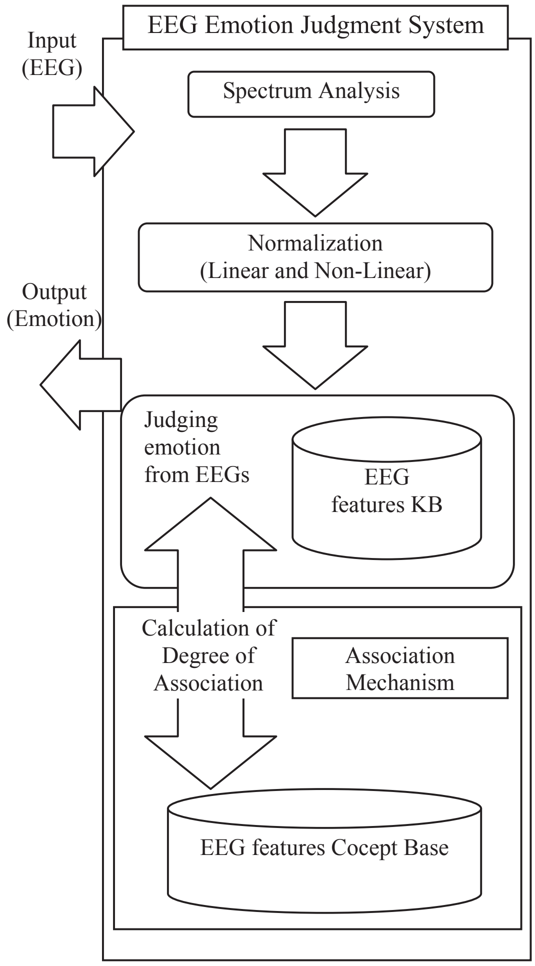


Fig.1 EEG Emotion Judgment System

EEGs acquired from the subject, and are used as source EEGs. Emotions of the subject are acquired simultaneously as source EEGs. Emotions have been assigned to spectrum analysis of the source EEGs and are performed every 1.28 second. The EEG features are determined to θ waves (4.0 Hz to 8.0 Hz), α waves (8.0 Hz to 13.0 Hz), and β waves (13.0 Hz to 30.0 Hz) which showed at Figure 2.

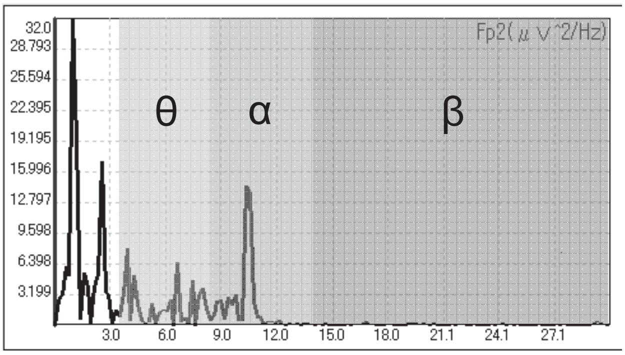


Fig.2 Spectrum analysis of the source EEGs

Emotion is judged from EEG features by an Association Mechanism. The EEG features association was realized by using the huge Concept Base [1, 2] which was automatically built from EEG features. A method to calculate the Degree of Association [3, 4, 5] evaluate the relationship between EEG features. Concept Base and Degree of Association are used at natural language processing, however, the structure of EEG features and the word was similar so applied the technique at EEGs. Hereafter, this Concept Base and the calculation method are called the “Association Mechanism”. Emotions judged in thus study pleasure, anger, sadness, and no emotion.

3 Acquisition of Source EEGs and Emotions

EEGs were measured at 14 locations that their positions conforming to the International 10-20 system showed at Figure 3 [6].

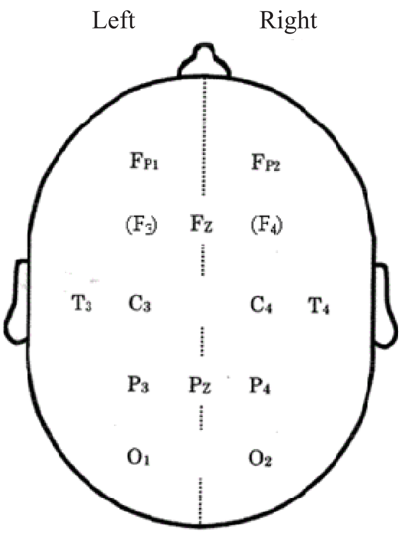


Fig.3 14 locations to measure EEG conforming to the International 10-20 system

Subjects fitted with an electroencephalography [7] caps, showed at Figure 4. The subjects were asked to watch Japanese film for approximately two hours. During watching film, subjects were asked to gauge the emotions each felt by speech in the film, and source EEGs were acquired simultaneously. Scene in the film were frozen for each of speech in the film, and the subject were asked what emotion they felt at that time watching the scene.

Fig.4 Image of Electroencephalography

Eighteen subjects (nine males and nine females) were used, and viewing was divided into four sessions to reduce the physical burden to subjects. Before and after the film, EEGs of open-eye and closed-eye states were measured for approximately one minute each for use in normalization of EEG features.

4 Normalization of EEG Features

EEGs show changes in voltage intensity over time within an individual, and base voltage intensity differs among individuals. For this reason, the possibility of misjudgment exists because those values differ greatly even among EEGs with similar waveforms. To solve this problem, linear normalization and non-linear normalization were performed.

4.1 Linear Normalization

This was performed to take into account for voltage intensity of EEGs varies over time depending on the subject. Since the eyes were open while viewing the film, linear normalization was performed to acquire EEGs by both before and after experiment based on EEG features from the eye-open state.

EEG features $Linear\_al_{ij}$, obtained by linear normalization of first EEG feature $alij$ at a certain point in time during the experiment, and is expressed by Formula 1:

$$Linear\_al_{ij} = al_{ij} + \left\{ \left(\frac{q_1 - q_2}{p_2 - p_1} \times l + q_2 \right) - \left(\frac{q_2 - q_1}{p_2 - p_1} \times l + q_2 \right) \right\} \Big/ 2 \quad (1)$$

4.2 Non-linear Normalization

This was performed to take into account for the difference among individuals in base voltage intensity.

Non-linear normalized values were obtained by using Formula 4.2. *f(x)* is the EEG features after non-linear normalization has been applied, x is the EEG features applied in non-linear normalization, x_{min} is the minimum EEG features of individual, and x_{max} is the maximum EEG features of individual. As a result, EEG features with large values are compressed and EEG features with small value are expanded by non-liner normalization. Thus, the degree of voltage intensity of an individual's EEGs is solved.

$$f(x) = \frac{\log(x - x_{\min})}{\log(x_{\max} - x_{\min})} \tag{2}$$

5 EEG Features Knowledge Base

EEG features Knowledge Base is data base constructed by EEGs and emotions. EEG features of 42 represented by three bandwidths which obtain from 14 locations are assumed to one EEG data, and each are matched with emotions, either anger, sadness, no-emotion, nor pleasure. EEG features Knowledge Base is consisted with each knowledge base of each emotion. EEG features Knowledge Base containing 2887 EEGs obtained by excluding outliers and noise from the total of 5670 EEGs. The emotions of the 2887 EEGs are comprised 541 anger features, 726 sadness features, 1226 no-emotion features, and 394 pleasure features. The image of EEG features Knowledge Base is showed in Figure 5.

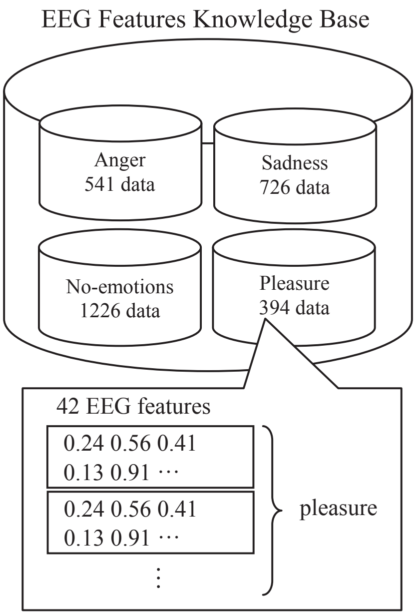


Fig.5 Image of EEG Features Knowledge Base

6 Association Mechanism

The Association Mechanism consists of the Concept Base and the Degree of Association. The Concept Base generates semantics from a certain EEG features. The Degree of Association is used for the semantics expansion, and it expresses the relationship between one EEG feature and another by a numeric value. The methods of a Concept Base and a Degree of Association were proposed in the field of the natural language processing, and the research results are apply to EEGs written in this paper.

6.1 Concept Base of EEG Features

First of all, a Concept Base of words is explained that is in the field of the natural language processing. The research results are applied to EEG in this paper.

A Concept Base is a large-scale database that is constructed both manually and automatically using words from multiple electronic dictionaries. The entry word in dictionary used as concepts and independent words in the explanation under the concept are used as an attributes. In the current research, a Concept Base containing approximately 90,000 concepts, in which auto-refining processing has been done after the base had been manually constructed. In this processing, attributes considered by the standpoint of human sensibility, that the inappropriate attributes were deleted and necessary attributes were added.

In the Concept Base, Concept A is expressed by Attribute a_i indicating the features and the meaning of the concept in relation to a Weight w_i, denoting how important an Attribute a_i is in expressing the meaning of Concept A. Assuming that the number of attributes of Concept A is N, Concept A is expressed by Formula 3 at below. Here, the Attribute a_i are called Primary Attribute of Concept A.

$$A = \{(a_1, w_1), (a_2, w_2), \cdots, (a_N, w_N)\} \tag{3}$$

By the reason of Primary Attribute a_i of Concept A is defined as the concepts in the Concept Base, attributes can be similarly elucidated from a_i. The Attributes a_{ij} of a_i are called the Second Attributes of Concept A. Attribute a_i is defined by a_{ij}, and also defined as the concepts, so a_{ij} is Primary Attribute of a_i. Thus, Concept Base can be connecting to N-dimension. Figure 6 shows the elements of the Concept "train" expanded as far as the Secondary Attributes.

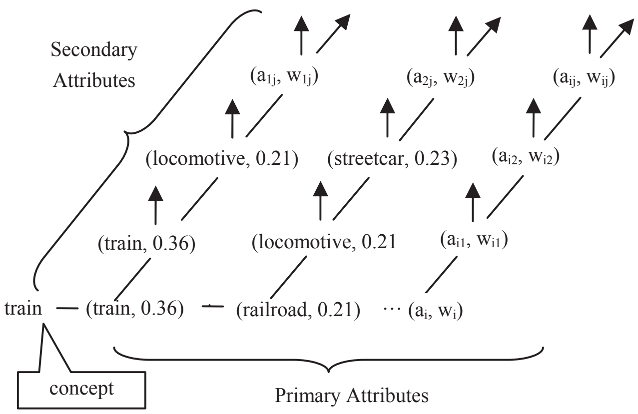


Fig.6 Example of demonstrating the Concept "train" expanding as far as Secondary Attributes

In this study, a Concept Base was made by using source EEGs instead of electronic dictionaries. In fact, EEG features are used instead of words.

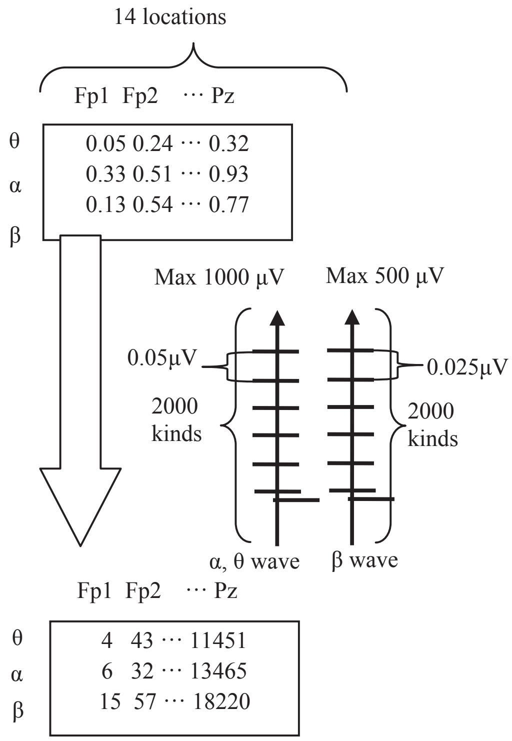


Fig.7 Process for conceptualization of EEG features

The voltage value in each location and each bandwidth is considered to be an each part of the word respectively. However, the granularity of the voltage value is more detailed than that of the word. Therefore, the voltage value of the certain scope is treated as the same as showing in Figure 7. As a result, the number of EEG features are controlled, similar to words are controlled by synonym. The concrete method is to delimit the θ waves and α wave by 0.05μV, and delimit β waves by 0.025μV. Furthermore, different numerical values in each divided group are allocated same part and bandwidth. This numerical value is treated as a word, and part and bandwidth are new information than in natural language processing. As a result, EEG features can be conceptualized similar to a Concept Base of words. In the present research of natural language processing, 20,252 concepts were used in a Concept Base.

6.2 Weight

Weight is performed by the method of *TF・IDF*. *TF・IDF* is popularly used in the field of natural language processing for searching information. Weight *W (A, B)* of Concept *A* for attribute *B*, is calculated as follows:

$$W(A,B) = tf(B) \times \log_2 \frac{D}{df(B)} \tag{4}$$

Concept *A* is one EEG feature of 42 EEG features, and remained 41 EEG features are considered as an attribute. EEG features in the same group, considered in figure 7 are also added to an attributes. As the premises, *tf(B)* express the frequency that come out within all attribute of Concept *A*. *D* is a number of concepts stored in EEG features Concept Base, and *df(f)* is a number of concepts to have Concept *A* included in each attribute. *idf* is calculated to divide *D* by *df(B)*, having logarithm as 2 for bottom. Thus, *tf ・ idf* is calculated as multiple *tf* by *idf.* The image of concept and attribute are showed at Figure 8.

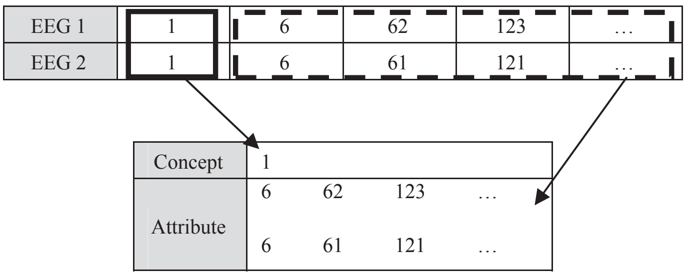


Fig. 8 Image of concept and attribute

6.3 Calculation of Degree of Association for EEG Features

First of all, a Calculation of Degree of Association is explained that is in the field if the natural language processing, calculating by words.

6.3.1 Degree of Match by Weight Ratio

Degree of Match by Weight Ratio is calculated by total value of each 42 EEG feature's degree of match. Regard input EEG as Concept *A*, and EEG in EEG features Knowledge Base as Concept *B*. Each EEG features of

Concepts are regarded as *A'* and *B'*, and attribute are defined as a_i and b_i. Weight of *A'* and *B'* are defined as u_i and v_i. If the numbers of attributes are *L* and *M* respectively to the concepts ($L \leq M$), they can be expressed as follows:

$$A' = \{(a_1, u_1), (a_2, u_2), \cdots, (a_L, u_L)\} \quad (5)$$

$$B' = \{(b_1, v_1), (b_2, v_2), \cdots, (b_M, v_M)\} \quad (6)$$

Electrodes part *p*, frequency *f*, and EEG features *e* are defined by a_L and b_M, showed by follows:

$$a_L = (p_L, f_L, e_L) \quad (7)$$

$$b_M = (p_M, f_M, e_M) \quad (8)$$

Due to this, Degree Match by Weight Ratio *DoM(A, B)* of Concept *A* and *B* is calculated as follows:

$$DoM(A', B') = \frac{(S_{A'}/n_{A'} + S_{B'}/n_{B'})}{2} \times \frac{\min(u_i, v_i)}{\max(u_i, v_i)} \quad (9)$$

$$S_{A'} = \sum_{a_i = b_j} u_i \qquad S_{B'} = \sum_{a_i = b_j} v_i \quad (10)$$

$$n_{A'} = \sum_{i=1}^{L} u_1 \qquad n_{B'} = \sum_{j=1}^{M} v_j \quad (11)$$

$$\min(u_i, v_j) = \begin{cases} u_i (u_i \leq v_j) \\ v_j (u_i > v_j) \end{cases} \quad (12)$$

$$\max(u_i, v_j) = \begin{cases} u_i (u_i > v_j) \\ v_j (u_i \leq v_j) \end{cases} \quad (13)$$

$a_i = b_j$ is expressed when attribute matches. $S_{A'}$ is the total weight of a_i when $a_i = b_j$ matched, and $S_{B'}$ is the total of b_j when $a_i = b_j$ matched. $n_{A'}$ and $n_{B'}$ is the total weight of Concept *A* and *B*. Thus, $S_{A'}/n_{A'}$ is a ratio of weight that matched to attribute from look from Concept *A*, and $S_{B'}/n_{B'}$ is a ratio of weight that matched to attribute from look from Concept B. Therefore, $(S_{A'}/n_{A'} + S_{B'}/n_{B'})/2$ express average of $S_{A'}/n_{A'}$ and $S_{B'}/n_{B'}$. Degree of Match by Weight Ratio is calculated by considering ratio of coincidence by attribute and weight.

6.3.2 Degree of Association by Weight Ratio

This paper applied Degree of Association by Weight Ratio, considers the coincidence of attribute and weight. For calculation, input EEG data and EEG features Knowledge Base data, such as para-concept *A* and *B*, use para-concept *A'* for standard to fix the row. Para-concept is a one block of concept, so EEG features in this block becomes first attribute. Then, sort the attribute in para-concept *B* for making the total of the degree of match largest between para-concept *A*. The attribute and weight of para-concept *B* is defined as (b_{xi}, v_{xi}). The electrode part p_{xi}, frequency band f_{xi}, and EEG features e_{xi} are defined by b_{xi}.

$$B' = \{(b_X, v_{X1}), (b_{X2}, v_{X2}), \cdots, (b_{X42}, v_{X42})\} \quad (14)$$

$$b_{Xi} = (p_{Xi}, f_{Xi}, e_{Xi}) \quad (15)$$

In this study, EEG features of 42 represented by three bandwidths which obtain from 14 locations are assumed to the attributes, and are used by the calculation of the Degree of Association.

7 Calculation Experiment

7.1 Experiment Method

The method of evaluation was a leave-one-out cross-validation, a technique which involves using one data extracted as the validation set and the remaining observations are as the training set for comparison. In this study, EEG features Knowledge Base containing 2887 EEGs obtained by excluding outliers and noise from the total of 5670 EEGs. The emotions of the 2887 EEGs are comprised 541 anger features, 726 sadness features, 1226 no-emotion features, and 394 pleasure features in this study.

The evaluation was performed for two proposed technique. The difference of two techniques is using different EEG features Knowledge Base. First proposed technique uses EEG features Knowledge Base include all EEG features of 18 subjects. Comparison to first technique, second proposed technique uses EEG features Knowledge Base, excluding one subject that used for evaluation. That is to say, EEG features Knowledge Base is constructed as the number of the subjects. Thus, second proposed technique used 18 EEG features Knowledge Base for the evaluation. Image of EEG features Knowledge Base of proposed technique showed in Figure 9.

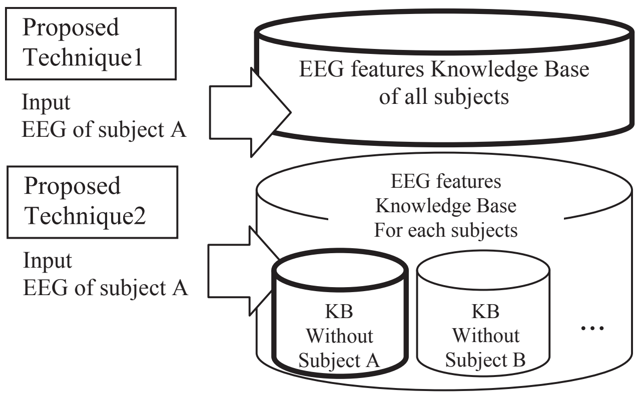


Fig.9 Image of EEG features Knowledge Base

7.2 Evaluation of Accuracy

The result of the emotion judgment form EEGs are showed in Figure 10.

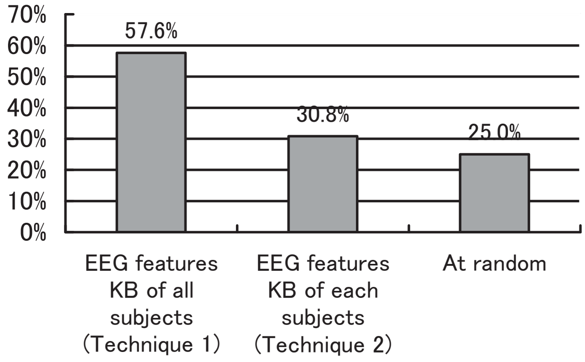


Fig.10 Result of the emotion judgment from EEGs

Accuracy of emotion judgment from EEGs using the Association Mechanism was 30.8%. As a comparison, accuracy used EEG features Knowledge Base that includes EEGs of all the subjects was 57.6%, accuracy of using EEG features Knowledgebase of each subject is 30.8%, and accuracy of emotion judgment at random was 25.0%..

8 Discussion

Accuracy that includes EEGs of all the subjects was a highest accuracy in the evaluation. The reason is EEGs of same subject as validation sets are in EEG features Knowledge Base so it influenced to the judgment of emotion. Then, the accuracy using EEG features Knowledge Base of each subject was 30.8%, and was higher than 25% at the random. From these result, we believe that the proposed method is effective

9 Conclusion

Authors are conducting research for new emotion judgment system by using EEG features. Especially, authors focus on constructing EEG features Concept Base and using the Degree of Association, to reduce noise in the EEGs.

EEGs acquired from the subject are used as source EEGs. Spectrum analysis of the source EEGs have assigned emotion flags, and performed every 1.28 second. This determined as the EEG features. Emotion is judged from EEG features by an Association Mechanism. The Association Mechanism consist the Concept Base and the Degree of Association. The Concept Base generates semantics from a certain EEG features, and the Degree of Association uses the result of the semantic expansion. The Degree of Association expresses the relationship between one EEG features and another as calculating a numeric value. The methods of a Concept Base and a Degree of Association were proposed in the field of the natural language processing, and their research result were applied to EEGs.

As a result, accuracy of EEG Emotion Judgment System used EEG features Knowledge Base of all subjects was 57.6%, and accuracy of using EEG features Knowledge Base of each subject is 30.8%. As comparison, accuracy of emotion judgment at random was 25.0%. By the result, we believe that the proposed method was effective.

10 Ackowledgements

This research has been partially supported by the Ministry of Education, Science, Sports, and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

11 References

[1] K. Kojima, H. Watabe, T. Kawaoka, "A Method of a Concept-base Construction for an Association System: Deciding Attribute Weights Based on the Degree of Attribute Reliability", Journal of Natural Language Processing, Vol.9, No.5, pp.93-110, 2002.

[2] N. Okumura, E. Yoshimura, H. Watabe, and T. Kawaoka, "An Association Method Using Concept-Base", KES2007/WIRN2007, Part I, LNAI4692, pp.604-611, 2007.

[3] H. Watabe and T. Kawaoka, "The Degree of Association between Concepts using the Chain of Concepts", Proc. Of SMC2001, pp.877-881, 2001.

[4] T. Hirose, H. Watabe, and T. Kawaoka, "Automatic Refinement Method of Concept-base Considering the Rule between Concepts and Frequency of Appearance as an Attribute", Technical Report of the Institue of Electronics, Information and Communication Engineers, NLC2001-93, pp.109-116, 2002.

[5] H. Watabe, E. Yoshimura, and S. Tsuchiya, "Degree of Association between Documents using Association Mechanism", KES2009, LNAI5711, pp.243-250.

[6] J.Clin, Guideline thirteen: guidelines for standard electrode position nomenclature, American Electroencephalographic Society, Neurophysiol 11, pp.111-113, 1994.

[7] T. Musha, Y. Terasaki, H. A. Haque, G. A. Iranitsky "Feature extraction from EEG associated with emotions", Art Life Robotics, pp.15-19.

Investigation of CI forecasting algorithms for short-time cash demand in ATM network

G. Žylius, V. Vaitkus, and R. Simutis
Department of Automation, Faculty of Electrical and Electronics Engineering, Kaunas University of Technology, Kaunas, Lithuania

Abstract - *Good ATM network cash management requires accurate information of future cash demand. In this paper we compare computational intelligence models when performing cash flow forecasting for one day. Adaptive input selection and model parameter identification are used with every forecasting model in order to perform more flexible comparison. Experimental data contains 200 ATMs from real ATM network with historical period of 26 months. Investigation of historical data length influence for forecasting accuracy with every model is also performed. Results suggest that v-SVR (support vector regression) forecasting model performs best when SMAPE forecasting accuracy measure is used.*

Keywords: Computational Intelligence, Cash Flow, One-Step-Ahead Forecasting.

1 Introduction

In order to optimize cash distribution in ATM network an estimation of cash demand in the future must be done. So cash demand forecasting accuracy determines overall performance of cash management system. Good cash management for ATM network brings savings for retail banks that are related to: 1) dormant cash reduction; 2) reduced replenishment costs; 3) decrease of cash preparation costs; 4) reduction of cash insurance costs.

Various uncertain factors influence cash demand in ATM that makes this process hard to forecast: nonlinear trend and seasonal component mixtures with non-stationary heteroscedastic uncertainty. However approximate empirical relationship between input and output variables can be obtained using complex data-based regression or time series models.

In this research we apply and compare data-based computational intelligence (CI) regression models for one day ahead cash demand forecasting. The dataset used consist of 200 ATMs.

This paper is further divided into following sections: 1) literature review (a review of existing methods applied for cash demand forecasting); 2) forecasting models (specification of each forecasting model used); 3) experimental data and methodology (short explanation of methodology used for forecasting and experimental data specifications); 4) results (analysis of forecasting results); 5) conclusions and future works.

2 Literature review

Process of cash demand in ATM is highly affected by holidays, seasonal and calendar effects [1]. These effects can be incorporated into classical neural network models that could be used for cash demand forecasting [2]. However cash demand varies in time, so more flexible approach [3] to incorporate neural networks for cash demand in ATM is needed. Advantage of popular support vector machines over neural networks applied for regression was experimentally denied [4] when cash demand forecasting with long historical period training data was performed. However, for data with shorter history support vector regression may be more effective. An interval type-2 fuzzy neural network (IT2FNN) applied [5] for cash demand forecasting is another approach that adapts to dynamic nature of ATM cash flow and (as author claims) is better than other systems based on time series.

Despite CI model applications, time series models are also used for cash demand forecasting problem. Researches show that SARIMA models among classical econometric models perform best [6] and even outperform joint forecasting approach using vector time series models [7]. Authors in [8] made a comparison between time series probability density forecast models: linear models, autoregressive models, structural time series models and Markov-switching models. Results showed that Markov-switching models performed best.

Other researchers use cash demand forecasting approaches that deeper investigate cash demand process (beyond aggregated data-based empirical relationships). Random-effects models [9] [10] were used to model individual cash withdrawal patterns for ATM withdrawal forecasting. Researcher in [11] treats intraday cash flow time series as random continuous functions projected onto low dimensional subspace and use functional autoregressive model as predictor of cash flow and intensity of transactions. ATM clustering approach was employed by [12] when integrated forecasting of aggregation of nearby-location ATM cash demand was performed.

3 Forecasting models

3.1 Support vector regression

Support vector machine (SVM) originally proposed in [13] is linear model that is used both for classification and regression. SVM application for regression is called support vector regression (SVR). Main idea of SVM is to map input data vectors into high dimensional feature space by using kernel functions. By doing so, linear nature of SVM model can be applied to nonlinear function approximation. This type of mapping is called *kernel trick*. In this research we use two types of SVR: 1) ν-support vector regression (ν-SVR) and 2) least squares support vector regression (LSSVR).

3.1.1 ν-SVR

Given the set of data points such that $\boldsymbol{x}_i \in R^n$ is an input vector (i-th observation n-dimensional vector) $y_i \in R^1$ is a target output, the optimization problem for ν-SVR algorithm is formulated by following equations [14]:

$$\min_{\boldsymbol{w},\varepsilon,\xi,\xi^*,b} \left\{ \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\left[\nu\varepsilon + \frac{1}{l}\sum_{i=1}^{l}\left(\xi_i + \xi_i^*\right)\right]\right\}$$
$$\text{subject to} \begin{cases} \left[\boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b\right] - y_i \le \varepsilon + \xi_i, \\ y_i - \left[\boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b\right] \le \varepsilon + \xi_i^*, \\ \xi_i, \xi_i^* \ge 0, \\ \varepsilon \ge 0. \end{cases} \quad (1)$$

Where $\phi(\boldsymbol{x}_i)$ is kernel function that performs mapping of input space to high dimensional feature space (the space where linear regression is performed); $\boldsymbol{w}$ is a parameter vector of n-dimensional hyperplane; b is hyperplane bias parameter; ξ_i^*, ξ_i are upper and lower training errors (slack variables) subject to ε – insensitive tube; C is a cost parameter, that controls the trade-off between allowing training errors and forcing rigid margins; ν is regularization parameter that controls parameter number of support vectors; l – is number of data points (observations). Data points that lie on the boundaries of ε – insensitive tube are called *support vectors*.

In this research we use ν-SVR code that is implemented in *LIBSVM* library (see [14]).

3.1.2 LSSVR

Least squares support vector regression optimization problem is formulated by following equations [15]:

$$\min_{\boldsymbol{w},b,e} \left\{ \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{\gamma}{2}\sum_{i=1}^{l} e_i^2 \right\}$$
$$\text{subject to} \begin{cases} \boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b + e_i = y_i, \\ \gamma \ge 0. \end{cases} \quad (2)$$

Where e_i are error variables and γ is regularization constant.

In this research we use LSSVR code from *LS-SVMlab* toolbox presented in [16].

Differently from ν-SVR, LSSVR has no insensitive tube and is only regularized by one parameter (γ). The loss function is quadratic for LSSVR and so the sparseness property for LSSVR is lost. For both ν-SVR and LSSVR in this research we use Gaussian kernel function with one dispersion parameter that needs to specify (σ). So for ν-SVR there will be total three parameters (ν, C, σ) and for LSSVR two parameters (γ, σ) to specify.

3.2 Relevance vector regression

Relevance vector regression [17] is model that has same linear functional form as support vector regression:

$$y(\boldsymbol{x};\boldsymbol{w}) = \sum_{i=1}^{M} w_i K(\boldsymbol{x},\boldsymbol{x}_i) + w_0. \quad (3)$$

Where $K(\boldsymbol{x},\boldsymbol{x}_i)$ is defined as kernel function and $\boldsymbol{w}$ is model weight vector.

Despite that SVR (except for LSSVR) is sparse model, RVR uses even less support vectors (is more sparse) that are called *relevance vectors* because of Bayesian inference methodology that is used during model parameter and relevance vector determination. RVR uses EM-like (expectation-maximization) learning algorithm and applies *a priori* distributions (because of Bayesian methodology) over parameters without need to be specified with some external parameters by user. With RVR we also use Gaussian kernel as in both LSSVR and ν-SVR cases.

In this research we use RVR implemented in *SparseBayes* package by author himself (see [17]).

3.3 Feed-forward neural network

We also apply feed-forward neural network (FFNN) models for cash demand forecasting with logarithmic sigmoid transfer functions in the hidden layer and linear transfer function in the output layer. For neural network training we use two backpropagation [18] – based algorithms: 1) Levenberg – Marquardt backpropagation [19]; 2) Levenberg – Marquardt backpropagation with Bayesian regulation [20].

Levenberg – Marquardt backpropagation training algorithm is frequently used as the most effective training algorithm for function approximation (regression) problems and uses mean-squared-error (MSE) cost function. This algorithm employs Jacobian matrix (**J**) w.r.t. network weights. An update of network weights using network output error (e) is calculated by formula:

$$\Delta\mathbf{w} = -\frac{\mathbf{J}^{\mathrm{T}}e}{\mathbf{J}^{\mathrm{T}}\mathbf{J}+\mu\mathbf{I}} \quad (4)$$

Where **I** is the identity matrix.

The main idea of this algorithm is to interpolate between Gauss – Newton and gradient descend algorithms by controlling scalar value μ.

Levenberg – Marquardt backpropagation with Bayesian regularization updates weights and bias values according to Levenberg – Marquardt optimization, but it minimizes a linear combination of squared errors and weights. The weight term in loss function doesn't let network to overfit and it corresponds to Bayesian regularization.

Both Levenberg – Marquardt backpropagation and Levenberg – Marquardt backpropagation with Bayesian regularization use validation set as stopping criterion in order to speed up training process. As parameters we use number of neurons in hidden layer. For forecasted value estimation we use average ensemble of neural networks, because of random weight initialization during training.

In this research we use *MATLAB Neural Network Toolbox* for FFNN forecasting.

3.4 Generalized regression neural networks

Generalized Regression Neural Network (GRNN) [21] is special case of radial basis function (RBF) neural network. It‘s first layer is the same as for radial basis neural network, but second layer is different.

GRNN does not require an iterative training procedure (error back propagation). Training procedure requires only specification of radial basis function spread parameter. It uses radial basis functions to cover input space and approximates function as weighted linear combination of radial basis functions. Number of RBF function is equal to number of observations. Each RBF is formed for each data point vector that is a center of RBF. RBF transfer function values are calculated according to input value Euclidean distance from the central point.

In this research we use GRNN implemented in *MATLAB Neural Network Toolbox*.

3.5 Adaptive neuro-fuzzy inference system

Adaptive neuro-fuzzy inference system (ANFIS) [22] is a combination of neural network and fuzzy inference system features. ANFIS model architecture with two membership functions is depicted in Fig. 1. ANFIS architecture has fuzzy layer (1), product layer (2), normalization layer (3), defuzzification layer (4) and summation layer (5).

For a 1^{st} order of Sugeno fuzzy model, a typical ***IF-THEN*** rule set can be expressed as:

1) ***IF*** *x is* A_1 ***AND*** *y is* B_1 ***THEN*** $f_1 = p_1x + q_1y + r_1$;
2) ***IF*** *x is* A_2 ***AND*** *y is* B_2 ***THEN*** $f_2 = p_2x + q_2y + r_2$.

Further each of five layer functionality is shortly explained:

1 layer. Forms output, which determines membership degree in each of membership functions (μ_{A1}, μ_{A2}, μ_{B1}, μ_{B2}):

$$O_{1,i} = \mu_{A_i}(x),\ i = 1, 2, \tag{5}$$

$$O_{1,i} = \mu_{B_{i-2}}(y),\quad i = 3, 4. \tag{6}$$

2 layer. In this layer each node is fixed and represents weight of particular rule. In each node ***AND*** operation is performed, which is product of inputs:

$$O_{2,i} = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y),\quad i = 1, 2. \tag{7}$$

3 layer. Each node of this layer is also fixed and calculates normalized rule excitation degree:

$$O_{3,i} = \overline{w}_i = \frac{w_i}{w_1 + w_2},\quad i = 1, 2. \tag{8}$$

4 layer. This layer is not fixed as other and parameters (p_i, q_i, r_i) are estimated during training process. Output of nodes are calculated as:

$$O_{4,i} = \overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i). \tag{9}$$

5 layer. This is an output layer, where output value is calculated as a sum of all inputs:

$$O_{5,1} = \sum_i \overline{w}_i f_i = \frac{\sum_i \overline{w}_i f_i}{\sum_i w_i}. \tag{10}$$

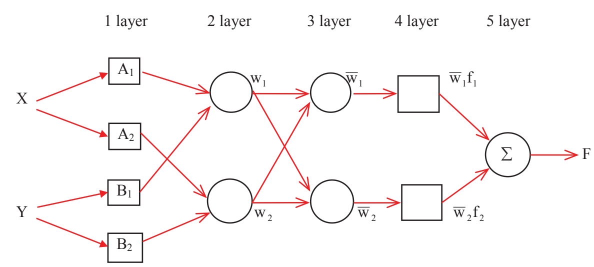


Fig. 1. Illustration of ANFIS model architecture with two membership functions.

ANFIS model training is usually performed using two training algorithms: gradient steepest descend backpropagation or hybrid algorithm. Hybrid learning combines gradient descend backpropagation and least squares methods. Backpropagation is used for parameters that are related with input membership functions, while least squares is applied for output function parameters (that are linear w.r.t. parameters). In this research we use both training algorithms and select best using validation set.

For membership function parameter initialization we use FCM (fuzzy c-means) clustering that extract set of rules that model input data behavior. So for ANFIS model number of clusters is parameter that we also select using validation set.

In this research we use ANFIS model that is implemented in *MATLAB Fuzzy Logic Toolbox.*

3.6 Extreme learning machines

Extreme learning machines (ELM) [23] are another type of single-hidden feed-forward neural networks that randomly chooses hidden nodes and analytically determines the output weights. Main advantage of this kind of learning over traditional backpropagation learning used in neural networks is speed.

Given N number of observations ($\boldsymbol{x}_i$, y_i), single layer neural network output with M hidden nodes is modeled as:

$$\mathrm{o_i} = \sum_{j=1}^{M} \beta_j \mathrm{f}(\mathrm{w_j x_i} + \mathrm{b_j})\,. \tag{11}$$

Where $\boldsymbol{x}_i$ is ith input vector; $\boldsymbol{w}_j$ is weight vector connecting the jth hidden node and the input nodes; β_j is weight scalar connecting jth hidden node and output node; b_j is bias parameter of jth hidden node.

In this research we use linear output nodes and sigmoid hidden nodes. Above equation can be written in vector form:

$$\mathrm{o} = \mathrm{H}\beta. \tag{12}$$

Where $\mathbf{H}$ is $N \times M$ hidden layer output matrix and $\mathbf{H}_{i,j} = f(\boldsymbol{w}_j\boldsymbol{x}_i + b_j)$.

The solution of applying extreme learning machines theory is simply estimated as:

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\boldsymbol{o}\,. \tag{13}$$

Where $\mathbf{H}^{\dagger} = (\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}$ is Moore – Penrose generalized inverse (pseudoinverse) matrix.

Because of speed of ELM we use larger average ensemble than with FFNN models to estimate forecasted value. As a parameter estimated using validation set for ELM we use number of neurons in hidden layer.

A *MATLAB* implementation of classical ELM is used in this research, which is available at webpage (see [23]).

4 Experimental data and methodology

Experimental data consist of 200 ATM real world daily cash demand time series, for historical period equal 26 months. Historical data period used for training varies from 6 months to 2 years (6, 12, 18 and 24 months) and forecasting one day ahead is performed for two months for each of 200 ATM daily cash demand time series. 10-fold cross-validation procedure is used for input and parameter selection with training set for every forecasting model. As forecasting accuracy measure we use symmetric mean absolute percentage error:

$$SMAPE = \frac{100}{N}\sum_{i=1}^{N}\frac{|\hat{y}_i - y_i|}{0.5(|\hat{y}_i| + |y_i|)}\,. \tag{14}$$

Where $\hat{y}_i$ is ith predicted value and y_i ith true value.

For all CI models following inputs were used: 1) week number; 2) day of the month; 3) cash flow value one day before $(i-1)$; 4) cash flow value 7 days before $(i-7)$; 5) cash flow value 14 days before $(i-14)$; 6) cash flow value 21 day before $(i-21)$; 7) cash flow value 28 days before $(i-28)$; 8) sum of cash flow values for last 5 days. All those 8 inputs were categorized into four groups with following input sets: $\{1, 2\}$; $\{3\}$; $\{4, 5, 6, 7\}$; $\{8\}$ (it was decided to do so in order to save computational time and preliminary experiments showed that this way of categorizing is reasonable). All those four groups were used for feature selection (when using 10-fold cross-validation with parameter selection) concluding to 15 different feature set combinations (using binomial formula $\sum_{k=1}^{n}\binom{n}{k} = 2^n - 1 = 2^4 - 1 = 15$) for reduced number of feature selection, whereas considering each of 8 features separately would conclude to 255 different feature combinations which was not accepted for practical purposes.

The example of cash demand time series is depicted in Fig. 2. This illustration shows the complexity of ATM cash flow demand and seasonality patterns: mixture of amplitude varying yearly, monthly and weekly seasonality (it is seen from autocorrelation function) including nonlinear trend that varies for different ATMs and also nonstationary noise which represents uncertainty degree in the cash demand process.

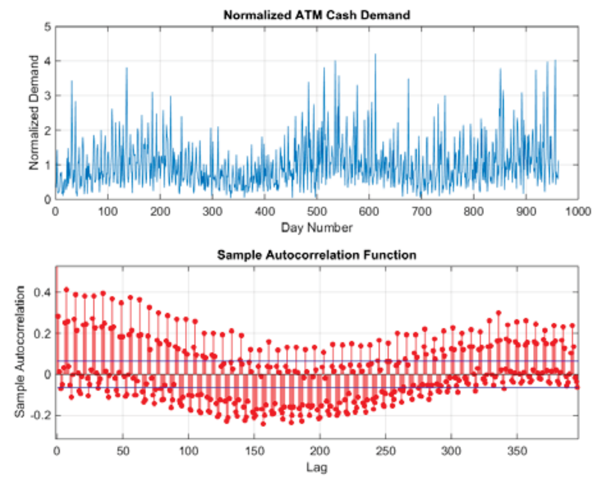


Fig. 2. An example of ATM cash demand.

5 Results

Overall forecasting results are presented in Table 1. It is seen that on average most accurate model is ν-SVR with 2 year training/validation dataset. Also results suggest that using 2 years of historical daily data yields best forecasting results over all models. However, interesting result is that using 18 month (1,5 year) history yields slightly less accurate forecasting results than using 12 month (1 year). This suggests that there was some disturbance event in the history that significantly affected data generating process. However as expected 6 month (0,5 year) data history significantly worsens forecasting accuracy of every forecasting model.

Table 1. Forecasting mean SMAPE (%) results

-	2 year training	1,5 year training	1 year training	0,5 year training	Avg.
ANFIS	44,12	44,24	44,17	45,71	44,56
ELM	44,18	44,20	44,00	45,95	44,58
LMBR-FFNN	44,03	44,13	44,26	45,74	44,54
LM-FFNN	43,87	44,07	44,10	45,61	44,41
GRNN	44,14	44,29	43,97	45,08	44,37
LS-SVR	43,85	44,05	43,88	45,28	44,27
ν-SVR	**43,72**	43,73	44,12	45,62	44,30
RVR	44,98	45,12	45,12	45,94	45,29
Avg.	44,11	44,23	44,20	45,62	-

Fig. 3 illustrates SMAPE distribution forecasting results using 2 year historical period for every model

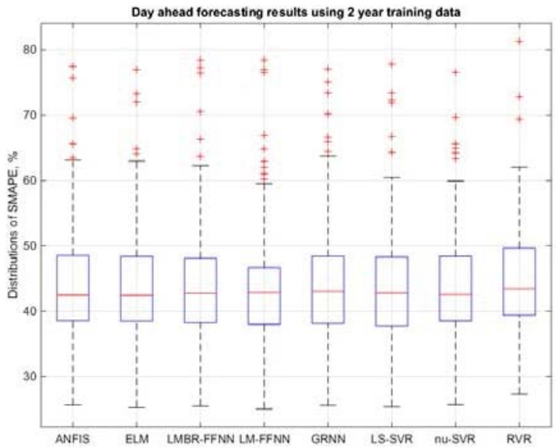


Fig. 3. SMAPE distribution for each forecasting model (2 year training case).

Table 2 show what percentage of all 200 ATMs every model gave best forecasting results. For example 10% show that model gave best forecasting results among all models for 20 ATMs. The results show that *ν*-SVR most often yield best accuracy for all training cases.

Table 2. Percentage of how often each model was most accurate for every training case separately.

-	2 year training	1,5 year training	1 year training	0,5 year training
ANFIS	9,0%	7,0%	15,0%	13,5%
ELM	8,5%	11,0%	10,5%	6,0%
LMBR-FFNN	11,0%	9,5%	10,5%	10,5%
LM-FFNN	14,5%	11,5%	11,0%	11,5%
GRNN	16,0%	15,0%	13,5%	10,5%
LS-SVR	8,0%	7,0%	11,5%	16,0%
***ν*-SVR**	21,0%	28,5%	18,5%	19,0%
RVR	12,0%	10,5%	9,5%	13,0%

Fig. 4 show the relationship between ATM average (median) cash demand and forecasting accuracy averaged over all models. Illustration clearly confirms the aggregation advantage: uncertainty approaches minimum (forecasting increases) as aggregation of more population values (aggregation of larger cash demand amounts) takes place.

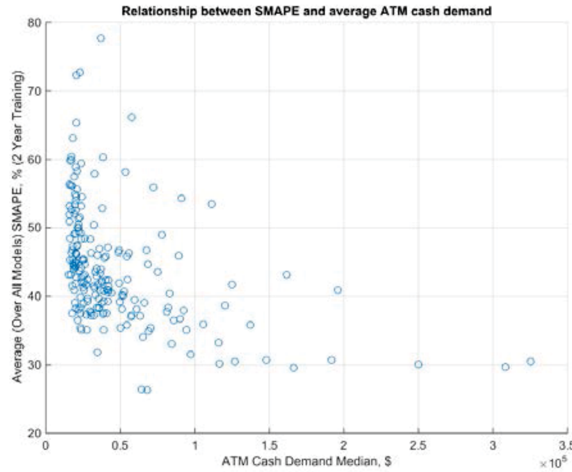


Fig. 4. Relationship between forecasting accuracy (averaged over all models) and ATM cash demand median (2 year training case)

6 Conclusions and future works

One day ahead forecasting results show that *ν*-SVR performs best compared to other models when using adaptive input selection. Also results confirm that using longer history can increase forecasting accuracy. However the relationship between historical data length and average forecasting accuracy was not smooth, showing that 1 year historical data period sometimes is better than using 1,5 year historical data. This suggests that some important factors affecting data generating process took place and deeper investigation of those factors (that can be related to structural breaks) is needed.

In the future works we are planning to investigate more deeply joint forecasting approach, when cash demand information of few or more ATMs is integrated or aggregated to increase forecasting accuracy. Also this study only contained one day ahead forecasting, when more often multiple step ahead forecasting is needed. We also aim to investigate multiple step ahead forecasting strategies with CI models more deeply in the future.

7 References

[1] Rodrigues, P. & Esteves, P. (2010). Calendar effects in daily ATM withdrawals. *Economics Bulletin*, *30*(4), 2587–2597.

[2] Kumar, P. & Walia, E. (2006). Cash forecasting: and application of artificial neural networks in finance. *International Journal of Computer Science & Applications, 3*(1), 61–77.

[3] Simutis, R., Dilijonas, D., Bastina, L. & Friman, J. (2007). A flexible neural network for ATM cash demand forecasting. In M. N. Katehakis, D. Andina & N. Mastorakis (Eds.), *Proceedings of the sixth WSEAS international conference on computational intelligence, man-machine*

systems and cybernetics (CIMMACS 07) (pp. 162–165), Tenerife, Spain: WSEAS.

[4] Simutis, R., Dilijonas, D. & Bastina, L. (2008). Cash demand forecasting for ATM using neural networks and support vector regression algorithms. In L. Sakalauskas, G. W. Weber & E. K. Zavidskas (Eds.) *Proceedings of the twentieth EURO mini conference on continuous optimization and knowledge-based technologies (EurOPT-2008)* (pp. 416–421), Neringa, Lithuania: VGTU press Technika.

[5] Darwish, S. M. (2013). A methodology to improve cash demand forecasting for ATM network. *International Journal of Computer and Electrical Engineering*, *5*(4), 405–409.

[6] Gurgul, H. & Suder, M. (2013). Modeling of withdrawals from selected ATMs of the Euronet network. AGH Managerial Economics, 13, 65–82.

[7] Wagner, M. (2010). Forecasting daily demand in cash supply chain. *American Journal of Economics and Business Administration*, *2*(4), 377–383.

[8] Brentnall, A. R., Crowder, M. J. & Hand, D. J. (2010b). Predictive-sequential forecasting system developement for cash machine stocking. *International Journal of Forecasting*, *26*, 764–776.

[9] Brentnall, A. R., Crowder, M. J. & Hand, D. J. (2008). A statistical model for the temporal pattern of individual automated teller machine withdrawals. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *57*(1), 43–59.

[10] Brentnall, A. R., Crowder, M. J. & Hand, D. J. (2010a). Predicting the amount individuals withdraw at cash machines using a random effects multinomial model. *Statistical Modeling*, *10*(2), 197–214.

[11] Laukaitis, A. (2008). Functional data analysis for cash flow and transactions intensity continuous-time prediction using Hilbert-valued autoregressive processes. *European Journal of Operational Research*, *185*, 1607–1614.

[12] Ekinci, Y., Lu, J. C. & Duman E. (2015). Optimization of ATM cash replenishment with group-demand forecasts. *Expert Systems with Applications*, *42*, 3480–3490.

[13] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

[14] Chih-Chung, C. & Chih-Jen, L. (2011). LIBSVM : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, *2*(27), 1–27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[15] Suykens, J. A. K., Van Gestel, T., De Brabanter, J., De Moor, B., Vandewalle, J. (2002). *Least Squares Support Vector Machines*. Singapore: World Scientific.

[16] Pelckmans, K., Suykens, J. A. K., Van Gestel, T., De Brabanter, J., Lukas, L., Hamers, B., De Moor, B., Vandewalle, J. (2002). LS-SVMlab : a Matlab/C toolbox for Least Squares Support Vector Machines. *Internal Report 02-44, ESAT-SISTA*. KU Leuven (Leuven, Belgium).

[17] Tipping, M. E. (2001). Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, *1*, 211–244. Software available at http://www.miketipping.com/sparsebayes.htm.

[18] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). Learning representations by back-propagation errors. *Nature*, *323*, 533–536.

[19] Hagan, M. T., & Menhaj, M. (1994). Training feed-forward networks with the Marquardt algorithm*.* *IEEE Transactions on Neural Networks*, *5*(6), 989–993.

[20] Dan Foresee, F. & Hagan, M. T. (1997). Gauss-Newton approximation to Bayesian learning. In *Proceedings of the 1997 International Joint Conference on Neural Networks* (pp. 1930–1935).

[21] Specht, D. F. (1991). A general regression neural network*.* *IEEE Transactions on Neural Networks*, *2*(6), 568–576.

[22] Jang, J.-S. R. (1993). ANFIS: adaptive-network-based fuzzy inference systems. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(3), 665–685.

[23] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, *70*, 489–501. Software available at http://www.ntu.edu.sg/home/egbhuang/elm_codes.html.

The DNA of Snakes

Md. Shahnawaz Khan[1] **and Walter D. Potter**[2]
[1,2]Institute of Artificial Intelligence, University of Georgia, Athens, Ga, United States

Abstract - *The Snake in the Box problem is an NP-Hard problem. The goal is to find the longest maximal snakes (a certain kind of path satisfying particular constraints described as "spread") in an n-dimensional hypercube [8]. With increasing dimensions the search space grows exponentially and the search for snakes becomes more and more difficult. This article identifies an underlying pattern among the known longest snakes in previously searched dimensions, which resembles the DNA of living cells in many ways. Surprisingly, these generic structures are fundamentally different for the four combinations of odd and even dimension and spread. It briefly explains the reason why they have different underlying structures. In odd dimensions with odd spread, there is one symmetric point and a unique mapping of complementary transition pairs and are discussed in detail in this paper. This article focusses only on one of these – odd dimension with odd spread. Later, it also reports three new lower bounds that are established using these generic structures from previously known longest maximal snakes. Another known longest snake in another odd dimension with odd spread is also found using this approach.*

Keywords: Snake-in-the-box, Generalization, DNA of snake, new lower bound, higher spread, longest maximal snake

1 Introduction

A snake is a special type of path in a graph (an n-dimensional hypercube) which does not violate its distance constraint described using the concept of "spread". Spread, being a concept of distance, is a non-negative number, and generally starts for spread k equal to 2. For spread 0, it has no meaning as technically it makes no contribution to the constraint. For spread 1, it simply requires a non-overlapping path traversing the n-dimensional hypercube and could be seen very similar to a Traveling Salesman Problem (often used as a standard problem in current AI literature). For spread 2 onwards it starts getting trickier and more computationally intensive to find such paths. The snake refers to the specific sequence of nodes in a graph and the edges joining these nodes form the path. While traversing it maintains the constraint that if the distance between any two nodes along the path is less than or equal to the spread then the shortest distance (Hamming distance) between them is equal to this distance along the path. For example, if node 0 and node x are placed like 0, _, x (where "_" could be any other node and node "x" is constrained) in a spread 2 or higher spread snake, then node x has to be a node which is exactly 2 Hamming distance away from node 0 (i.e. node x differs from node 0 in exactly 2 bits). If the distance between the two nodes along the path is greater than the spread then the shortest distance between these two nodes is greater than or equal to the spread. For example, if node 0 and node x are placed like 0, _, _, _, x in a spread k snake (for spread $k \leq 3$) then node x has to be a node which is at least k Hamming distance from node 0 (i.e. node x differs with node 0 in at least k bits). The maximally longest snake refers to the longest snake that can be found in a particular dimension-spread and cannot be grown further. So a path through nodes 0, 1, 3, 7, 6 would be the longest maximal snake of length 4 (distance between first node and last node in the path) in dimension 3 with spread 2.

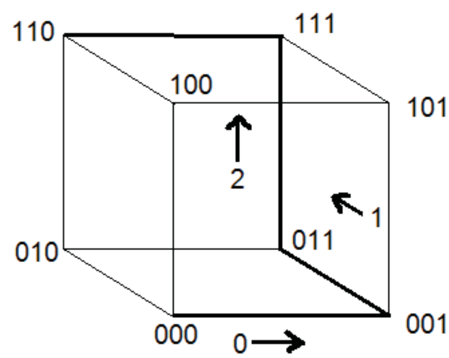


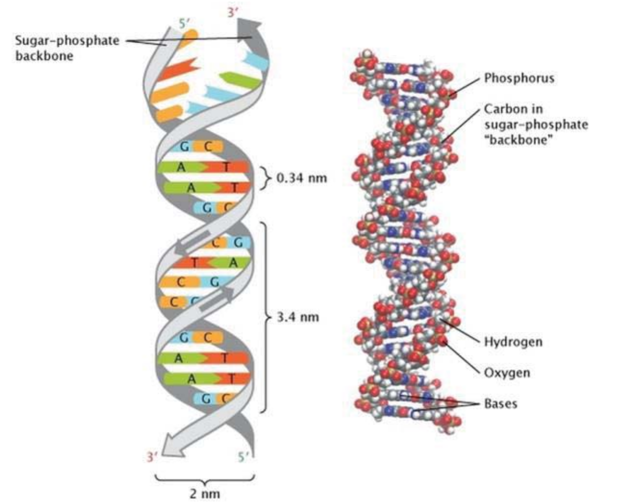
Figure 1: A spread 2 snake in a 3-dimensional hypercube - Snake (3, 2)

Snakes have been represented in various forms. Node-sequence representation, being the most naive and primary form of representation, is nothing but the ordered sequence of nodes that are traversed in an n-dimensional hypercube along the path (previously mentioned 0, 1, 3, 7, 6 is one such node-sequence representation). Among various other representations of snakes, transition sequence is a simple and parsimonious representation. For a 0-based transition sequence representation, it is a non-negative integer describing the transition of nodes (the position of change of the bit between the previous node and current node when the nodes are represented in a binary code) to build a snake. The change of node 0 to node 1 can be represented by transition "0" (node 000 changes to node 001 by changing the bit at position 0). Likewise, the traversal of node 1 to node 3 can be represented by transition 1 (node 001 changes to node 011 by changing the bit at position 1). In short, the node sequence 0, 1, 3, 7, 6 can be written as 0, 1, 2, 0 in transition sequence. For any transition sequence, only the first node needs to be chosen but due to the symmetric nature of a hypercube any node would serve the purpose by naming it as node 0. A canonical snake, in a transition sequence representation, is a snake transition sequence such that the first occurrence of any transition precedes the first occurrence of any other transition that is bigger than it. For example, a snake starting as 0, 1, 2, 3, 1, 0, 4 would be a canonical snake, since the first occurrence of transition "0" precedes the first occurrence of all other transitions that are bigger than it and so on and so forth for all the other transitions in it. While a snake starting as 0, 1, 2, 4, 0, 3 would not be a canonical snake since the first occurrence of transition "3" does not precede the first occurrence of transition "4". This transition sequence (transition sequence 0, 1, 2, 4, 0, 3) can be represented in its canonical form by using the smallest unused transition for the first occurrence of every new transition while rewriting it (and using this replacement elsewhere). So for 0, 1, 2, it would still be 0, 1, 2 in its canonical form. When we encounter transition "4" we use the next smallest unused transition "3" for it (and replace "4" with "3" everywhere else). So the sequence 0, 1, 2, 4, 0 would become 0, 1, 2, 3, 0. Later when we encounter a new transition "3" we have to use the next unused smallest transition i.e. transition "4" (and replace transition "3" in the old sequence with transition "4" in its canonical form). So its canonical form would be 0, 1, 2, 3, 0, 4. Also previous works have shown that a canonical representation of transition sequence can be used to represent any snake [4].

The snake in the box (SIB) problem has been an interesting and challenging problem for both mathematicians and computer scientists [8]. The challenge has been taken to another level every time a particular dimension's longest maximal snake(s) are found, as the search space grows exponentially. As the search space grows exponentially, it gets more and more difficult to do an exhaustive search and some kind of heuristic is required. David Kinny mentions some complete search techniques and illustrates the role of branching factor while backtracking [1]. He mentions the crucial pruning of the search space by using a canonical form [2].

2 DNA Basics

In this section, some basic and generic information about DNA is discussed which will help the reader to follow and appreciate the similarities discussed in the latter sections. Deoxyribonucleic acid or DNA is a double-stranded helix, with the two strands connected by hydrogen bonds [2] [3]. Its structure is shown in Figure 2.



The 3-dimensional double helix structure of DNA, correctly elucidated by James Watson and Francis Crick. Complementary bases are held together as a pair by hydrogen bonds.

Figure 2: A double helix structure of DNA* [9]

Courtesy:
http://www.nature.com/scitable/topicpage/discovery-of-dna-structure-and-function-watson-397

*The referenced web page was visited on November 2, 2014

It is found in every living cell and encodes the genetic instructions used in various aspects of development and functioning of living organisms. DNA controls the growth, functioning and reproduction of cells in the living organisms. The information in DNA is stored as a code which is made up of four chemical bases: adenine (A), cytosine (C), guanine (G), and thymine (T). The order, or sequence, of these bases determines the information available for building and maintaining an organism. These DNA bases pair up with each other, A with T and C with G, to form units called base pairs. The base pairs are constant, i.e. base A would always pair up with base T and base C with base G. It is beyond the scope of this article to discuss the reason why these bases always pair up with each other.

3 Building Canonical Snakes

The canonical snakes are representative of all the snakes in the search space or in other words all the snakes in the search space can be represented using one of the canonical snakes. We first introduced an exhaustive search algorithm to build canonical snakes in transition sequence as shown in Table 1. This algorithm is the first known algorithm to validate a snake in transition sequence representation without converting it into any other form. The validating algorithm is based on the idea of number of unpaired transitions that helps in maintaining the snake spread-k constraints. The exhaustive search makes no assumption about its search space. It tries a transition by adding it to a snake and validating the sequence. If it succeeds it moves to search for the next transition else it tries another transition until all the transitions available have been tried, after which it backtracks to its last successful transition and tries another transition from it. This is repeated until all the transitions at the first position have been tried and there is no other backtracking possibility.

Table 1: Building Snakes for Dimension n Spread k

- Initialize an ordered list (for transition sequence), call it the Primary List (**PL**) and 2 auxiliary sets (Paired and Unpaired Transition Set – **PTS** and **UTS**, which are mutually exclusive)
- Initialize **PTS** with transitions 0 to n-1 // 0 based transition sequence
- Add transitions 0 to k in the **PL** and in the **UTS** and remove these transitions from **PTS**
- Set a flag **isValid** to **true**

While (number of elements in **UTS** >= **k** and flag **isValid**)

{
  - Add an element i to **PL** which is different from last **k** transitions* and is a member of {0,..,**n**-1}
  - Flip the membership of this element i between **PTS** and **UTS** // mutually exclusive
  - If (number of elements in **UTS** >= **k)**

    {
    - Copy the **PTS** and **UTS** to new temporary sets Temporary PTS (**T-PTS**) and Temporary **UTS** (**T-UTS**)
    - Starting from the first transition in PL, flip the membership of each transitions between the current **T-PTS** and **T-UTS** and in each step check the number of elements in **T-UTS**>=**k** else set the flag **isValid** to **false** and break from this loop

    }
  - Else

    {
    - Set the flag **isValid** to **false**

    }

}
- If (not **isValid**)
  - Remove the last added transition

*Pruning the search space by removing certain invalid snakes

In an n-dimensional hypercube, for a 0-based transition sequence, the transition sequence consists of numbers between 0 to n-1. The unpaired number of transitions maintains the spread in the path. So, if we are looking at a sequence 0, 1, 2, 3, 1, we see that there are two transition "1" (in other words paired), while 0, 2, 3 are unpaired. As the number of unpaired transitions drops below k, the k-spread constraint is violated. For any transition chunk of length greater than or equal to k, it should hold that there are at least k unpaired transitions. And for any transition chunk of length d less than k there should be at least d unpaired transitions. So, {0, 1, 2, 3, 1}, {1, 2, 3, 1}, {2, 3, 1} and {0, 1, 2} are some of the examples of such transition chunks.

Figure 3: A transition chunk

As shown in Table 1 for a spread k snake, choosing a transition different from the last k transitions prunes the search space by removing the invalid snakes. An extra pruning step that is added is that if the next element that is being added to the list matches with the element which is at the last $(k+1)^{th}$ position in the list then the transition at the last $(k+2)^{th}$ position should not be there in the last k-transitions. For example, consider a transition sequence as $\{a_{k+2}, a_{k+1}, a_k, a_{k-1}, \ldots, a_1\}$ consisting of k+2 transitions. If we want to add transition "a_{k+1}" as the next transition in this transition sequence then we can add it only if transition "a_{k+2}" is not there in the subsequence $\{a_k, a_{k-1}, \ldots, a_1\}$.

4 The DNA

Snake 1 (11, 5): [0, 1, 2, 3, 4, 5, 6, 7, 1, 2, 8, 5, 9, 7, 0, 10, 8, 1, 4, 5, 7, 6, 10, 8, 3, 4, 2, 5, 10, 9, 6, 4, 1, 5, 7, 0, 9, 6, 3]

Snake 2 (6, 2): [0, 1, 2, 3, 1, 0, 4, 3, 0, 5, 4, 0, 1, 3, 4, 0, 2, 4, 1, 0, 4, 3, 1, 5, 3, 4]

Snake 3 (7, 3): [0, 1, 2, 3, 0, 4, 5, 1, 0, 3, 6, 4, 0, 1, 2, 3, 0, 4, 5, 1, 0]

4.1 The Underlying Structure

The three snakes shown above have a few things in common. Apart from being the longest maximal snake in a particular dimension-spread, they also share a particular underlying structure upon which it is built. Snake 1 is a spread 5 snake in dimension 11. Snake 2 is a spread 2 snake while Snake 3 is a spread 3 snake. All these snakes are the longest maximal snakes and are canonical palindromes (a canonical snake whose reverse when expressed in a canonical form is equal to the original canonical snake). They have one or two points of symmetry based on if they have an even spread or an odd spread. Let us look at the same snakes again with the highlighting.

Snake 1 (11, 5): [0, 1, 2, 3, 4, 5, 6, 7, 1, 2, 8, 5, 9, 7, 0, **10, 8,** **1, 4, 5, 7, 6**, **10, 8,** 3, 4, 2, 5, 10, 9, 6, 4, 1, 5, 7, 0, 9, 6, 3]

Snake 2 (6, 2): [0, 1, 2, 3, 1, 0, 4, 3, 0, 5, **4, 0, 1, 3**, **4, 0**, 2, 4, 1, 0, 4, 3, 1, 5, 3, 4]

Snake 3 (7, 3): [0, 1, 2, 3, 0, 4, 5, **1, 0, 3, 6, 4**, **0, 1**, 2, 3, 0, 4, 5, 1, 0]

The shaded region highlights the basic structure of the snake which lays the foundation of a particular snake, similar to specific sequencing of genes in DNA which later decides everything for the organism. This shaded region, which is termed as the DNA, is defined as "*DNA of a valid snake is the smallest portion of the snake (approximately at the center of the snake) that contains all the possible transition sequences for the snake and has one or more points of symmetry. It also defines the complementary pairs of the transitions that should be used in the remaining parts of the snake."* There is one or more than one point of symmetry in the DNA. In the simplest case, where there is only one point of symmetry, the equidistant transitions to the left and right of this symmetric point occur in pairs and are called complementary pairs henceforth in the paper. These complementary pairs always occur in pairs to the left and right of the DNA throughout the snake.

4.2 Odd and Even Dimensions

Odd and even dimensions have different DNA in their longest snakes, primarily because the number of possible transitions in the two types of dimensions is different, i.e. for odd dimensions it is odd, while for the other it is even. In an odd dimension, the arrangement of a possible odd number of transitions for pairing in the underlying structure (similar to the base pairing in DNA) will be different than the even transitions where the number of possible transitions is even. The spread of the snake also plays a role in defining the structure of the DNA. The number of initial transitions that are used in the DNA (shown in red color) is equal to its spread (since no k transitions can be the same in a spread k snake). These initial transitions form the core of the DNA. For odd spread, an odd number of transitions is already used in the DNA to form its core. Now, for odd spread snakes in an odd dimension, the remaining transitions that have to be paired uniquely, after forming the core, are even in number and can be uniquely paired. But for such snakes (snakes with odd spread) in even dimensions, the remaining transitions are odd in number and cannot be uniquely paired. Let us take an example of snake (7, 2) to illustrate more on the pairing of complementary pairs and symmetric points in DNA.

Snake (7, 2) : 0, 1, 2, 0, 3, 1, 0, 4, 2, 1, 0, 3, 5, 0, 1, 2, 4, 0, 6, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 4, 0, 3, 5, 0, 4, 2, 0, 3, 4, 0, 1, 2, 0, 6, 1, 0, 4, 2, 1, 0

The snake shown above is the longest maximal snake in dimension-spread (7, 2) [7]. Since for spread 2 snakes no two consecutive transitions can be the same, transitions 3 and 4 appear in the middle as shown using red color. The remaining 5 transitions have been paired but not uniquely, most of the transitions have been paired with more than 1 transition in the DNA (shown as the highlighted grey area). Also since there is more than one point of symmetry their pairing varies for three ways of finding the point of symmetry, i.e. {(3, 4), (3), (4)}. Say for example "5" can be paired with "0" if "3, 4" is the point of symmetry as both are equidistant from this point of symmetry. "5" can be paired with transition "3" if "4" is the point of symmetry. "5" can also be paired with "4" if "3" is the point of symmetry. The DNA for even spreads is difficult to create and we will restrict ourselves to the odd spreads. As explained earlier, for odd spreads in even dimensions the remaining transition options for creating the DNA would be odd which again would create non-unique pairing. To simplify our task we will confine ourselves to odd dimensions with odd spread. The remaining dimension and spread combinations are intended to be pursued as future work.

4.3 Similarity with DNA

So how is the underlying structure similar to DNA? And what role do these subsequences play in building snakes? If we observe closely we will find that all the transitions {0… n-1} have been used in creating this shaded part. Similar to the DNA in living cells, it contains all the information/ingredients that could be used later. Apart from having all the transitions it also defines two more interesting features, the base-pairing and the length of the longest snake possible that can be grown using this underlying structure. The first feature is easier to explain and demonstrate while the second feature can only be explained from the results obtained as is the case with mapping of particular genes to a particular characteristic in a living organism (i.e. mapping genotype with phenotype). Similar to the base pairing in DNA, i.e. base A always occurs with base T and base C always occurs with base G, the transition sequences also always occur in pairs defined using this underlying structure. In other words this underlying structure decides the transition that would appear with its complementary transition at any two equal distances from the symmetric point. Let us take the example of Snake 1, we see that the distance of transition "7" on the left side of transition "5" (the symmetric point), is always the same as the distance of transition "4" on the right side of transition "5" and vice versa. This is what also makes it a canonical palindrome (a canonical snake whose reverse when expressed in a canonical form is equal to the original canonical snake).

4.4 Building the DNA

Let us start from scratch while rebuilding these underlying structures for snakes. Building upon the idea from the previous section (Section 2.3), which described basic rules for a canonical snake in a transition sequence representation, we have the following mandatory guidelines:

1. No k subsequent transitions can be the same in a spread k snake.
2. In the snake, for all subsequences of size greater than the spread, the number of unpaired transitions is greater than or equal to the spread.

Let us build the DNA of Snake 1, the DNA of the longest snake in dimension 11 with spread 5. Let us start from the symmetric point (for odd spreads there is one symmetric point). So for keeping it simple, let us use "0" as the symmetric point. Now since no adjacent k (k is 5 in this case) sequences can be the same we can put four other transitions in this structure as shown below.

3, 1, 0, 2, 4

The order of transitions does not matter as this is the defining stage where the pairs are being defined and whatever transition we decide to put would form the definition of pairing. We could have used {3, 1, 0, 2, 4} or {0, 1, 2, 3, 4}. We built the first sequence by adding "1" to the left of transition "0" and "2" to the right of "0". Then we added "3" to the left and "4" to its right. From the above sequences we have defined that transition "1" is paired with transition "2" and transition "3" is paired with transition "4" as they are at equal distance from "0" on the left and right side. So far, for spread k, if the dimension n is equal to k then putting all the transitions like this would make the longest snake of length k. But as we increase the dimension, we need to decide where the other extra transition sequences would be placed. In our example the next two transitions (say transition "5" and transition "6") we can have the following sequences where either each of these transitions is placed in the same way to each side of the structure or switched on the other side as shown:

6, 5, <u>3, 1, 0, 2, 4,</u> 5, 6   or   6, 5, <u>3, 1, 0, 2, 4,</u> 6, 5

Both of these would be the longest snakes for dimension-spread (7, 5). As we go higher in the dimensions, we start adding new transitions or reusing the previous used transitions to left and right (if these transitions do not make the snake invalid). Based on the structure of the longest snake found so far, the second one containing {6, 5,…., 6, 5} is more common in odd dimensions with odd spread. So, let us add the next two transitions to this sequence, as shown:

7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8

After adding these, we can re-use the pair of transitions 2 and 1. One of the common patterns that have been found is that during reusing the transitions the transition that was placed on the left side last time is preferred on the right side and vice versa. So the new structure would look like:

2, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 1

At this point adding the remaining transitions (transitions 9 and 10) would look like:

9, 2, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 1, 10

This is all we need for the longest snake. This is the DNA of the longest snake in dimension-spread (11, 5). This is the same structure as that of Snake 1. In fact, when we used this underlying structure to build the longest snake, we found the following snake whose canonical form is Snake 1.

Found: [7, 3, 10, 8, 1, 0, 4, 2, 3, 10, 5, 0, 9, 2, 7, 6, 5, 3, 1, 0, 2, 4, 6, 5, 8, 1, 10, 0, 6, 9, 4, 1, 3, 0, 2, 7, 9, 4, 8]

Canonical: [0, 1, 2, 3, 4, 5, 6, 7, 1, 2, 8, 5, 9, 7, 0, 10, 8, 1, 4, 5, 7, 6, 10, 8, 3, 4, 2, 5, 10, 9, 6, 4, 1, 5, 7, 0, 9, 6, 3]

The above snake is the longest known snake in (11, 5) and is of length 39. This is also the maximally longest snake in this dimension-spread and is confirmed through exhaustive search in dimension-spread (11, 5).

4.5 Working -Skin Nodes and Shadows

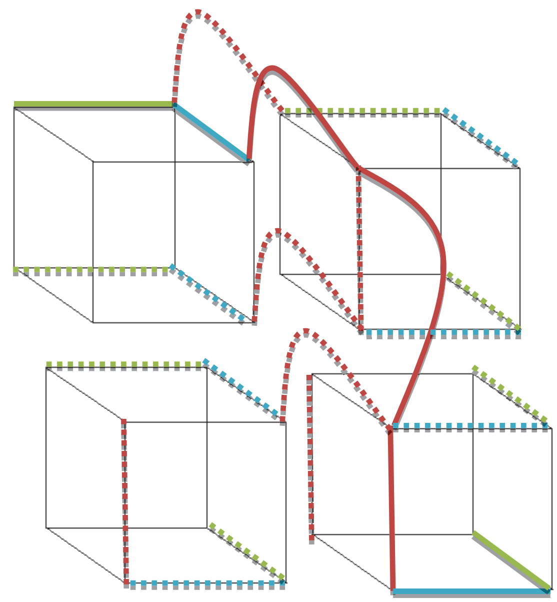


Figure 4: The longest snake and its shadows in dimension-spread (5, 3).

No hypothesis is ever complete without an attempt to explain its workings. In this section we will attempt to explain why this works. When a node is used in the path of a spread k snake, it makes all of its neighbors, at a Hamming distance of k or less, unusable for future path options (except the k-nodes in the path). These unusable nodes are called skin nodes. These skin nodes when joined in the sequence of being created by the snakes are termed as "shadows" in a smaller hypercube. Figure 4 shows a dimension 5 spread 3 longest snake and the shadows (connection of skin nodes) it casts upon its composite smaller hypercube (i.e. Hamming distance = 1). In an n-dimensional hypercube, for spread k, when we traverse an edge of the hypercube, this edge casts its shadow in all the adjoining smaller hypercubes and is resonated until spread k. These shadows inhibit the growth of snakes in the future. But if an algorithm can strategically place the edges considering our future moves such that our paths are less and less affected by these shadows then a long snake would be possible. The pairing of transitions helps us in maintaining and strategically placing the shadows. Of course, choosing the correct pair is decided by both of the factors, the past transitions that have been used and the future transitions that are left unused. The complementary shadows pave the way for a snake to move inside these tightly packed shadows.

5 Results and Discussion

Once the DNA is chosen, we start assembling the pairs to the left and right side of the structure (DNA) while following the pairing rule (using complementary pairs on the left and right side of the DNA). After adding the complementary pairs on both sides the snake is validated. For validating the snakes faster, we store a map of unpaired transitions from each position for the current snake and update it when new pairs are added on both sides (following the same fundamentals described in Section 2.3). Picking up the complementary pair is done as an exhaustive search and we can call it an exhaustive complementary pair search. Though the DNA occupies a very small part of the snake and intuitively it seems that the search for the snake is still the same old difficult job, the reality is quite the contrary. First of all, by only allowing a unique pair of transitions from the DNA (rather than an arbitrarily large combination of transition pairs) we restrict the search space to a much smaller area. The second and the most important contribution of the DNA is that the right DNA lays the required foundational structure that can only grow to be the longest snake in the hypercube. One of the current limitations of this approach is that these structures are very simple only for odd dimensions with odd spread. For the other three combinations of odd and even dimension-spread, these structures are far more complex and become less analogous to the helical structure of DNA with unique base-pairing. We intend to pursue the research in the remaining types of dimension-spread combinations, but for now we confine ourselves to odd dimensions with odd spread.

Using this DNA structure we were able to find new lower bounds for the snakes in (13, 5), (15, 7) and (17, 7). The previous best known results are from [5] [6].The longest snake known so far for (9, 3), of length 63, was also found using this approach. The results are summarized in Table 2. The value in the parentheses in the right hand column is the previously known lower bound. The search in dimension-spread (15, 7) was completed by exhausting the complete search space defined by the structure which means no other longer snake is possible for the structure. For the other dimension-spreads the search could not be completed at present and longer snakes are possible for such structures.

Table 2: New Lower Bounds for Snakes

Dimension-spread	Lower bound
(13, 5)	85 (79)
(15, 7)	57^c (55)
(17, 7)	103 (98)

c - Complete search for the given structure

6 References

[1] David Kinny. (2012). “A New Approach to the Snake-In-The-Box Problem,” *Proc. 20th European Conference Artificial Intelligence*, 462–467
[2] F. H. Crick, J. D. Watson. (1954). “The Complementary Structure of Deoxyribonucleic Acid”, *Proceedings of the Royal Society (London) A223*, 80
[3] J. D. Watson, F. H. C. Crick. (1953). “Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid”, *Nature 171*, 737
[4] Kochut, K. J. (1996). “Snake-In-The-Box Codes for Dimension 7”. *Journal of Combinatorial Mathematics and Combinatorial Computations* 20:175-185
[5] S. Hood, D. Recoskie, J. Sawada, D. Wong. (2011). Snakes, coils, and single-track circuit codes with spread k, *Journal of Combinatorial Optimization*, 1–21
[6] S. Hood, J. Sawada, C.H. Wong, (2010). “Generalized Snakes and Coils in the Box”
[7] W. D. Potter, R. W. Robinson, J. A. Miller, K. J. Kochut, D. Z. Redys. (1994). "Using the genetic algorithm to find snake-in-the-box codes”, *Proceedings of the 7th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, 421-426
[8] W.H. Kautz. (1958). “Unit-distance error-checking codes”, *IRE Trans. Electronic Computers*, 179–180
[9] http://www.nature.com/scitable/topicpage/discovery-of-dna-structure-and-function-watson-397

Reducing the number of lighting control attempts before Illuminance Convergence in the Intelligent Lighting System using the Layout Map of Lightings and Illuminance Sensors

Ryohei JONAN[2], Mitsunori MIKI[1], Shohei MATSUSHITA[2], Daichi TERAI[2], and Hiroto AIDA[1]
[1]Department of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *We research and develop an Intelligent Lighting System to realize the illuminance levels required by each office worker. This system controls each lighting appropriately for the level influence by the lighting's luminance on illuminance sensor measurements (illuminance/ luminance influence factor) . And by using illuminance/ luminance influence factor, the system quickly realized illuminance levels requested by workers even in larger-scale offices. In the past cases, we were allowed to measure illuminance/ luminance influence factors in the actual office before designing an Intelligent Lighting System. However, it is not always easy to have access into a user's office prior to installation to measure illuminance/ luminance influence factors: this inaccessibility has been a challenge to the popularization of Intelligent Lighting Systems. This study proposes a method to reduce the number of illuminance control attempts before illuminance convergence without using illuminance/ luminance influence factors.*

Keywords: Lighting, Intelligent Lighting System, illuminance simulation, illuminance sensor

1. Introduction

In recent years, there has been a rise in attention to approaches for improving the intellectual productivity, creativity and comfort of office workers [1]. A study by Boyce et al. have revealed that providing the brightness (illuminance) optimized for the work of each worker is effective from the viewpoint of improving the lighting environment[2]. Against this backdrop, the authors have undertaken studies on Intelligent Lighting Systems aimed at improving worker comfort in offices and reducing power consumption by lightings[3],[4]. An Intelligent Lighting System realizes the illuminance level requested by each worker (target illuminance) at the relevant illuminance sensor position with a minimum power consumption. An office with an Intelligent Lighting System is expected to allow workers to work in lighting environments customized for each of them, which will improve their comfort and reduce their stress. Moreover, providing the necessary levels of luminance at areas in need can lower the average illuminance in the whole room, which will result in a significant reduction of power consumption. As these advantages of Intelligent Lighting Systems are recognized, verification experiments have been underway at several offices in Tokyo, which have successfully realized required illuminance levels at the points where they are required, realizing high energy efficiency[5].

Intelligent Lighting Systems use a lighting control algorithm called Adaptive Neighborhood Algorithm using Regression Coefficient (ANA/RC) based on a hill climbing method for the optimization of lighting patterns[3],[6]. The ANA/RC varies the luminance of each lighting appropriately by using the level of influence (hereinafter referred to as illuminance/ luminance influence factor) on the measurement of each illuminance sensor. To dynamically learn the illuminance/ luminance influence factor, the ANA/RC performs regression analysis based on luminance variations and illuminance variations occurring upon microscopic variations in the luminance of lighting fixtures. Besides, since each worker is assigned a desk in a fixed position in most offices, illuminance sensor positions are also fixed already: hence, the illuminance/ luminance influence factors for each illuminance sensor can be measured by turning on and off lighting fixtures one by one in the office before introducing an Intelligent Lighting System.

An Intelligent Lighting System realizes the target illuminance for each worker by conducting about 30 to 100 lighting control attempts each spanning about one second. However, there is a concern that in larger-scale offices, the numbers of lighting fixtures and illuminance sensors will be larger, and the time required for each lighting control attempt will be longer. When illuminance/ luminance influence factors are known in accurate terms, then an accurate simulation of the user's office environment will be possible with considerations for effects of factors unique to the environment, such as reflections of lighting fixtures' light by walls or the effects of partitions, which cannot be known from a lighting layout map alone. By quickly deriving lighting patterns which achieve the target illuminance levels requested by workers in a simulated environment, and applying them to the real user environment, the system was able to quickly realize the target illuminance level for each worker[7]. In this way, an Intelligent Lighting System installed in a large scale environment can realize target illuminance levels requested by workers at a speed equivalent to or higher than in past applications.

However, while the system learns illuminance/ luminance influence factors by regression analysis, temporary correlation phenomenon between random numbers may occur, to cause a regression coefficient of a lighting fixture distant from an illuminance sensor to be assessed too highly. For this reason, an accurate estimation of illuminance/ luminance influence factors takes a very long time. Besides, sometimes the user's office may not readily allow the entry of outsiders for illuminance/ luminance influence factor measurement, which can be a factor hindering Intelligent Lighting Systems from being more popular.

This study proposes an alternative approach to reducing the number of lighting control attempts before illuminance level convergence is reached, instead of using illuminance/ luminance influence factors which require estimation or measurement. Verification experiments demonstrate that the number of illuminance/ luminance control attempts before reaching illuminance level convergence can be reduced even without using illuminance/ luminance influence factors which require either estimation or measurement. The study aims to make Intelligent Lighting Systems easier to introduce by eliminating the need of illuminance/ luminance influence factor measurement in the user environment.

2. Intelligent Lighting System

2.1 Construction of Intelligent Lighting System

An Intelligent Lighting System realizes an illuminace level desired by the user while minimizing energy consumption by changing the luminous intensity of lightings. The Intelligent Lighting System, as indicated in Fig.1, is composed of lighting fixtures equipped with lighting control device, illuminance sensors, and electrical power meters, with each element connected via a network.

The lighting control device evaluates the effectiveness of the current lighting pattern based on the illuminance data from illuminance sensors and electrical power data from a power meter. By repeating microscopic lighting pattern variations and effectiveness evaluations, the control system tries to minimize power consumption while satisfying the illuminance conditions required by each worker.

2.2 Control algorithm of Intelligent Lighting System

Intelligent Lighting System controls use a control algorithm (Adaptive Neighborhood Algorithm using Regression Coefficient: ANA/RC) based on Simulated Annealing (SA)[3],[6]. SA is a general-purpose local search method in which an approximate solution within a range near the current solution is generated and the approximate solution is accepted if the objective function improves. Taking the luminance of the lighting fixture as design variable, it randomly

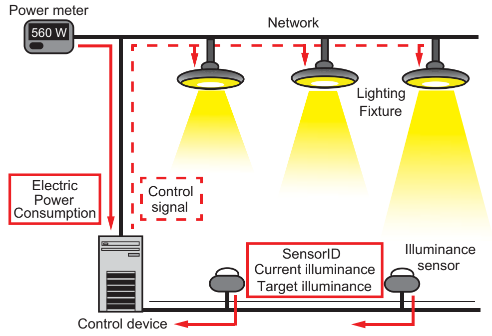


Figure 1: Configuration of the Intelligent Lighting System

varies the luminance of each lighting fixture in each search to an extent unnoticeable by workers to search an optimum lighting pattern. By repeating lighting control attempts of about a second 30 to 100 times, an Intelligent Lighting system realizes the target illuminance level requested by each worker.

The Intelligent Lighting System aims to adjust the illuminance to equal or greather than the target illuminance fot the location where the sensors are installed, and autonomously finds the lighting intensity to minimize the amount of electrical power used for lightings. This illuminance must be formulated as an objective function. The objective function is indicated in the Eq. 1.

$$f_i = P + w \times \sum_{j=1}^{n} R_{ij}(I_j - I_j^*)^2 \quad (1)$$

i : lightng ID, j : illuminance sensor ID
P : power consumption [W], w : weight
n : number of illuminance sensors
R : illuminance/ luminance influence factor [lx/cd]
I : current illuminance [lx], I^* : target illuminance [lx]

As indicate in the Eq. 1, the ovjective function consists of power consumption and illuminance constraint. Also, changing weight w enables changes in the order of priority for electrical energy and illuminance constraint. The illuminance constraint brings current illuminance to target illuminance or greater, as indicated by formula.

2.3 Illuminance/ luminance influence factor

It is known that the luminance of a lighting fixture is in proportion to the illuminance sensor measurement, which is shown by Eq. 2.

$$I = RL \quad (2)$$

I : illuminance [lx], L : luminance [cd]
R : illuminance/ luminance influence factor [lx/cd]

The illuminance/ luminance influence factor R is a value dependent on the working environment, which is considered to be a constant unless there is a change to the lighting environment.

In ANA/RC, the illuminance/ luminance influence factor of a lighting fixture to an illuminance sensor is estimated by regression analysis based on luminance variations and illuminance sensor measurement variations which occur when the lighting fixture luminance is varied in microscopic steps. It takes about 2 minutes for the system to learn an illuminance/ luminance influence factor because it needs to vary the luminance of lightings about 120 times. Besides, since each worker is assigned a desk in a fixed position in most offices, illuminance sensor positions are also fixed: hence, the illuminance/ luminance influence factors for each illuminance sensor can be measured by turning on and off lighting fixtures one by one in the office before introducing an Intelligent Lighting System. In the past applications of Intelligent Lighting Systems in actual offices, we were allowed to enter the user's office before installation to measure illuminance / luminance influence factors.

2.4 Reducing the number of lighting control attempts before illuminance convergence

When the scale of the environment of an Intelligent Lighting System is larger, the number of lighting fixtures and illuminance sensors will be larger. This may prolong the time required for the transmission of light control signals to a lighting fixture and the acquisition of illuminance data from an illuminance sensor, prolonging each attempt of lighting control, causing the total time required to realize the target illuminance for each worker to be longer than in the past.

When the illuminance/ luminance influence factor of each lighting fixture to each illuminance sensor is accurately known, the illuminance at the illuminance sensor in any given lighting pattern can be calculated from Eq. 2. Hence, an accurate simulation of the user's environment is possible taking account also of factors unique to the environment which are not known from a lighting layout map alone, such as the reflection of light from lighting fixtures by walls or the effects of partitions. By quickly deriving a lighting pattern which achieves the target illuminance level requested by each worker in a simulated environment and controlling lighting fixtures accordingly, the system was able to realize the target illuminance level for each worker with a single lighting control attempt. Furthermore, also in an environment with daylight, the target illuminance level requested by each worker was achieved within two lighting control attempts, by estimating the illuminance of daylight from the illuminance sensor measurement obtained in the first lighting control attempt, then using the estimated daylight illuminance value in the second control attempt (hereinafter referred to as simulation method) [7]. This demonstrates that using this approach, Intelligent Lighting System applications in large-scale environments can also realize target illuminance levels requested by workers more quickly than in the past.

2.5 Challenges for the estimation/ measurement of illuminance/ luminance influence factors

In estimating illuminance/ luminance influence factors using regression analysis, the change in illuminance at an illuminance sensor may be similar to the change in luminance of a lighting fixture distant from an illuminance sensor by chance, causing the regression coefficient of a lighting fixture distant from an illuminance sensor to be assessed too highly. For this reason, it is not easy to correctly estimate illuminance / luminance influence factors within the learning time of about 2 minutes: it will require quite a long time for the system to learn the illuminance/ luminance influence factors of lighting fixtures on each illuminance sensor with accuracy. For this, to achieve the target illuminance level for each worker quickly, it is desirable that illuminance/ luminance influence factors are known from the beginning.

By measuring in the user's environment before installation, accurate measurements of the illuminance/ luminance influence factors can be obtained without resorting to estimation based on regression analysis. However, sometimes the user's office may not readily allow the entry of outsiders for illuminance/ luminance influence factor measurement, which can be a factor hindering Intelligent Lighting Systems from being more popular. In some cases, a tenant objecting to an entry by outsiders may hinder illuminance/ luminance influence factor measurement. In order to make it easier to introduce Intelligent Lighting Systems and increase their popularity, an alternative approach to reduce the number of lighting control attempts before reaching illuminance convergence will be needed which does not rely on illuminance/ luminance influence factors requiring either estimation or measurement.

3. Reducing the number of lighting control attempts without using illuminance/ luminance influence factors

3.1 Fundamental principles of the proposed method

Here we propose an alternative approach to reduce the number of lighting control attempts without using illuminance/ luminance influence factors which require estimation or measurement. In the proposed method, the illuminance value at each illuminance sensor is calculated from the luminance data of lighting fixtures, based on the illuminance sensor positions relative to lighting fixtures known from a lighting fixture/ illuminance sensor layout map. In this way, the user's environment can be simulated more easily than by using illuminance/ luminance influence factors. By quickly deriving a lighting pattern which realizes the target illuminance level requested by each worker in a simulated environment and controlling lighting fixtures accordingly,

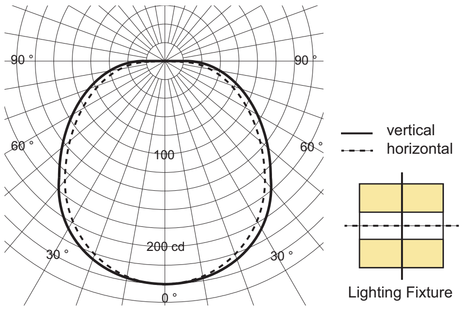


Figure 2: Luminous intensity distribution curve

the system quickly realizes the target illuminance level for each worker with a smaller number of lighting control attempts.

3.2 Formula of illuminance based on the illuminance sensor position

The relation between the luminance of a lighting fixture and illuminance at an illuminance sensor is expressed by Eq. 3.

$$I = \frac{L}{d^2}\cos\theta \tag{3}$$

I:illuminance [lx], L:luminance [cd]
d:Straight line distance from the lighting [m]
θ:vertical angle of the lighting to the sensor

From Eq. 3, the illuminance value at the illuminance sensor is calculated based on the straight line distance and the vertical angle from the lighting fixture. However, because Eq. 3 does not take account of the radiation characteristics of the lighting, the calculated illuminance value will include an error relative to the straight line distance between the lighting fixture and the illuminance sensor. Thus, the radiation characteristics of the lighting need to be considered. The radiation characteristics of the lighting can be known from the luminous intensity distribution curve, which is released as a design data for general lighting products. Fig. 2 shows the luminous intensity distribution curve of the LED lighting system manufactured by SHARP Corporation which is used in our verification experiment to be mentioned later.

From Fig. 2, the larger the vertical angle of the lighting to the illuminance sensor is, the more the luminance toward that direction is attenuated. Thus, errors in illuminance calculation can be lessened by considering the attenuation of luminance relative to the vertical angle based on the luminous intensity distribution curve.

3.3 Control algorithm of proposed method

The number of lighting control attempts before illuminance convergence is reduced by applying the lighting pattern obtained from simulation to each lighting fixture. To calculate the illuminance value at an illuminance sensor, the illuminance from daylight needs to be considered. The illuminance from daylight at the illuminance sensor position is expressed by Eq. 4.

$$D = I_j - I_L \tag{4}$$

j:illuminance senor ID
D:illuminance from daylight [lx]
I_j:illuminance acquired from the illluminance sensor [lx]
I_L:illuminance from lightings [lx]

Now the illuminance from a lighting fixture in a simulated environment is calculated, then the illuminance from daylight is estimated using Eq. 4. The proposed method uses the following control steps:

1) Each worker sets a target illuminance.
2) The system obtains illuminance information from the illuminance sensor.
3) The illuminance from daylight is estimated from Eq. 4.
4) A lighting pattern taking account of daylight in the simulated environment is obtained from calculation.
5) The system applies the lighting pattern obtained from calculation to the lighting fixture.
6) When the worker modifies the target illuminance, return to steps (1).

To respond to the constant variation of illuminance from daylight, steps (2) through (6) are repeated.

Because the proposed method cannot take account of factors unique to the user's environment, there occurs a difference between the illuminance calculated from simulation and the measurement by the illuminance sensor. The estimated illuminance from daylight should be inaccurate for this reason, but in any case, the system realizes the target illuminance requested by the worker by regarding this difference between the illuminance sensor measurement and the illuminance calculated from simulation as the illuminance from daylight.

4. Verification experiment

4.1 Verification experiment overview

To demonstrate the effectiveness of the Intelligent Lighting System using the proposed method, a verification experiment was conducted. An experimental environment as shown in Fig. 3 was configured with nine lighting fixtures and three illuminance sensors (A, B and C). In an experimental room measuring 5.4m x 5.4m x 2.45m, illuminance sensors were installed at the height of 0.7m above the floor, which is the height of an office desk recommended by JIS. As lighting fixtures, dimmable LED lighting system manufactured by SHARP was used.

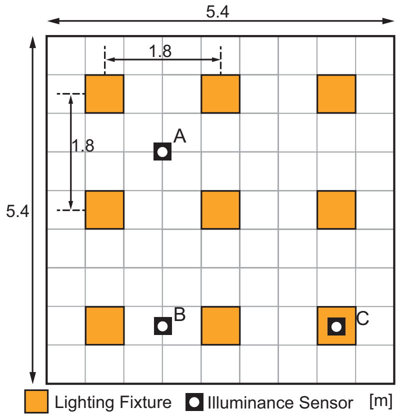


Figure 3: Experiment Environment (ground plan)

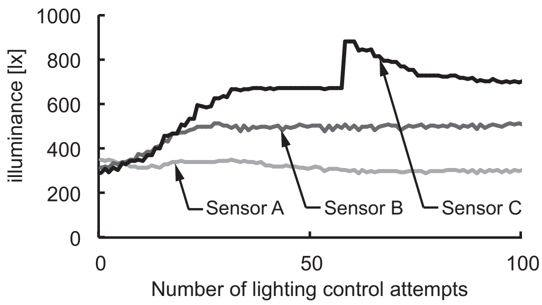


Figure 4: Illuminance History in ANA/RC

4.2 Comparison experiment concerning the number of lighting control attempts and lighting statuses

In the experimental environment shown in Fig. 3, an experiment concerning illuminance convergence was conducted. For illuminance sensors A, B and C, the target illuminance was set to 300 lx, 500 lx and 700 lx respectively. To verify the effectiveness of the proposed approach taking account of daylight, light from a fluorescent lamp was introduced to illuminance sensor C after 60 lighting control attempts. Illuminance convergence was tested for each of the ANA/RC, the simulation method and the proposed method. The intention was to verify the effectiveness of the proposed method by comparing the illuminance histories and lighting statuses between different methods.

The illuminance histories of the three illuminance sensors with ANA/RC are shown in Fig. 4; the illuminance histories of the three sensors with the simulation method in Fg.5; and the illuminance histories of the three sensors with the proposed methods in Fig. 6. Also the statuses of the lighting after 200 lighting control attempts in the three methods are shown in Fig. 7. From Fig. 4 and 6, about 30 lighting control attempts were required to realize the target illuminance level with ANA/RC, while one attempt of lighting control was enough to realize the target illuminance level with the

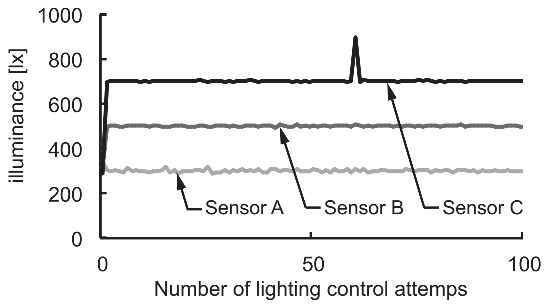


Figure 5: Illuminance History in Simulation Method

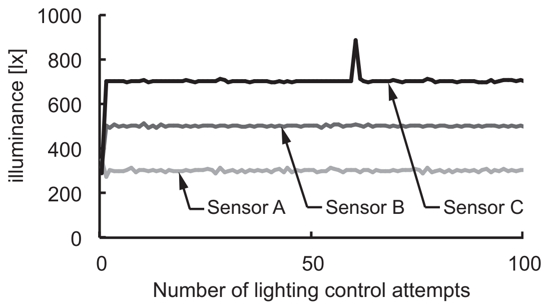


Figure 6: Illuminance History in Proposed Method

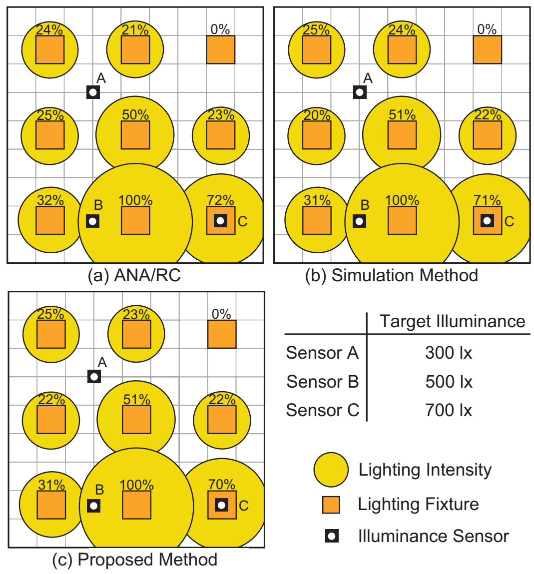


Figure 7: Satus of Lighting (200 lighting control attempts)

proposed method. From Fig. 5 and Fig. 6, no significant difference is found between the illuminance histories from the simulation method and the proposed method. The illuminance histories from the three methods show also that the introduction of the illuminance from daylight temporarily raises the illuminance measurement of illuminance sensor C at the point of 60 lighting control attempts. From Fig. 6, with the proposed method however, one attempt of lighting control was enough after the illuminance from daylight was

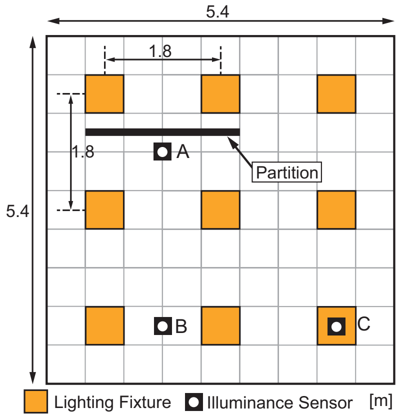


Figure 8: Experiment Environment（ground plan）

detected by the illuminance sensor for the system to realize again the target illuminance level. Next, from Fig. 7, a comparison between the lighting statuses resulting from the three methods shows that the three methods alike realized similar lighting patterns. These results demonstrate that the proposed method can reduce the number of lighting control attempts while quickly realizing the target illuminance level for each worker.

5. Verification experiment in a partitioned environment

5.1 Verification experiment overview

With the proposed method, it is not possible to take into consideration from the beginning the reflections of lighting fixtures's light by walls or shielding by partitions. Out of all, shielding by partitions has particularly large effects on illuminance measurements by illuminance sensors. Hence, an experiment was conducted to verify the effectiveness of an Intelligent Lighting System using the proposed method in a partitioned environment. An experimental environment was configured as shown in Fig. 8 in which a 1.8 m-high partition was installed above illuminance sensor A.

5.2 Comparison experiment concerning the number of lighting control attempts and lighting statuses

An illuminance convergence experiment was conducted in an experimental environment shown in Fig. 8. Under the conditions same as those used in the experiment in Chapter 4, illuminance convergence was tested with ANA/RC, the simulation method and the proposed method. To verify the effectiveness of the proposed method, illuminance histories and the lighting statuses from the three methods were compared.

Fig. 9 shows the history of illuminance at each illuminance sensor in the ANA/RC method; Fig. 10 shows the

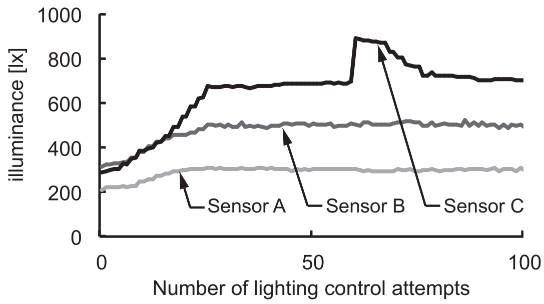


Figure 9: Illuminance History in ANA/RC

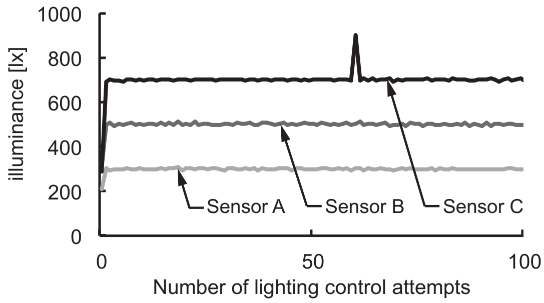


Figure 10: Illuminance History in Simulation Method

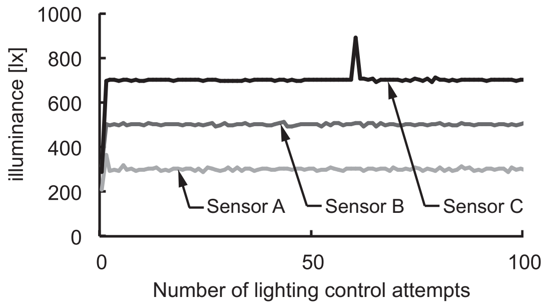


Figure 11: Illuminance History in Proposed Method

history of illuminance at each illuminance sensor in the simulation method; and Fig. 11 shows the history of illuminance at each illuminance sensor in the proposed method. Also, the statuses of lighting after 200 lighting control attempts in the three methods are shown in Fig. 12. From Fig. 9, with ANA/RC, about 30 lighting control attempts were required to realize the target illuminance level. In contrast, Fig. 11 shows that in the proposed method, the target illuminance was achieved with only one control attempt for illuminance sensor B and illuminance sensor C, and with two control attempts for illuminance sensor A which is more affected by the shielding effect of the partition. Based on Fig. 10 and Fig. 11, between the proposed method and the simulation method, the difference in the number of lighting control attempts required before illuminance convergence was only one at the greatest, which is not significant. Next, from Fig. 12, with ANA/RC and the simulation method, an energy-

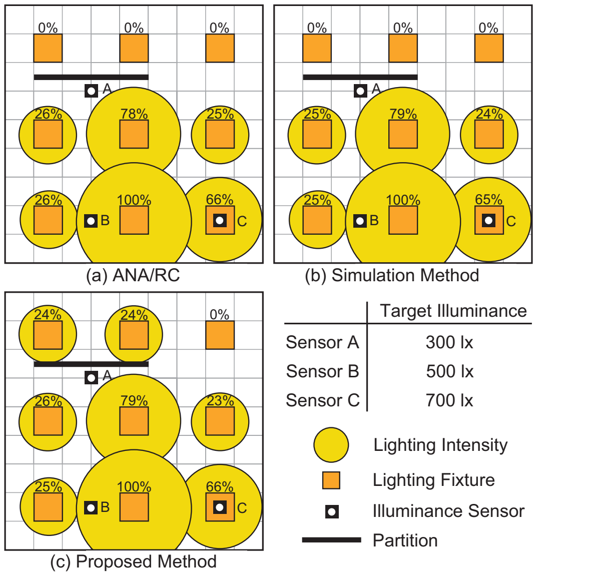


Figure 12: Satus of Lighting（200 lighting control attempts）

efficient lighting pattern is realized with the lighting fixtures above the partition turned off, as they have no effect on the illuminance sensor. In contrast, the proposed method fails to realize an optimum lighting pattern, without turning off the lighting fixtures above the partition.

These results demonstrate that the proposed method can quickly realize the target illuminance level desired by each worker even in a partitioned environment with a small number of lighting control attempts. However, because the proposed method incorporates no means to detect the presence of a partition, it turns on also those lighting fixtures which have no effect on illuminance sensors. Hence, the results indicate also that the lighting pattern may be less energy efficient with the proposed method than with ANA/RC or the simulation method.

owever, most partitions used in Japanese offices are short types which are not likely to block the effect of lighting influential on an illuminance sensor. Therefore, we consider that in reality, there will not be many cases where the proposed method sacrifices energy efficiency in determining its lighting patterns compared to the conventional method. Furthermore, in cases where a tall partition is to be installed, we also consider using a layout map including partitions to configure a simulated environment taking account also of any light-shielding structures.

Fig. 13 shows the side view of a partitioned environment in which an Intelligent Lighting System is installed. In such a case, the value of luminance from the lighting fixture is reduced in relation to the share of the shield to the arrowed range when configuring a simulated environment. It is considered that by incorporating the shielding effect of partitions from the beginning in this way, we will be able to realize more energy efficient lighting patterns.

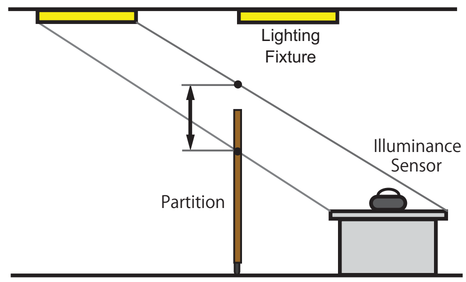


Figure 13: Intelligent Lighting System in a partitioned environment (side view)

6. Conclusion

In this study, we have proposed a method to reduce the number of lighting control attempts before reaching illuminance level convergence without using illuminance/ luminance influence factors. In the proposed method, the positions of illuminance sensors relative to lighting fixtures are checked from a layout map of lighting fixtures and illuminance sensors, then illuminance values are calculated based on the sensor positions. By calculating a lighting pattern which realizes the target illuminance level desired by each worker and applying the pattern to relevant lighting fixtures, the number of lighting control attempts before reaching illuminance convergence can be dramatically reduced. Verification experiments have demonstrated that the proposed method can quickly realize the target illuminance levels desired by each worker while reducing the number of lighting control attempts. Eliminating the need of entering the user's office to measure illuminance/ luminance influence factors before each installation, the proposed method can make Intelligent Lighting Systems easier to introduce.

References

[1] Olli Seppanen, William J. Fisk, "A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control", Proceedings of ASHRAE, 2005

[2] Peter R. Boyce, Neil H. Eklund, S. Noel Simpson,"Individual Lighting Control: Task Performance, Mood, and Illuminance", Joyrnal of the Illuminating Engineering Society, pp.131-142, Winter 2000

[3] M.Miki, T.Hiroyasu and K.Imazato, "Proposal for an intelligent lighting system, and verification of control method effectiveness", Proc. IEEE CIS, **1**, 520-525 (2004).

[4] M.Miki, K.Imazato and M.Yonezawa, "Intelligent lighting control using correlation coefficient between luminance anc illuminance", Proc.IASTED Intelligent Systems and Control, **497**[078], 31-36 (2005).

[5] Fumiya Kaku, Mitsunori Miki, Tomoyuki Hiroyasu, Masato Yoshimi, Shingo Tanaka, Takeshi Nishida, Naoto Kida, Masatoshi Akita, Junichi Tanisawa, Tatsuo Nishimoto, "Construction of intelligent lighting system providing desired illuminance distributions in actual office environment", Artifical Intelligence and Soft Computing, vol. 6114, pp. 451-460, 2010.

[6] S.Tanaka, M.Miki, T.Hiroyasu, M.Yoshikata, "An Evolutional Optimization Algorithm to Provide Individual Illuminance in Workplaces", Proc IEEE Int Conf Syst Man Cybern, **2**, 941-947 (2009).

[7] Shohei Matsushita, Sho Kuwajima, Mitsunori Miki, Hisanori Ikegami and Hiroto Aida, "Reducing the Number of Times Lighting Control is required to reach Illuminance Convergence in the Intelligent Lighting System", The 2014 International Conference on Artificial Intelligence (ICAI2014)

Insects detection in maize by endoscopic video analysis

Andre R. de Geus[1,2], Marcos A. Batista[1,2], Tércio A. Santos Filho[2], Sérgio F. da Silva[1,2]
[1]Modeling and Optimization Program, Federal University of Goias, Catalao, Goias, Brazil
[2]Biotechnology Institute, Federal University of Goias, Catalao, Goias, Brazil
Emails: geus.andre@gmail.com, marcos.batista@pq.cnpq.br, tercioas@ufg.br, sergio@ufg.br

Abstract— *Insects cause significant quantity and quality losses in stored grains. Producers are recommended to avoid excessive use of insecticides because they are harmful to living beings that consume the grains. Thus, it is of vital importance to early identify insects in grains to take control measures. Insects identification is usually done by collecting samples of grains from warehouses, followed by visual or laboratory analysis. However, this is a difficult and costly process. We propose to carry out this identification task automatically, using computational methods to perform endoscopic video analysis. The videos are recorded inside of grains warehouses by a endoscopic camera. As the classification process of moving objects in video depends fundamentally on precise segmentation of moving objets, we propose a new method of background subtraction and compared their results with the main methods of the literature according to a recent review. Experimental results show that the proposed method achieve more accurate results than state of art methods.*

Keywords: Background Subtraction, Segmentation, Video Analysis, Classification

1. Introduction

The growing need for food to meet the global demand, increased by the population growth, requires the grains harvested to be maintained with minimal losses to final consumption. However, stored grains are highly susceptible to insects infestation. Elias et. al (2008) in [8] reports that currently has several types of grain storages available in Brazil, but in all are susceptible to insects infestation. As intensive use of insecticides discouraged because they are harmful for living beings that manage and consume the grains, a timely identification of insects in the grains is of great importance to take measures to avoid losses. Yet according to [8], in Brazil, the grains annual losses caused by insects and arthropods infestation are estimated at 2 billion dollars, calculated as function of weight, volume and quality losses.

According to [1], in 2001, the quantitative average losses of grains in Brazil, estimated by the Ministry of Agriculture, Livestock and Supply are approximately 10% of the annual production. However the losses in quality are even greater, since that deteriorated grains have smaller monetary value.

Loeck (2002) [10] and Elias et. al (2008) [8] argue that periodic sampling is one of the most effective methods to indetify insects in grains. However, to collect such samples requires high cost of labor. Also, considering a real metal silo (as illustrated in Fig. 2), the sampling is a complicated process given that to take distributed samples throughout the storage is a very difficult task. This process can be greatly simplified by the installation of endoscopic cameras in silos and then perform computer vision analysis to determinate if there are insects in the grains.

In this work we analyse the presence of insects in maize. Cornfields cover around 20 millions hectares in Brazil with a average production around of 80 millions tons/year of maize. Aiming at avoid losses during harvest due rainy periods, growers normally crop once a year. The cropped maize grains are stored for approximately a year, being highly susceptible to weight and quality losses due to insects infestation mainly. Estimates suggest that, in Brazil, the post-harvest losses in maize are of about 14% of the total weight. Apart from causing quantitative losses, insects in stored grain are also linked to aflatoxin contamination that can lead to poisoning of living beings that eat the grains.

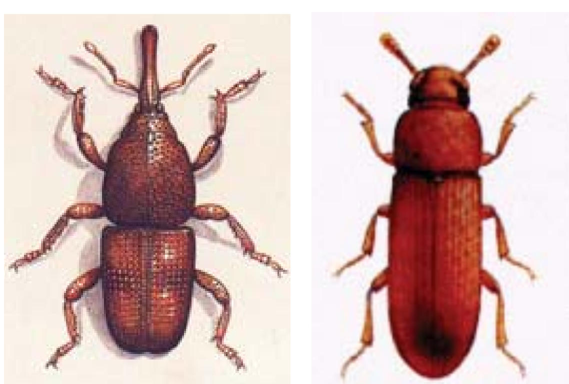
In Brazil, the maize weevil (Sitophilus zeamais) and the brown beetle (Tribolium castaneum) are main causes of maize losses. The Figure 1 illustrate these insects.

The maize insects control have been made mainly by pesticides. However, several studies have shown that population of pesticide-resistent maize insects are emerging, in this way pesticides should be used sparingly, when the insects appear. Thereby, to avoid losses in stored grain is necessary a early detection and classification of insects in grains to quickly apply the right pesticide. Thus, automated tools based on computer vision are promising to detect moving objects inside grain's storages and to classify them, following.

There are two basic approaches to detect moving objects in videos: optical flow and background subtraction (or foreground identification). Briefly, optical flow quantifies velocity vectors of the moving objects. Once computed, the measurements of moving object velocity can be used for a wide variety of tasks ranging from scene interpretation to autonomous, and active exploration by computer vision agents [2]. Background subtraction methods estimate and keep a background model, which is subtracted from the current frame. Such subtraction produces the foreground that is a delimitation of the moving objets. In this investigation we

Method ID	Method name	Reference	Settings
Basic method: mean and variance over time			
AdaptiveBackgroundLearning	Adaptive Background Learning	[14]	$T = 15$, $\alpha = 0.5$
Fuzzy based method			
FuzzyChoquetIntegral	Fuzzy Choquet Integral	[5]	$T = 0.67$, $LF = 10$, $\alpha_{learn} = 0.5$, $\alpha_{update} = 0.05$, *RGB + LBP*
Statistical method using one Gaussian			
DPWrenGABGS	Gaussian Average	[15]	$T = 12.15$, $LF = 30$, $\alpha = 0.05$
Statistical method using multiple gaussians			
MixtureOfGaussianV1BGS	Gaussian Mixture Model	[9]	$T = 10$, $\alpha = 0.01$
Type-2 Fuzzy based method			
T2FGMM_UM	Type-2 Fuzzy GMM-UM	[6], [7], [3]	$T = 1$, $k_m = 2.5$, $n = 3$, $\alpha = 0.01$
Statistical method using color and texture features			
MultiLayerBGS	Multi-Layer BGS	[16]	Original default parameters from [16]
Method based on eigenvalues and eigenvectors			
DPEigenbackgroundBGS	Eigenbackground/ SL-PCA	[12]	$T = 255$, $HS = 10$, $ED = 10$
Neural method			
LBAdaptiveSOM	Adaptive SOM	[11]	$LR = 180$, $LR_{training} = 255$, $\sigma = 100$, $\sigma_{training} = 240$, $TS = 40$

Table 1: Background subtraction algorithms used and the parameters settings of each algorithm. The parameter settings used are the same of Sobral and Vacavant investigation [14].



(a) (b)

Fig. 1: Maize insects: (a) the maize weevil (Sitophilus zeamais); (b) brown beetle (Tribolium castaneum). Sitophilus zeamais photo is taken from http://www.cnpms.embrapa.br/publicacoes/publica/2006/circular/Circ_84.pdf and the Tribolium castaneum photo is taken from http://www.pragas.com.br/poscolheita/pragasgraos/besouros/besouros.php.

Fig. 2: Metal silo. Photo taken from www.agencia.cnptia.embrapa.br

focus on background subtraction method because different species of maize insets have particular shapes that allows to classify such species.

Several studies revels that to classify objets in video correctly, segmentation of moving objets play a fundamental role. In this research, we propose a segmentation method of moving objets and compare them with the best methods from the literature, according to the Sobral and Vacavant investigation [14]. Although several studies compare background subtraction methods, to the author's knowledge, this is the first study to compare such methods to segment moving insects in grains using endoscopic video. Experiments with real videos, obtained with an endoscopy camera, reveal that the proposed method produces more precise segmentation result than state-of-art methods.

The remainder of this article is organized as follows: section 2 describes the main concepts of background subtraction techniques and presents the proposed background subtraction method; section 3 lists the methods used in our comparative study and describes the experimental results. Finally, the section 4 summarizes the main contributions of this investigation.

2. Proposed Method

Several methods for background subtraction have been proposed to track objects of interest in a scene. Basically, all of these methods try to effectively estimate a background model from a temporal sequence of frames. The background model is first initialised and then maintained along the time. To estimate the foreground, i.e., the moving objects, the current frame is subtracted from the current background model. There is a wide variety of techniques for estimate a background model. A reader interested in the subject may like to consult review papers as [4], [13], [14].

This section describes the main steps of the proposed algorithm.

2.1 Background Bootstrapping

An important step to every background subtraction method is initialise the background model, which in most cases do not have a starting clear background sequence of frames to build it. This step is called Bootstrapping and has to be fast and accurate. Therefore, a robust approach must be created to initialise the background model as quickly as possible. Our approach partitions the image into blocks (of 16x16 pixels) and only adds this region to the background model if high portions of the pixels are not moving. To determine if the pixels are not moving we analyse two consecutive frames. If more than 90% of the pixels in the block still have the same value, they are not moving. If it happens for more than 5 times consecutively, the block is set to "ready" and no more checked. Once all blocks are classified as "ready", the initial background is determined.

2.2 Background Updating

In our scenario, the insects are in constant movement and frequently push the maise skin, which should be considered as background. Therefore, updating the background becomes an essential step. That is the motivation to implement the learning rate background updating. After the bootstrapping step the background is updated considering every frame at time t in the video. Each new input frame I_t updates the background BG_t according to Eq 1.

$$BG_t = (\alpha * I_t) + ((1 - \alpha) * BG_{t-1}) \tag{1}$$

where α is the learning rate that determines how fast the background absorbs the moving objects, which in this case can be a high value, due to the high velocity of the insects. Our preliminary tests show that 0.01 produces good results, but could be better estimated if the velocity of the insects were measured to weigh the α value.

2.3 Foreground Extraction and Binarization

To determine the mask, or foreground, the algorithm proposed uses the background difference technique to remove the background frame by frame. The difference between the current frame and the background is determined by:

$$M_t(i,j) = \frac{(I_t(i,j) - BG_t(i,j)) * i^T}{3} \tag{2}$$

where i^T is de identity vector, $I_t(i,j)$ and $BG_t(i,j)$ are the vector (R,G,B) of pixel (i,j). M_t is a two-dimensional array at time t that represents the gray-scale intensity level of each pixel. The Fig. 3 below shows a mask example.

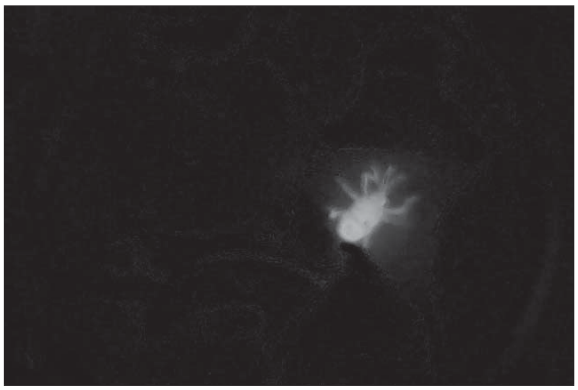
Given that the mask contains the moving object, we use the thresholding technique to determine which pixels are relevant and will be presented on the binarized image. The Fig. 4 shows the binarized mask of the example above.

3. Results

Our video data set consists on short-time videos recorded within a metal silo using an endoscopic camera with resolution of 640x426 pixels. The use of a low resolution camera is motivated by its low cost that became feasible for a real-world application in metal silos, where is required around of hundred endoscopic cameras.

To compare the proposed method we choose the best background subtraction algorithms reported in the recent review by Sobral and Vacavant [14] and implemented by the same authors in the BGS library available in https://github.com/andrewssobral/bgslibrary. The Table 1 lists the methods of BGS Library used and the settings for each of them. The settings used for each methods are the same used by Sobral and Vacavant in [14]. For more details about each method, the reader is referred to reference [14]. The Figures 5 and 6 show the results applying the proposed method and the selected state of art methods in four different frames extracted from our data set. In an visual analysis one can see that our approach delineates more precisely the insect's shape. Among the experimented methods, those who achieved more accurate results are our approach, MixtureOfGaussianV1BGS, LBAdaptiveSOM and MultiLayerBGS, respectively.

(a)



(b)

Fig. 3: (a) Original image and (b) mask

Fig. 4: Binarized mask with threshold 75

4. Conclusion

As known from many researches, object segmentation plays a crucial role in computer vision systems aimed to classify objects. In this study we present a object segmentation method based on the background subtraction technique. Our experiments using a video data set containing insects shows that the proposed method achieves more accurate segmentation results than state-of-art background subtraction methods listed in a recent research review. As object segmentation play a key role in object classification, our results indicate that the proposed method can be applied to build a insects recognition approach with more accurate results than when using others background subtraction methods. Additionally, to the best of our knowledge, this is the first study carried out to compare background subtraction methods applied to insect segmentation from endoscopic videos. As future investigation we intend to experiment the proposed methods in another object segmentation domains.

Acknowledgment

The authors would like to thank the financial support of the Brazilian National Council for Scientific and Technological Development (CNPq), the Foundation for Research Support of the State of Goias (FAPEG) and the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES).

References

[1] BARBERATO, C. Sem insetos grãos têm maior valor. *Folha do Paraná* (2001), online: http://www.folhaweb.com.br/?id_folha=2–1–114&dt=20010224.

[2] BARRON, J., FLEET, D., AND BEAUCHEMIN, S. Systems and experiment performance of optical flow techniques. *International Journal of Computer Vision 12*, 1 (1994), 43–77.

[3] BOUWMANS, T. ANF EL BAF, F. Modeling of dynamic backgrounds by type-2 fuzzy gaussians mixture models. *MASAUM Journal of Basic and Applied Sciences 1*, 2 (2010), 265–276.

[4] BOUWMANS, T. Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review 11*, 12 (2014), 31–66.

[5] EL BAF, F., BOUWMANS, T., AND VACHON, B. Fuzzy integral for moving object detection. In *IEEE International Conference on Fuzzy Systems* (Hong Kong, 2008), pp. 1729–1736.

[6] EL BAF, F., BOUWMANS, T., AND VACHON, B. Type-2 fuzzy mixture of gaussians model: Application to background modeling. In *International Symposium on Advances in Visual Computing* (Las Vegas, NV, 2008), pp. 772–781.

[7] EL BAF, F., BOUWMANS, T., AND VACHON, B. Fuzzy statistical modeling of dynamic backgrounds for moving object detection in infrared videos. In *IEEE Conference on Computer Vision and Pattern Recognition* (Miami, FL, 2009), pp. 60–65.

[8] ELIAS, M., LORINI, I., OLIVEIRA, M., MORÁS, A., AND SCHIAVONE, R. Pragas e microorganismos no armazenamento de grãos e derivados. In *Manejo tecnológico da secagem e do armazenamento de grãos*, M. C. Elias, Ed. Ed. Santa Cruz. Pelotas, 2008.

[9] KAEWTRAKULPONG, P., AND BOWDEN, R. An improved adaptive background mixture model for real-time tracking with shadow detection. In *European Workshop on Advanced Video Based Surveillance Systems* (London, UK, 2002), pp. 135–144.

[10] LOECK, A. E. *Pragas de produtos armazenados*. Pelotas: EGUFPEL, 2002.

[11] MADDALENA, L., AND PETROSINO, A. A self-organizing approach to background subtraction for visual surveillance applications. *IEEE Transactions on Image Processing 17*, 07 (2008), 1168–1177.

[12] OLIVER, N., ROSARIO, B., AND PENTLAND, A. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*, 8 (2000), 831–843.

[13] PICCARDI, M. Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics* (San Diego, CA, 2004), vol. 4, pp. 3099 – 3104.

[14] SOBRAL, A., AND VACAVANT, A. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding 122* (2014), 4–21.

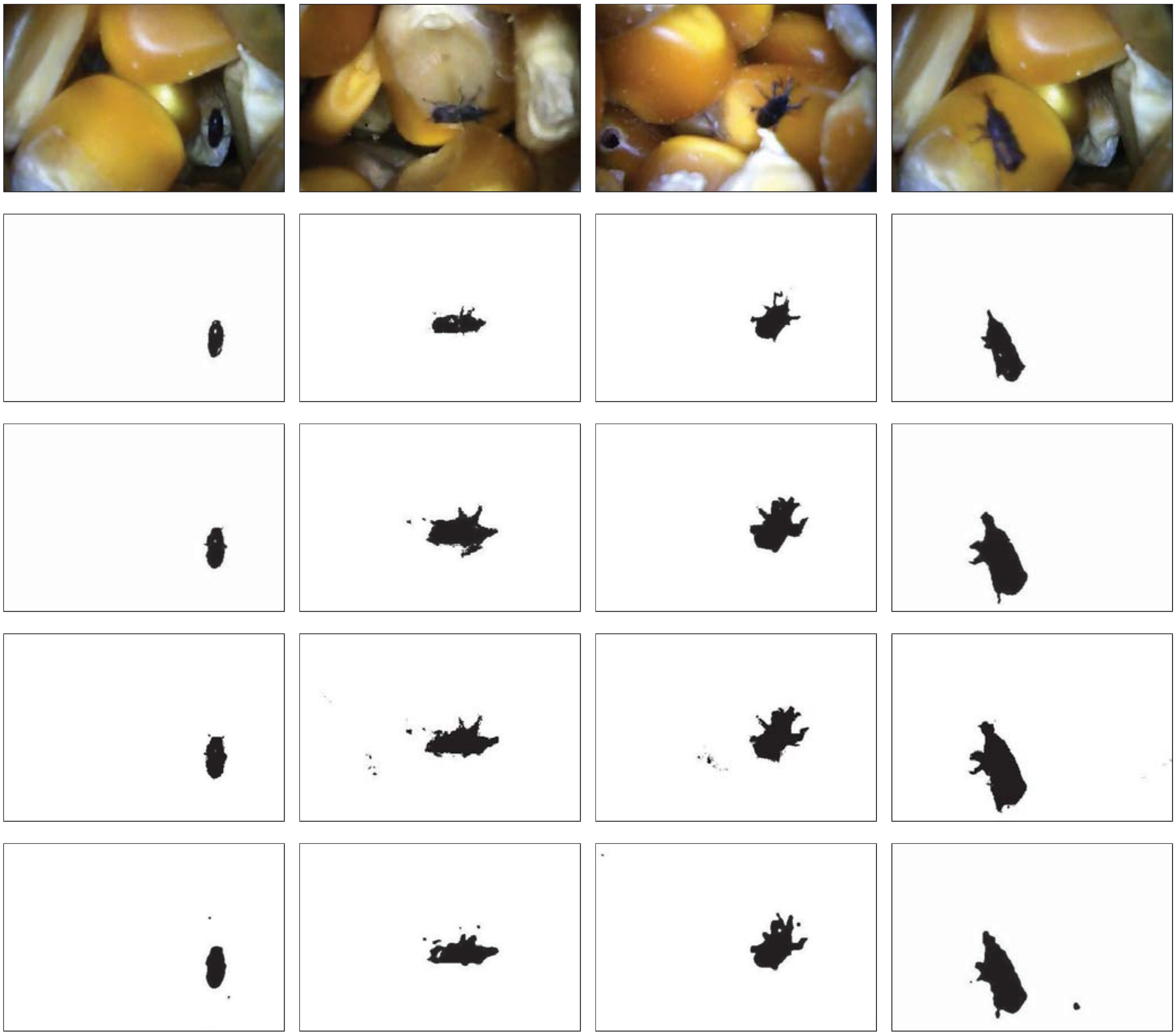


Fig. 5: Original imagens in the first row and Result obtained by the methods analysed: second row – proposed method; third row – MixtureOfGaussianV1BGS; fourth row – LBAdaptiveSOM; fifth row – MultiLayerBGS. The settings of each algorithm are given in the Table 1.

[15] WREN, C., AZARBAYEJANI, A., DARRELL, T., AND PENTLAND, A. Pfinder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence 19*, 7 (1997), 780–785.

[16] YAO, J., AND ODOBEZ, J. Multi-layer background subtraction based on color and texture. In *IEEE Conference on Computer Vision and Pattern Recognition* (Minneapolis, MN, 2007), pp. 1–8.

Fig. 6: This Fig. is a continuation of Fig. 5. Original imagens in the first row and Result obtained by the methods analysed: second row – DPWrenGABGS; third row – DPEigenbackgroundBGS; fourth row – AdaptiveBackgroundLearning; fifth row – FuzzyChoquetIntegral; sixth row – T2FGMM_UM. The settings of each algorithm are given in the Table 1.

Optimal PID Controller Design Using Krill Herd Algorithm for Frequency Stabilizing in an Isolated Wind-Diesel System

H. A. Shayanfar*[1], H. Shayeghi[2], A. Younesi[2]
[1] Center of Excellence for Power System Automation and Operation, Electrical Engineering Department, Iran University of Science & Technology, Tehran, Iran
[2]Electrical Engineering Department, University of Mohaghegh Ardabili, Ardabil, Iran

Abstract – *The main purpose of this paper is to design an optimal PID controller for frequency stabilization in an isolated wind-diesel power system. Optimal tuning of PID controller, formulated as an optimization problem and solved using Krill Herd (KH) algorithm. In order to prove the performance of proposed method, simulation carried out in four cases, step change in load of diesel side, step change in wind speed, random changing of the diesel side load, and random changing of the wind speed. Also, performance indices like overshoot, undershoot, settling time, ITAE and ISTSE are calculated and compared with Bee Colony (BC) algorithm. Results show that the proposed method is very robust and effective.*

Keywords: Wind-Diesel System, Krill Herd Algorithm, PID Controller

1 Introduction

In the entire world, electricity is one of the most demanding forms of energy in every one's daily life. There are groups of people that do not have access to the grid electricity, they are located in either remote or isolated communities, where grid connectivity is not at all neither economical nor viable. For this group of people, electricity is mainly supplied by small diesel-based power generation that it is very harmful for the environment [1]. Therefore renewable energy sources are used for reduce the dependency of this power generation systems to the fuel and as a result reduce the harmful effect of this systems to the environment. However, renewable energy sources are mostly intermittent, so they can't supply quality power constantly. This problem solved by combining more renewable energy sources together with non-renewable or storage devices [2-5]. The oscillations of wind speed and load demand lead to mismatch between the power generation and load demand resulting in mismatch in system frequency (f) and power (P) from their nominal values.

In the past, many researches have been proposed for controlling the oscillations of the frequency in hybrid power system generations.

*Corresponding Author, H. A. Shayanfar, hashayanfar@yahoo.com

This controllers are include control of pitch in wind side and governor in diesel side.

Many control strategies have been proposed in the literature. In [6-7], optimization of controller parameters proposed. Also, in [8-9] PI controller, in [10-12] variable structure control and in [13-14] energy storage controller have been reported.

The proportional-integral-derivative (PID) controller has its widespread acceptance in the industrial processes due to its simplicity in understanding and its applicability to a large class of process having different dynamics [15]. Thus, in this paper two PID controllers designed simultaneously, one for diesel side and another for pitch control of wind side.

It is shown that the appropriate selection of PID controller parameters results in satisfactory performance during system upsets. Thus, the optimal tuning of a PID gains is required to get the desired level of robust performance. Since optimal setting of PID controller gains is a multimodal optimization problem (i.e., there exists more than one local optimum) and more complex due to nonlinearity, complexity and time-variability of the real world power systems operation. Hence, local optimization techniques, which are well elaborated upon, are not suitable for such a problem. For this reason, a new bio-based swarm intelligence algorithm, called Krill Herd (KH) is proposed for optimal tuning of the PID controller gains to stabilize a synchronous machine connected to an infinite bus in this paper. The KH algorithm is based on the description of the herding of the krill agents in response to specific biological and environmental processes. The objective function of each krill individual is defined as its distances from food and highest density of the agent [16].

In this paper, optimal PID controller design for the frequency oscillation damping of a wind-diesel hybrid system, is formulated as an optimization problem and solved using Krill Herd (KH) algorithm. The results of the proposed method is compared with the Bee Colony (BC) algorithm. Simulation results show that the KH-based PID controller (KH-PID) has better performance in compared with BC-based PID controller (BC-PID) from the perspective of the response to step change in load demand of diesel side, step change in wind speed. Also, two performance indices according to the system defined and with overshoot, undershoot, and settling time calculated for both KH-PID and BC-PID controllers, which proves that the proposed KH-PID controller is very effective and robust compare to BC-PID controller.

2 System Modeling

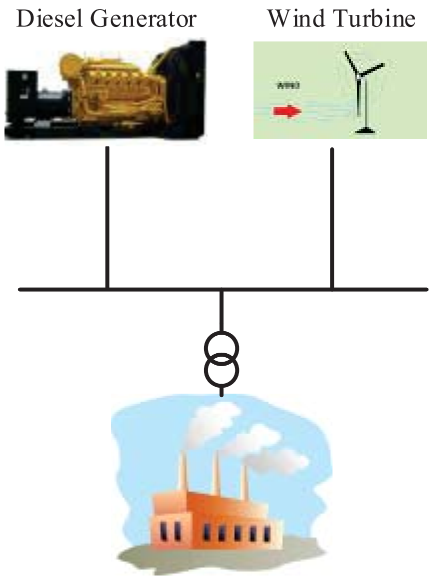


Fig. 1. Configuration of a hybrid wind-diesel isolated power system

Fig.1 shows the configuration of the hybrid wind-diesel isolated power system. Also, transfer function model of this system is shown in Fig. 2 [17]. Parameters of this system are given in [9].

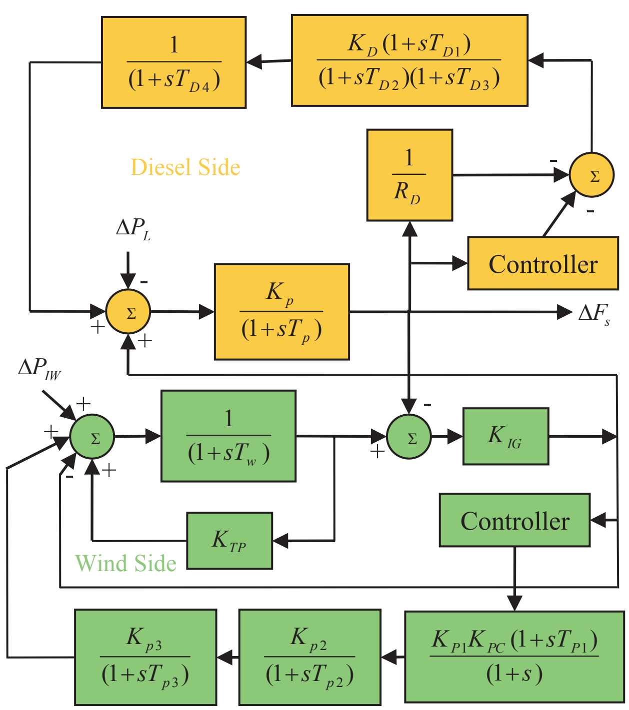


Fig. 2. Transfer function model of hybrid wind-diesel system

3 Krill Herd (KH) Optimization Algorithm

Krill Herd algorithm is one of the bio-inspired optimization algorithms that is being used for solving optimization problems. KH algorithm inspired from the krill herding motions. The time-dependent position on an individual krill in 2D surface is governed by the three main actions: movement induced by other krill individuals, foraging activity, and random diffusion. Eq. (1) shows the lagrangian model of this three actions.

$$\frac{dX_i}{dt} = N_i + F_i + D_i \tag{1}$$

Where N_i is the motion induced by other krill individuals, F_i is the foraging motion, and D_i is the physical diffusion of the ith krill individuals. N_i for a krill individual is defined in Eqs. (2) and (3).

$$N_i^{new} = N^{\max}\alpha_i + \omega_n N_i^{old} \tag{2}$$

Where,

$$\alpha_i = \alpha_i^{local} + \alpha_i^{target} \tag{3}$$

And N^{max} is the maximum induced speed, ω_n is the inertia weight of the motion induced in the range [0,1], N_i^{old} is the last motion induced, α_i^{local} is the local effect provided by the neighbors, and α_i^{target} is the target direction effect provided by the best krill individuals. The foraging motion is formulated in term of two main effective parameters. The first one is the food location and the second one is the previus experience about food location. Eqs. (4) and (5) describe this motion.

$$F_i = V_f \beta_i + \omega_f F_i^{old} \tag{4}$$

Where,

$$\beta_i = \beta_i^{food} + \beta_i^{best} \tag{5}$$

And V_f is the foraging speed, ω_f is the inertia weight of the foraging motion in the range [0,1], F_i^{old} is the last foraging motion, β_i^{food} is the food attractive and β_i^{best} is the effect of the best fitness of the ith krill so far. The physical diffusion of the krill individuals is considered to be a random process, and described in Eq. (6).

$$D_i = D^{\max}\left(1 - \frac{iter}{iter_{\max}}\right)\delta \tag{6}$$

Where D^{max} is the maximum diffusion speed, and δ is a random directional vector and its arrays are random values between -1 and 1.

Finally, the position vector of a krill individual during the interval t to $t + \Delta t$ is given by Eq. (7).

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt} \tag{7}$$

Δt is a very important constant and should be carefully set according to the optimization problem. Δt completely depends on the search space and can be simply obtained using Eq. (8).

$$\Delta t = C_t \sum_{j=1}^{NV} (UB_j - LB_j) \tag{8}$$

Where NV is the total number of variables, LB_j and UB_j are lower and upper bounds of the jth variables, respectively. C_t is a constant number between [0,2]. Simplified flowchart of the Krill Heard algorithm shown in Fig. 3 [16].

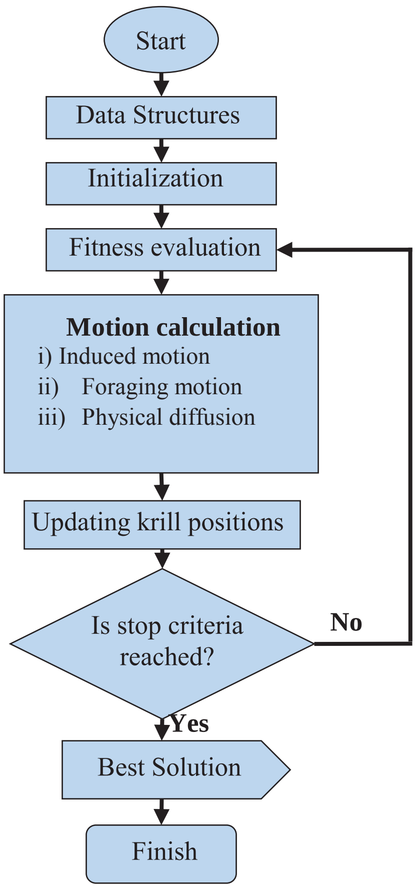


Fig. 3. Flowchart of the KH algorithm

4 Problem Formulation

PID controllers are being extensively used by industries today due to their simplicity. Its main duty in this paper is to eliminate the steady state error and improvement of dynamic response. The structure of PID controller that is used in this paper shown in Fig. 4. It has three parameters, K_{PN}, K_{IN}, and K_{DN}. Where N can be D or W, for diesel side and wind sides, respectively.

In the proposed system, there are two PID controllers, one for diesel side that its parameters denoted by K_{PD}, K_{ID}, and K_{DD}, and another for wind side that its parameters denoted by K_{PW}, K_{IW}, and K_{DW}. Therefore, there are six parameters that have to be well tuned. To increase frequency stabilization, a time based objective function is considered as follows:

$$J = \int_0^5 t \left| \Delta F_s \right| dt \tag{9}$$

Where, ΔF_s is the frequency deviation, and t is the simulation time. In the optimization process, it is aimed to minimize J in order to damp frequency oscillations.

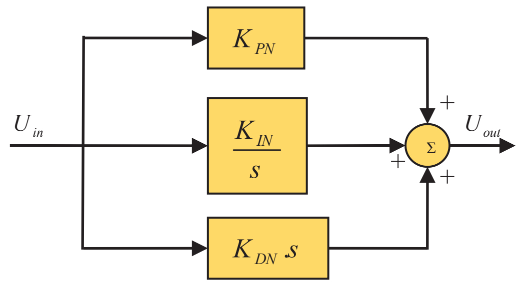


Fig. 4. PID controller structure

The design problem can be formulated as the following constrained optimization problem, where the constraints are the PID gains.

Minimize J Subject to:

$$\begin{aligned} &0 \le K_{PD} \le 350 \\ &0 \le K_{ID} \le 150 \\ &0 \le K_{DD} \le 50 \\ &0 \le K_{PW} \le 350 \\ &0 \le K_{IW} \le 150 \\ &0 \le K_{DW} \le 50 \end{aligned} \tag{10}$$

Results of the PID parameters based on the objective function J, solved using the proposed KH and BC algorithms (see Ref. [17] for more details) are given in Table 1. Fig. 5 shows the minimum fitness functions evaluating process.

Table 1. Optimal PID gains

Algorithm	K_{PD}	K_{ID}	K_{DD}	K_{PW}	K_{IW}	K_{DW}
KH	299.9	65.98	25.49	214.9	125.5	4.9
BC	240.36	55.67	0.326	38.69	0.779	9.08

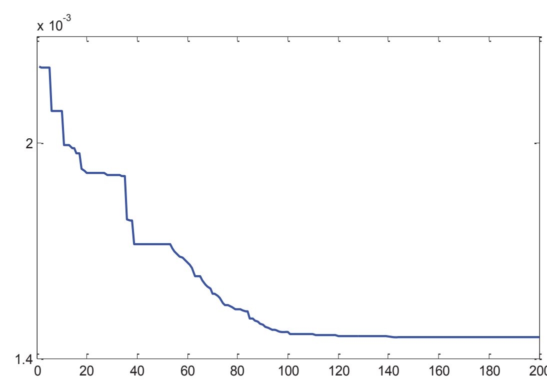


Fig. 5. Fitness convergence of the proposed KH algorithm

5 Simulation Results

In order to show the effectiveness of the proposed algorithm four case of simulations are considered.

Case 1. Step change in load demand of the diesel side

In this case of the simulations it's assumed that in t = 1 sec, a step change (ΔP_L=0.01 pu) occurred in diesel side load. Fig. 6 shows the frequency response of the system with KH-PID and BC-PID controllers.

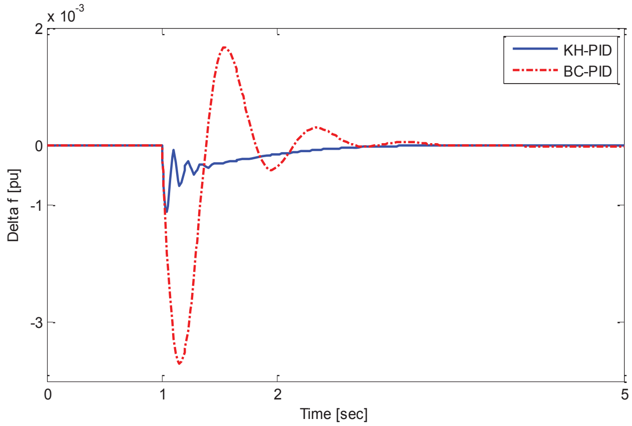


Fig. 6. Frequency response under load increase in diesel side

Also, performance indices like overshoot, undershoot, settling time, ITAE, and ISASE are calculated and shown in Table 2.

Table 2. Comparison of performance indices of two controllers for case 1

Algorithm	OS[%]	US[%]	Ts[sec]	ITAE	ISTSE
KH	0	0.1	1.25	0.6	2.59
BC	0.16	0.36	1.57	2.22	50.77

ITAE and ISTSE based on the system performance characteristics are defined as:

$$ITAE = 1000 \times \int_0^{\text{Upper limit}} t\,|\Delta f_s|\,dt \qquad (11)$$

$$ISTSE = 10^6 \times \int_0^{\text{Upper limit}} t^2 \Delta f_s^{\,2}\,dt \qquad (12)$$

As Fig. 6 and data of Table 2 show, the proposed KH-PID controller is very effective in compared with BC-PID controller.

Case 2. Step change in wind speed

In this case of the simulations it's assumed that in t = 1 sec, a step change (ΔP_{IW}=0.01 pu) occurred in wind speed. Fig. 7 shows the frequency response of the system with KH-PID and BC-PID controllers.

Also, performance indices like overshoot, undershoot, settling time, ITAE, and ISASE are calculated and shown in Table 3.

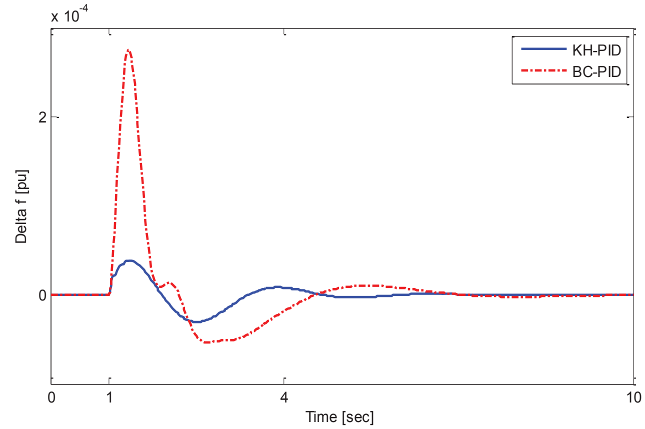


Fig. 7. Frequency response under wind speed increase

Table 3. Comparison of performance indices of two controllers for case 2

Algorithm	OS[%]	US[%]	Ts[sec]	ITAE	ISTSE
KH	0.0038	0.003	5.2	0.1523	0.0653
BC	0.0275	0.005	5.3	0.5247	0.7843

Data of Table 3, and Fig. 7 proves the results of case 1 of simulations.

Case 3. Random change in load of diesel side

In this case of simulations, assume that random load change shown in Fig. 8 applied to the system. Fig. 9 shows that the control effect on the system frequency deviation of the KH-PID controller is superior to that of the BC-PID controller.

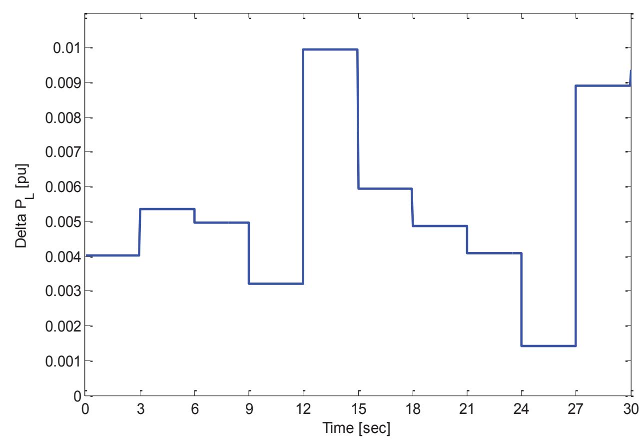


Fig. 8. Random load change

Case 4. Random change in wind speed

Same as case 3, in this case of simulations, assume that random wind speed shown in Fig. 10 applied to the system. Fig. 11 shows that the control effect on system frequency deviation of the KH-PID controller is superior to that of the BC-PID controller.

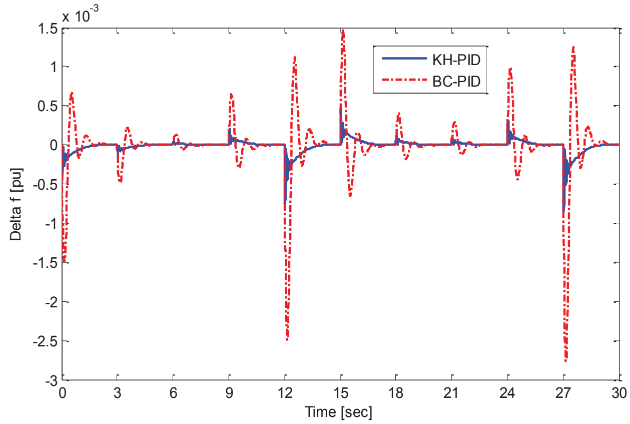


Fig. 9. System frequency response under random change in load of diesel side

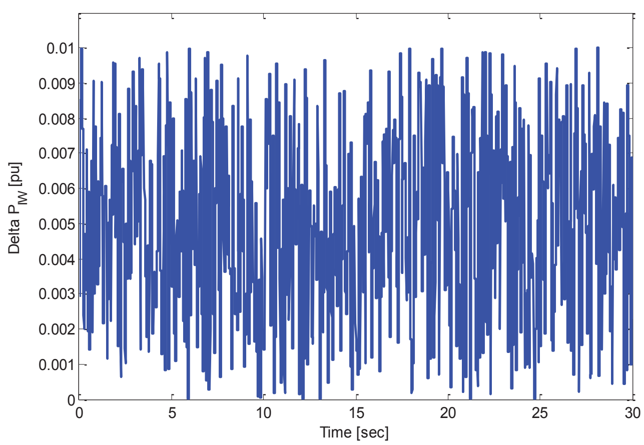


Fig. 10. Random change of wind speed

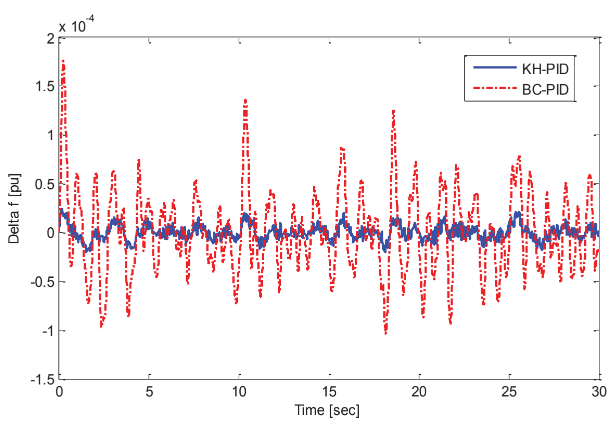


Fig. 11. System frequency response under random change in wind speed

6 Conclusions

In this paper, optimal PID controller design for the frequency control of a wind-diesel hybrid system, is formulated as an optimization problem and solved using Krill Herd algorithm. The results of the proposed method compared with Bee Colony algorithm. In order to prove the performance of proposed method, simulation carried out in four cases, step change in load of diesel side, step change in wind speed, random changing of the diesel side load, and random changing of the wind speed. Simulation results show that the KH-based PID controller (KH-PID) has better performance in compared with BC-based PID controller (BC-PID) from the perspective of the response to step change in load demand of diesel side, step change in wind speed. Also, two performance indices according to system defined and with overshoot, undershoot, and settling time calculated for both KH-PID and BC-PID controllers, which proves that the proposed KH-PID controller is very effective and robust compare to BC-PID controller.

7 References

[1] D. Hayashi, T. Senjyu, R. Sakamoto, N. Urasaki, T. Funabashi and H. Sekine, "Generating power leveling of renewable energy for small power system in isolated island," in *Intelligent Syst Application to Power Syst Conf*, Arlington, VA, 2005.

[2] M. Carmeli, F. C. Dezza, M. Mauri, G. Marchegini and D. Rosati, "Control strategies and configurations of hybrid distributed generation systems," *Renew Energy,* vol. 41, pp. 294-305, 2012.

[3] R. B. Bagen, "Evaluation of different operating strategies in small standalone power systems," *IEEE Trans Energy Convers,* vol. 20, no. 3, pp. 654-60, 2005.

[4] C. Wang and H. Nehrir, "Power management of a stand-alone wind/photovoltaic/fuel cell energy system," *IEEE Trans Energy Conv,* vol. 23, no. 3, pp. 957-967, 2008.

[5] R. Pena, R. Cardenas, J. Proboste, J. Clare and G. Asher, "Wind-diesel generation using doubly fed induction machines," *IEEE Trans. on Energ. Conv.,* vol. 23, no. 1, pp. 202-2014, 2008.

[6] V. Singh, S. Mohanty, N. Kishor and P. Ray, "Robust H-infinity load frequency control in hybrid distributed generation system," *Int J Electr Power Energy Syst,* vol. 46, pp. 294-305, 2013.

[7] A. Perera, R. Attalage, K. Perera and V. Dassanayake, "A hybrid tool to combine multi-objective optimization and multi-criterion decision making in designing standalone hybrid energy systems," *Appl Energy,* vol. 107, pp. 412-425, 2013.

[8] C. Nandar, "Robust PI control of smart controllable load for frequency stabilization of microgrid power system," *Renew Energy,* vol. 56, pp. 16-23, 2013.

[9] T. S. Bhatti, A. A. F. Al-Ademi and N. K. Bansal, "Load frequency control of isolated wind diesel hybrid power systems," *Energy Conversion and Management,* vol. 38, no. 9, pp. 829-837, 1997.

[10] W. Gu, W. Liu, C. Shen and Z. Wu, "Multi-stage under frequency load shedding for islanded microgrid with

equivalent inertia constant analysis," *Int J Electr Power Energy Syst,* vol. 46, pp. 36-49, 2013.

[11] N. Mendis and K. Muttaqi, "An integrated control approach for standalone operation of a hybridised wind turbine generating system with maximum power," *Int J Electr Power Energy Syst,* vol. 49, pp. 339-348, 2013.

[12] R. Hooshmand and M. Moazzami, "Optimal design of adaptive under frequency load shedding using artificial neural networks in isolated power system," *Int J Electr Power Energy Syst,* vol. 42, no. 1, pp. 220-238, 2012.

[13] Y. Wang, F. Ronilaya, X. Chen and A. Roskilly, "Modelling and simulation of a distributed power generation system with energy storage to meet dynamic household electricity demand," *Appl Thermal Eng,* vol. 50, no. 1, pp. 523-535, 2013.

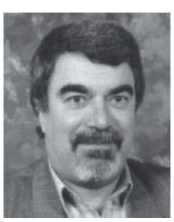
[14] I. Ngamroo, "Robust coordinated control of electrolyzer and PSS for stabilization of microgrid based on PID-based mixed H2/H1control," *Renew Energy,* vol. 45, pp. 16-23, 2012.

[15] M. N. Anwar and S. Pan, "A frequency response model matching method for PID controller design for processes with dead-time," *ISA Transactions,* vol. 55, pp. 175-187, 2015.

[16] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Communication in Nonlinear Science and Numerical Simulation,* vol. 17, no. 4, pp. 4831-4845, 2012.

[17] T. Chayatman, I. Ngamroo, S. Pothiya and S. Vachirasricirikul, "Design of optimal fuzzy logic-PID controller using bee colony optimization for frequency control in an isolated wind-diesel system," in *Transmission & Distribution Conference & Exposition: Asia and Pacific*, Seoul, 2009.

8 Biography



Heidar Ali Shayanfar received the B.S. and M.S.E. degrees in electrical engineering in 1973 and 1979, respectively. He received the Ph.D. degree in electrical engineering from Michigan State University, East Lansing, MI, USA, in 1981. Currently, he is a Full Professor with the Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran. His research interests are in the application of artificial intelligence to power system control design, dynamic load modeling, power system observability studies, voltage collapse, congestion management in a restructured power system, reliability improvement in distribution systems, smart grids and reactive pricing in deregulated power systems. He has published more than 497 technical papers in the international journals and conferences proceedings. Dr. Shayanfar is a member of the Iranian Association of Electrical and Electronic Engineers.

Abdollah Younesi received B.S degree in Electrical Engineering from Faculty of Technical Eng. Department of the Mohaghegh Ardabili University, Ardabil, Iran in 2012. Currently He is a M.S.E. student in Technical Eng. Department of the University of Mohaghegh Ardabili, Ardabil, Iran. His area of interest are application of artificial intelligence in power system automation and control, application of Reinforcement Learning to power system control, Fuzzy Systems, Heuristic optimization in power system control. He is a student member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Hossein Shayeghi received the B.S. and M.S.E. degrees in Electrical and Control Engineering in 1996 and 1998, respectively. He received his Ph. D. degree in Electrical Engineering from Iran University of Science and Technology, Tehran, Iran in 2006. Currently, he is a Full Professor in Technical Engineering Department of University of Mohaghegh Ardabili, Ardabil, Iran. His research interests are in the Application of Robust Control, Artificial Intelligence and Heuristic Optimization Methods to Power System Control Design, Operation and Planning and Power System Restructuring. He has authored of 6 books in Electrical Engineering area. He has published more than 280 papers in International Journals and Conference proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE

Verification of a Seat Occupancy/Vacancy Detection Method Using High-Resolution Infrared Sensors and the Application to the Intelligent Lighting System

Daichi TERAI[1], **Mitsunori MIKI**[2], **Katsuya ITO**[1], **Kohei YAMAGUCHI**[1] **and Hiroto AIDA**[2]
[1]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Department of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *We have conducted research on the Intelligent Lighting System that realizes illuminance demanded by a worker at minimum power. In real office the system realized to improve in office workers comfort and to reduce the power consumption. In the system, workers are necessary to change the occupancy/ vacancy status. However some workers didn't change appropriately it. Thus there were lights that provided brightness more than required even though there were no workers. Consequently, we propose a method for automatically controlling changes in the occupied/ unoccupied status of a worker's seat. This study uses high-resolution infrared sensors that can detect temperature. The method detects the occupied/ unoccupied status of a worker's seat by using temperature values output by high-resolution infrared sensors and their differences. We confirmed that the system incorporating the proposed method detects the occupied/ unoccupied status and realizes illuminance demanded by a worker and reduces the power consumption.*

Keywords: infrared sensor, optimization, energy conservation, lighting control

1. Introduction

It has been demanded in recent years to improve intellectual productivity and creativity of workers in an office environment. It has also been reported that improving office lighting environment enhances workers' intellectual productivity[1]. In particular, there have been extensive studies on the impact of office lighting environment on workers' comfort, and it has been clarified that providing each individual with the optimal brightness for their work leads to an improvement in their comfort[2].

Against such a backdrop, we have conducted research on the Intelligent Lighting System which realizes illuminance demanded by a worker at minimum power[3]. The intelligent lighting system realizes the target illuminance for each worker. The target illuminance refers to brightness desired by a worker and is set by the illuminance sensor button on the PC on his/ her desktop or the physical button installed on an illuminance sensor. The effectiveness of the Intelligent Lighting System has been verified so far[4]. As its effectiveness was acknowledged, demonstration tests have been conducted in real office environments.

The Intelligent Lighting System judges that it is not required to provide illuminance to an area unoccupied by any worker. When turning off the lighting fixture providing illuminance to the area in question, the system dims or turns off surrounding lighting in order not to affect working space of other workers, thereby realizing even higher energy saving. The results of demonstration tests at real offices, however, showed that workers did not appropriately change the occupied/ unoccupied status. Those results confirmed that there were lighting fixtures which provided brightness more than required even though there were no workers. Therefore we propose a method for automatically controlling changes in the occupied/ unoccupied status of a worker's seat. This is expected to improve the power consumption reducing effect.

This study proposes a method for detecting the occupied/ unoccupied status of a worker's seat by installing infrared sensors, which are generic. Existing methods using infrared sensors control lighting on the basis of two values: whether a person is or is not in the detection range of infrared sensors. With the Intelligent Lighting System, however, it is required to identify the detailed state of occupancy by workers because multiple workers are in the detection range. Consequently, we propose and verify a method for detecting occupancy or vacancy using high-resolution infrared sensors that can detect temperature for each of 256 pixels corresponding to the same number of sections of the detection range.

2. Intelligent lighting system

2.1 Configuration of Intelligent Lighting System

An Intelligent Lighting System is a lighting control system to realize illuminance demanded by a worker(target illuminance) at minimum power. It consists of lighting fixtures, control devices, illuminance sensors, a power meter and a network to connect them. Fig. 1 shows the configuration of the Intelligent Lighting System.

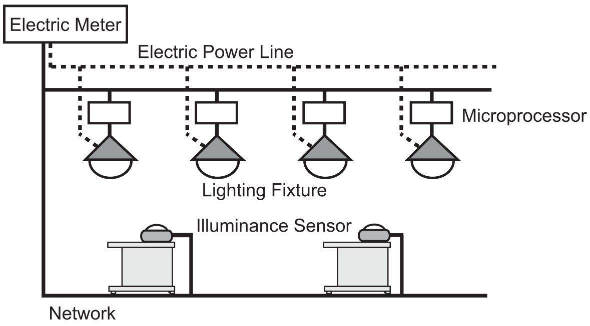


Figure 1: Configuration of the Intelligent Lighting System

As shown in Fig. 1, control devices acquire the illuminance information, the target illuminance information and the occupancy/ vacancy information. From these information, the control device controls the luminance of lighting fixtures using an optimization method. By this control, the system realizes the target illuminance required by each worker and saves power consumption.

2.2 Illuminance control algorithm

An Intelligent Lighting System uses an Adaptive Neighborhood Algorithm using Regression Coefficient (ANA/RC) based on Simulated Annealing (SA). The system solves an optimization problem in an autonomous distributed style. In this algorithm the luminance of each lighting fixture is the design parameter, the target illuminance of each illuminance sensors is the constraint and the total power consumption of the lighting is the objective function. Thus, the system derives the luminance of each lighting fixtures to realize the target illuminance for each worker and the reducing of power consumption. the system learns the effect of each lighting fixtures on each illuminance sensors using a regression analysis and varies appropriately the luminance of each lighting fixture. In this way, the system can realize the optimal luminance. The flow of control by the Intelligent Lighting System using ANA/RC is shown below.

1) Set the target illuminance of each illuminance sensor.
2) Turn on each lighting fixture at the initial luminance.
3) Obtain values measured by illuminance sensors and a power meter.
4) Calculate the values of the objective function.
5) Determine the next luminance in accordance with illuminance/ luminance influence factor and turn on each lighting fixture at the next luminance.
6) Obtain values measured by illuminance sensors and a power meter.
7) Calculate the values of the objective function referred to in step (5) under the new lighting condition.
8) Conduct a regression analysis with changes in luminance of each lighting fixtures and changes in the illuminance of each illuminance sensor to estimate illuminance/ luminance influence factor.
9) If the value of the objective function is improved, accept the next luminance. Otherwise, revert to the original luminance.
10) Go back to step (3).

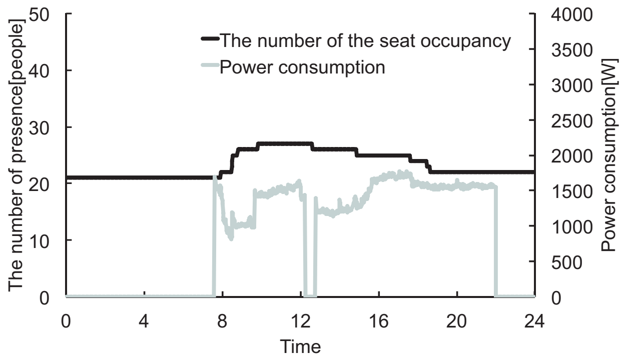


Figure 2: History of the number of the seat occupancy and power consumption of the Intelligent Lighting System

The objective function is indicated in the Ecuation(1).

$$f_i = P + \omega \times \sum_{j=1}^{n} g_{ij} \tag{1}$$

$$g_{ij} = \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ R_{ij} \times (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{cases}$$

$$R_{ij} = \begin{cases} r_{ij} & r_{ij} \geq T \\ 0 & r_{ij} < T \end{cases}$$

i:lighting ID , j:illuminance sensor ID, P:power consumption [W], ω:weight[W/lx^2]
Ic:current illuminance [lx], It:target illuminance [lx], r:regression coefficient, T:threshold

As shown in Fig. (1), the objective function f consists of power consumption and illuminance constraint. Illuminance constraint g_{ij} which set target illuminance of each illuminance sensors as constraint is changed by illuminance/ luminance influence factor r_{ij}. The objective function f operates to emphasize illuminance constraint as illuminance/ luminance influence factor is larger. The algorithm also can restrict lighting fixtures affect an illuminance sensor to lighting fixtures near the illuminance sensor by setting a threshold. Therefore the system controls lighting fixtures far from the illuminance sensors as a purpose only minimization of power consumption.

2.3 Issues Regarding Occupancy and Vacancy in the Intelligent Lighting System

The Intelligent Lighting System judges that space does not need brightness if it is not occupied by a worker. It is possible to reduce unnecessary brightness by treating the target illuminance for an unseated worker as 0lx. A change in the occupied/ unoccupied status is made by each worker by the occupied/ unoccupied button in the web user interface or the physical button installed on the illuminance sensor.

When the system was introduced to a real office, however, the occupied/ unoccupied status was not appropriately changed. Fig. 2shows the number of seated workers and the transition in the power consumption of the Intelligent Lighting System on January 15, 2011 in the Tokyo Building, in which the Intelligent Lighting System was introduced. The number of workers was 42 in the area in the Tokyo Building to which the Intelligent Lighting System was introduced.

Fig. 2 indicates that, for about 20 workers, the occupied/ unoccupied status was always indicated as "occupied" on the system regardless of a period of time. It also shows that the percentage of workers who changed the occupancy status was 10% of all workers. Power consumption declined temporarily in the morning during which the number of seated workers increased. This is considered to be because the impact of light from outside was so great as to suppress the luminance of lighting fixtures as it was fine in Tokyo on that day. It is also found that power consumption increased after 15:00 due to the smaller impact of light from outside resulting from a reduction in solar radiation. It is found that the light was turned off at 22:00 as power consumption became 0 at 22:00.

Assuming that the number of workers increases linearly from 8:00 to 9:00, during which period workers come to the office, and decreases likewise from 18:00 to 22:00, during which period they leave the office, power consumption is also considered to increase/ decrease linearly in the same manner. Because of workers who do not change the occupied/ unoccupied status, however, brightness more than necessary was provided even though the space was unoccupied by a worker. The occupied/ unoccupied status was not changed either when workers temporarily leave their seat for attending a meeting or going out. It can be said to be important to grasp the occupied/ unoccupied status of a worker's seat correctly from the perspective of energy saving in lighting.

One cause cited for the fact that the occupied/ unoccupied status is not changed appropriately is that changing it is somewhat bothersome because it is required to start up a web browser or press the button attached to the illuminance sensor. Therefore, this study proposes a method for reducing power consumption further by using high-resolution infrared sensors to detect the occupied/ unoccupied status of workers' seats and setting the target illuminance for a worker not occupying their seat as 0.

Table 1: Specification of the high-resolution infrared sensor

Measurement	MEMS Thermal IR Sensor
Element Type	16×16 pixels
Detection Range	3.6 m × 3.6 m (3 m detection length)
View Angle	90 °
Temperature Range	5 — 50 °C
Temperature Resolution	0.15×10^{-2} °C

3. Occupancy/vacancy Detection Using High-resolution Infrared Sensors Method

3.1 Outline of a High-resolution Infrared Sensor

A high-resolution infrared sensor is a noncontact temperature sensor that combines a wide range of visual field and high-precision area temperature detection. Table. 1 gives the specifications of a high-resolution infrared sensor.

The difference between existing infrared sensors and the high-resolution infrared sensor is that the latter can roughly identify the number of persons in the detection range of a sensor. Existing infrared sensors cannot identify the number and locations of persons because they detect a person in the detection range bivalently: whether there is a person or not. On the other hand, the high-resolution infrared sensor can grasp the number and locations of persons since it can detect the temperature of every part of the detection range. The high-resolution infrared sensor outputs the average temperature in each of 256 sections of the detection range corresponding to the same number of pixels on the sensor.

3.2 Occupancy/vacancy Detection Algorithm

We propose a method which detects the occupancy/ vacancy status of a worker's seat by using temperature values output by high-resolution infrared sensors and their differences. The flow of the occupancy/ vacancy detection procedure is shown below. Let T be the threshold temperature for distinguishing heat sources, ΔT be the temperature difference threshold for determining whether a seat is occupied or unoccupied by a person, and N be the number of adjacent pixels required for detecting occupancy.

1) Assign the value "unoccupied" to all pixels in their initial condition.
2) Obtain the difference between the temperature t seconds ago and the present temperature.
3) Assign the value "occupied" to the pixel representing the section for which temperature difference is ΔT °C or greater with temperature being T °C or above.
4) Assign the value "unoccupied" to a pixel representing the section for which temperature difference is $-\Delta T$ °C or less with temperature being T °C or below.

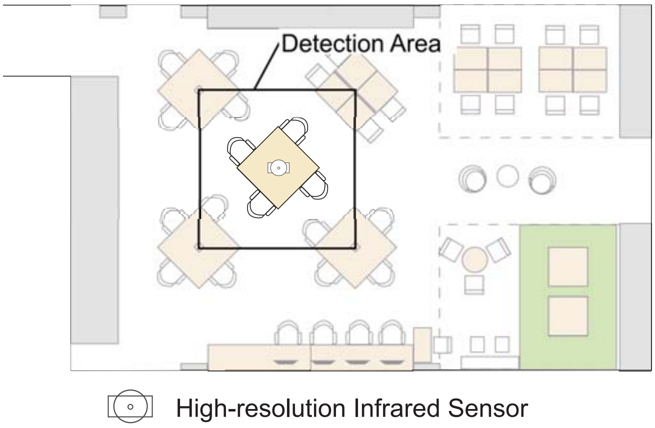


Figure 3: Experiment environment of the seat occupancy detection verification

5) Detect a pixel the number of whose adjacent occupied pixels is N or greater and classify it as occupied.
6) Detect a pixel the number of whose adjacent occupied pixels is less than N and classify it as unoccupied.

By repeating steps (2) to (6) every second, it is detected whether a seat is occupied or unoccupied by a worker. As noted above, a pixel representing a section with a large positive temperature difference is classified as occupied, and a pixel representing a section with a large negative temperature difference is classified as unoccupied. Each pixel is classified as either occupied or unoccupied. Because of the nature of this method using temperature difference, pixels are classified as unoccupied in their initial condition. If the temperature difference $\Delta T'$ is such that $-\Delta T < \Delta T' < \Delta T$, the previous occupied/ unoccupied status of the relevant pixel is retained unless the relevant temperature falls below the threshold T.

4. Verification Experiment for the Occupancy/Vacancy Detection Method Using High-resolution Infrared Sensors

4.1 Outline of Verification

We verified the effectiveness of the system detecting occupancy/ vacancy by using a high-resolution infrared sensor. Fig. 3 shows the experimental environment. A high-resolution infrared sensor was placed directly above the center of the desk in the center shown in Fig. 3. In this experiment, assuming PC work, 4 seats shown in Fig. 3 were sometimes occupied and sometimes not during 20 minutes. Occupancy or vacancy was detected every second to output the value 1 when a seat was occupied and the value 0 when it was unoccupied. Detection precision was confirmed by visual inspection using log data based on camera images.

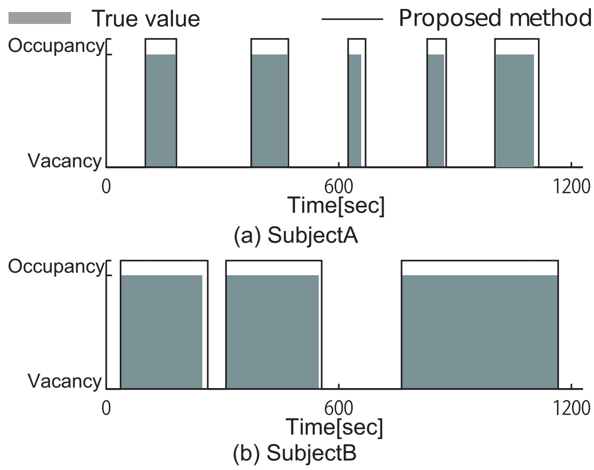


Figure 4: the history of occupancy (with PC)

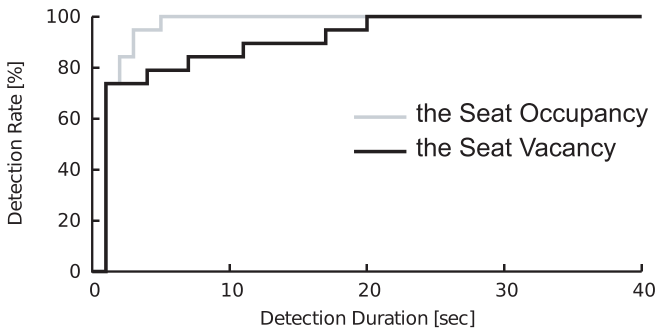


Figure 5: Relation between duration and detection rate (with PC)

4.2 Result of Verification in a PC Work Environment

This section examines the detection rates of occupancy and vacancy in a PC work environment by the proposed method and visual inspection as well as time required for detecting occupancy and vacancy. A PC was placed on the top of each desk at which each subject was to be seated. During 20 minutes, their seats were sometimes occupied and sometimes not.

Fig. 4 shows the transition of the occupied/ unoccupied status with Subjects A and B in accordance with the proposed method and visual inspection. Fig. 4 suggests that the method detects occupancy and vacancy well. Sometimes detecting vacancy took more time than detecting occupancy.

Next, we examine time required for detecting occupancy and vacancy respectively. Fig. 5 shows the relation between time required for detecting occupancy and vacancy and the corresponding detection rate. Fig. 5 indicates that the detection rate of occupancy reaches 100% 5 seconds after a seat is occupied by a subject. The detection rate of vacancy reaches 100% about 20 seconds after a seat becomes unoccupied by a subject.

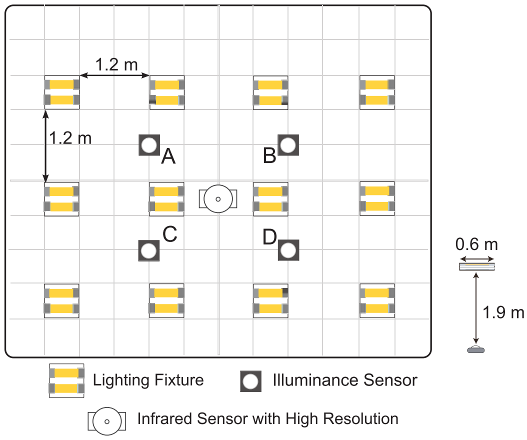


Figure 6: Experiment environment of the illuminance convergence experiment

4.3 Summary of Verification Result

The result of the verification experiment confirmed that the proposed method enables the detection of human occupancy and vacancy using high-resolution infrared sensor. In addition, it was found that detecting vacancy requires more time than detecting occupancy because the effect of a subject's body temperature on the desk and the chair lingers after the seat becomes unoccupied by the subject. It was confirmed that the detection rate of both occupancy and vacancy reaches 100% after 21 seconds.

In the Intelligent Lighting System, if the processing of occupancy and vacancy is delayed for 1 minute, assuming that the lighting luminance of the relevant lighting fixture changes from the maximum to 0 as a result of the processing, the corresponding difference in power consumption amounts to several Wh. Therefore, an impact of a 1-minute delay in the Intelligent Lighting System on power consumption is small, and the delay of 21 seconds is within the permissible range.

5. Intelligent Lighting Control Using High-resolution Infrared Sensors

5.1 Outline of Experiment

A verification experiment was conducted with the Intelligent Lighting System incorporating the occupancy/ vacancy detection method using high-resolution infrared sensors. Fig. 6 shows the environment of the verification experiment. As shown in Fig. 6, 12 lighting fixtures and one high-resolution infrared sensor were installed in the experiment environment. Four illuminance sensors were placed to simulate a real office. Assuming a typical office, pairs of desks placed opposite each other were placed apart from each other pairs.

In this experiment environment, an illuminance convergence experiment for the Intelligent Lighting System was conducted using the occupancy/ vacancy detection method with high-resolution infrared sensor. Table 2 shows target illuminance set for each illuminance sensor. In the illuminance convergence experiment, a target illuminance is set for a subject's illuminance sensor detecting occupancy and vacancy by the proposed method as shown in Table 2.

Table 2: Setup of the target value for experiment

Time [sec]	A [lx]	B [lx]	C [lx]	D [lx]
0	—	—	—	—
350	—	500	—	600
750	400	500	—	600
1050	400	500	350	600
2150	400	—	350	—
2500	—	—	—	—

We conducted an illuminance convergence experiment for the Intelligent Lighting System incorporating the occupancy/ vacancy detection by high-resolution infrared sensors and compared illuminance convergence results with each other.

The flow of control by the Intelligent Lighting System incorporating the proposed method is shown below.

1) Set the target illuminance of each illuminance sensor.
2) Turn on each lighting fixture at the initial luminance.
3) Obtain values measured by illuminance sensors and a power meter.
4) Determine whether a seat is occupied or unoccupied by the proposed method.
5) Calculate the values of the objective function.
6) Determine the next luminance in accordance with illuminance/ luminance influence and turn on each lighting fixture at the next luminance.
7) Obtain values measured by illuminance sensors and a power meter.
8) Determine whether a seat is occupied or unoccupied by the proposed method.
9) Calculate the values of the objective function referred to in step (5) under the new lighting condition.
10) Conduct a regression analysis with changes in luminance of each lighting fixtures and changes in the illuminance of each illuminance sensor to estimate illuminance/ luminance influence.
11) If the value of the objective function is improved, accept the next luminance. Otherwise, revert to the original luminance.
12) Go back to step (3).

5.2 Result

Let us state the result of the verification experiment performed in the experiment environment shown in Fig. 6. Fig. 7 shows the log of illuminance indicated by each illuminance sensor under the proposed method. In light of Fig. 7, it is confirmed that, by using the proposed method changes in the occupied/ unoccupied status for each worker were detected to change the luminance of the lighting fixture to meet the target illuminance. The illuminance value

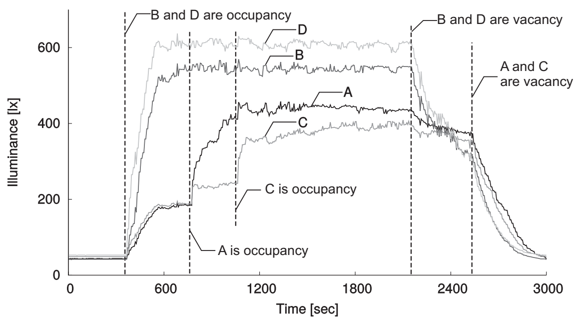


Figure 7: Illuminance history of lighting control based on the proposed detection approach

obtained by each illuminance sensor converged to the target illuminance 200 seconds after a change in the occupied/ unoccupied status. The system also subsequently worked to maintain the target illuminance. Illuminance again converged to the target illuminance demanded by Subjects A, B, and D 200 seconds after the presence of Subject A was detected at 750 seconds. After detecting the absence of Subjects B and D at 2150 seconds, illuminance converged to the target illuminance demanded by Subjects A and C, which confirmed that the target illuminance was maintained. As the result of the verification experiment shows that the occupied/ unoccupied status can be correctly managed by automatically changing the occupied/ unoccupied status using high-resolution infrared sensors, the Intelligent Lighting System using the proposed method can be said to be useful.

5.3 Verification of Power Consumption Reduction

The previous section confirmed that the Intelligent Lighting System incorporating the proposed method detects occupancy and vacancy to make illuminance converge to the target illuminance. We thus verify how power consumption changes by the occupancy rate to show the usefulness of the proposed method.

Fig. 8 shows the simulation environment. 30 lighting fixtures and 33 seats were placed, with an illuminance sensor installed on each seat. The minimum luminance of the lighting fixture was set to 0 cd, and the maximum luminance was set to 1,300 cd. The target illuminance of the illuminance sensor for each seat was set randomly from 300 lx to 700 lx at 50 lx interval. The occupancy rate of a worker's seat was set from 10% to 100% at 10% interval, and simulation was run 100 times for each rate set.

Fig. 9 shows the average power consumption for each occupancy rate. Since power consumption and luminance of lighting fixtures are in a linear relationship, on the basis of this linear relationship, power consumption for each occupancy rate was calculated by assuming power consumption to be 100% when the target illuminance is set to 750 lx for all illuminance sensors. Therefore, power consumption is about

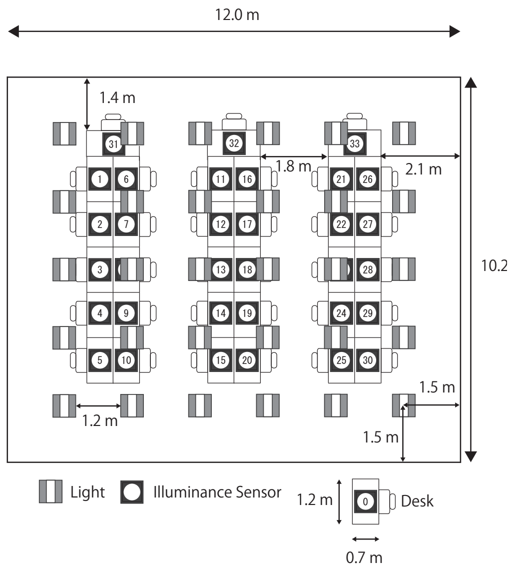


Figure 8: Simulation environment

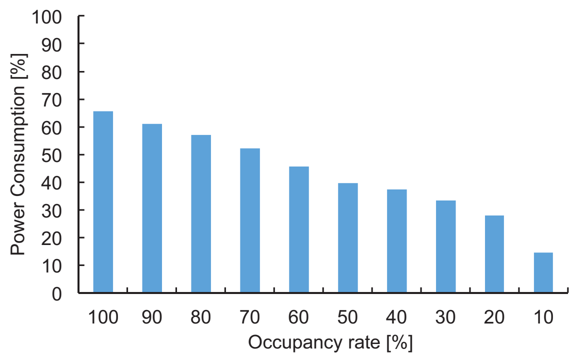


Figure 9: each occupancy rate and power consumption of the Intelligent Lighting Systems

118% when all lighting fixtures are turned on at uniform luminance with the minimum value of the illuminance on each worker's desk being set to 750 lx.

These results show that power consumption can be reduced by 35% even if the occupancy rate is 100% by introducing the Intelligent Lighting System. This reduction is made because, as noted above, the average illuminance in an office as a whole decreases by meeting the individual target illuminance for each worker. Fig. 9 also shows that power consumption decreases almost linearly as the occupancy rate decreases.

On the other hand, as shown in Fig. 2, power consumption increased during the periods when workers arrived at and left the office because the occupied/ unoccupied status was not changed appropriately. Simulation was thus run for the following three cases to compare power consumption.

1) The occupancy rate is 100%, and the Intelligent Lighting System is used. (The target illuminance for each

worker is uniform at 750 lx.)

2) The occupancy rate is 100%, and the Intelligent Lighting System is used. (The target illuminance for each worker is random and ranges from 300 lx to 750 lx.)
3) he occupancy rate is 70%. The number of workers is assumed to increase and decrease linearly, respectively, when they come to and leave the office. The Intelligent Lighting System is used. (The target illuminance of each worker is random and ranges from 300 lx to 700 lx.)

Cases 1 and 2 assume that workers do not change the occupied/ unoccupied status at all, come to the office at 8:00 at the earliest, and leave the office at 22:00 at the latest. The worker coming to the office first turns on lighting, and the one leaving the office last turns it off. Simulation is run using the Intelligent Lighting System when the occupancy rate is 100%.

Case 3 assumes that workers come to the office from 8:00 to 9:00 and leave the office from 18:00 to 22:00 and that the number of workers increases and decreases linearly during those periods of time. Since Fig. 2 indicates that the average seat occupancy by workers in office is considered to be 70% taking into account their attending meetings and going out, the occupancy rate is assumed to be 70%. By assuming power consumption to be 100% when the target illuminance is set to 750 lx for all illuminance sensors, power consumption was calculated for each case.

Fig. 10 gives the result of simulation for these cases. Fig. 10 shows that, in the Intelligent Lighting System, power consumption can be reduced by about 35% if the target illuminance for each worker (300 lx — 700 lx) is met individually compared with the case where, for all illuminance sensors, the target illuminance is set to 750 lx on the desk, which is generally recommended for an office. The reduction rate assumes that all workers are seated and never leave their seat. This is the basic energy saving performance of the Intelligent Lighting System.

It was found that power consumption reduced further by about 20% by introducing the Intelligent Lighting System incorporating the occupancy/ vacancy detection method using high-resolution infrared sensor proposed in this study. Thus it was found power consumption reduction amounted to about 55% if the Intelligent Lighting System was used, with each individual working with different preference for illuminance, and if occupancy and vacancy were detected correctly.

Note that power consumption is assumed to be 100% when the target illuminance on the desk is set to 750 lx using the Intelligent Lighting System. If uniform lighting were used without using the Intelligent Lighting System, power consumption would be 118% in Fig. 10. Therefore, the Intelligent Lighting System incorporating high-resolution infrared sensors proposed herein can reduce power consumption by about 73% compared to uniform lighting.

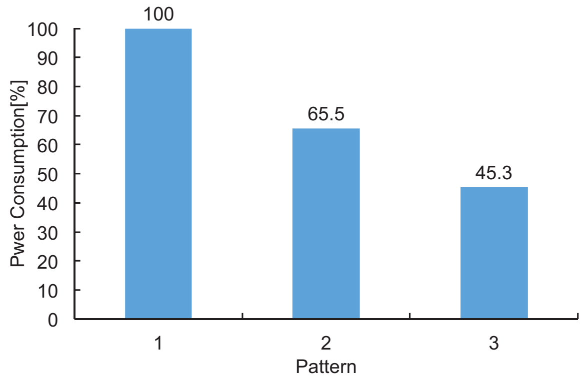


Figure 10: Simulation result

6. Conclusion

As a result of verification experiment in real office for practical realization, it is found that workers didn't change the occupied/ unoccupied status appropriately. Thus, there were lighting fixtures which provided brightness more than required even though there were no workers and energy saving deterioration the Intelligent Lighting System. Consequently, we propose a method for controlling changes using a high-resolution infrared sensor in the occupied/ unoccupied status of a worker's seat. We verified the effectiveness of proposed method. As a result we confirmed that the high-resolution infrared sensor can detect the occupied/ unoccupied status. We also confirmed that The Intelligent Lighting System incorporating the proposed method detects the occupied/ unoccupied status and realizes the illuminance demanded by a worker and reduces the power consumption. Moreover we confirmed that power consumption reduces by about 20% in simulation environment. Therefore the system detects the occupied/ unoccupied status automatically and improve the energy saving performance in the Intelligent Lighting System.

References

[1] Hiroshi Shimoda, Yoko Hattori, Kazuhiro Tomita, Misa Kawauchi, Hirotake Ishii, Fumiaki Obayashi, Masaaki Terano and Hidekazu Yoshikawa, "A Study on Environmental Control Method to Improve Productivity of Office Workers - Development of Productivity Evaluation Method, CPTOP", Human Interface Society, vol1, no.1322, pp.151-156, 2006

[2] P. R. Boyce, N. H. Eklund and S. N. Simpson, "Individual Lighting Control: Task Performance", Mood and Illuminance JOURNAL of the Illuminating Engineering Society, pp.131-142, 2000

[3] M. Miki, T. Hiroyasu and K. Imazato, "Proposal for an intelligent lighting system and verification of control method effectiveness", Proc IEEE CIS, pp.520-525, 2004

[4] S.Tanaka, M.Miki, T.Hiroyasu, M.Yoshikata, "An Evolutional Optimization Algorithm to Provide Individual Illuminance in Workplaces", Proc IEEE Int Conf Syst Man Cybern, vol.2, pp.941-947, 2009

Indoor Floor Map Construction with Video Survey

Tsz-lung Wong, Albert Kai-sun Wong, Ka-sing Lin,and Chin-Tau Lea
Emails: tlwongaa@ust.hk, eeablert@ust.hk,kslin@ust.hk, eelea@ece.ust.hk
ECE Department, The Hong Kong University of Science and Technology, Hong Kong

Abstract— *This paper describes a system which uses simple video surveys to automatically construct an indoor floor map for the purpose of supporting indoor people localization, tracking, and navigation applications. We show that this video-based system, using the concept of spatial segmentation through similarity matching followed by graph construction as developed in the WiFi RSS-based Intelligent Mobility Mapping System (IMMS) presented in an earlier work, is capable of constructing an indoor floor map based on simple video survey sequences alone. We show also that when used together with WiFI RSS, the video-based system can easily identify and label points-of-interest such as rooms along corridors, enriching the information content in the indoor map.*

Keywords: intelligent mobility mapping system, simultaneous localization and mapping, indoor localization and tracking, SIFT, optical characters recognition

1. Introduction

Indoor localization based on WiFi RSS (Received Signal Strength) fingerprinting has been a subject of research for over a decade. The most important advantage of using WiFi RSS is convenience. WiFi coverage is found in most indoor environments today and WiFi RSS can be measured without requiring permission. However, the off-line survey process to create a fingerprint, or a radio map, is tedious and time-consuming. Recently, an Intelligent Mobility Mapping System (IMMS), proposed in [1], applies the concepts of crowd-sourcing and Simultaneous Localization and Mapping (SLAM) to simplify the off-line survey step. The concept in IMMS is to make use of crowd-sourced WiFi RSS sequences or traces to identify location segments within the indoor coverage area by identifying highly similar segments across many traces. Afterwards, the identified location segments, which presumably are corridor segments in a grid-like indoor environment, can be used to construct a graphical indoor map through a graph construction algorithm. However, IMMS will create some false location segments and vertices. In addition, the orientation of the individual location segments are unknown, making matching of the graphical map to the physical map difficult.

Other researchers have conducted research into indoor localization using video devices. Compared to WiFi RSS, video is more accurate for identifying the exact number and positions of intersection points of corridor segments. On the other hand, video is processing and bandwidth intensive and possibly less convenient for the online localization/navigation phase. Our work intends to combine WiFi RSS with video for indoor floor map and radio map construction and localization.

In this paper, we describe methods to extract useful information from video sequences labeled with WiFi RSS. Our work first focuses on indoor physical spaces that are grid-like and can be modeled as an interconnection of corridor segments called Atomic Location Segments (ALSs) [1]. The information of interest is the number and location of the intersection points of corridors, and rooms and their room numbers alongside corridors. Accurate identification of corridor intersection points, which are combined with WiFi RSS data, will enable accurate automatic reconstruction of the indoor map, and labeling of rooms along corridors will enrich the information content of the indoor map.

This paper is organized as follows. Section 2 gives an overview of the system. Section 3 describes the algorithm to find the intersection points. Section 4 describes the unique ALS identification. Section 5 describes the floor map construction. Section 6 describes the algorithm to find and recognize room numbers. Experimental results are shown in section 7.

2. System Overview

Figure 1 shows a simple floor map which illustrates the concept of Atomic Location Segment (ALS), which are corridor segments, and Breaking Point (BP), which are intersections of corridors. A BP is where people can make turn into different corridor segments. We call an intersection a BP because it is where the similarity between two WiFi RSS or video traces may end if the two WiFi RSS or video signal collectors turn into different corridor segments. An ALS is the corridor segment between two BPs.

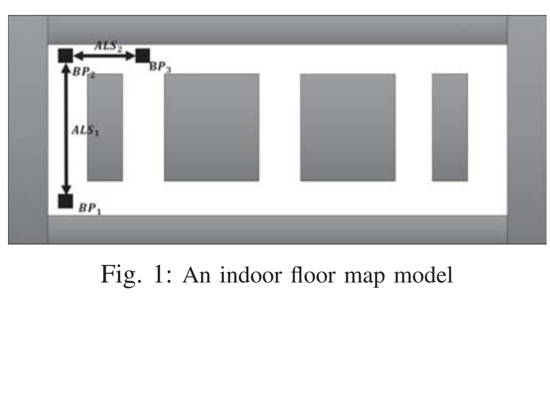


Fig. 1: An indoor floor map model

Our indoor map construction involves three steps:

1. Data collection: As described in the introduction, our overall objective is to combine video and WiFi RSS to perform indoor floor and radio map constructions to support WiFi fingerprint based indoor localization. The primary focus of this paper is on using video data to enhance the map construction. To enhance the detection of corridor intersections and points of interest, we will record two video clips. The first clip is taken with the surveyor walking causally around the coverage area without pausing, making turn (randomly left or right) whenever an intersection is encountered, until most of the corridor segments in the coverage area is traversed more than once. In taking the second clip, the surveyor pauses at all or selected rooms along the corridors to direct the camera at the doorplate of each for several seconds. Both clips are taken while WiFi RSS measurements are collected at an interval of one second. The timestamp for each frame in the video clips and for each measurement in the RSS trace are synchronized.

2. Information extraction: In this stage, for clip 1, each encounter of an intersection and the surveyor's turning direction is identified and the occurrence time is marked. For clip 2, each pausing and the room number captured is identified. The methods of Scale Invariant Features Transform (SIFT) and Optical Characters Recognition (OCR) are applied.

3. Map construction: In this stage, video clip 1 is divided into segments based on the BPs identified. Video segments that are taken at the same corridor segment are identified so that a list of unique corridor segments called ALSs can be created. The list of unique ALSs, their connectivity relationships through BPs, and turning directions are used to construct the floor map. The room numbers identified in clip 2 are used to mark points of interest in the floor map.

3. Intersections Finding Algorithm (IFA)

For our indoor map construction, accurate identification of the set of breaking points is very important. In the WiFi RSS-based approach in [1], some superfluous BPs are often identified, resulting in false location segments. Another problem with the WiFi RSS approach in [1] is that the right-handedness or left-handedness of intersections of location segments is not determinable, and hence the graphical planar floor map constructed is not unique. Thus, unless the graphical floor map is manually aligned with the actual floor map, the value of the graphical floor map is limited in navigation applications as it cannot be used to guide a user to make right turn or left turn. The objective of IFA here is to use video clips to determine the set of BPs accurately, and to record the direction of each turn.

3.1 Scene matching

In the video trace recorded as the surveyor walks through the coverage area, scenes in successive frames are expected to change slowly unless the surveyor is traversing a BP. This means that we can find the BPs by monitoring changes in successive video frames. We have considered different ways for measuring changes in video frames. We first tried a relatively simple method which is to compute the correlation between frames (e.g.,[3], [4]). However, this does not work well since the correlation can be easily affected by small changes in the camera position or orientation. A more robust method is to match two frames based on finding some interest points in the frames and creating a descriptor for these interest points. We can then match two frames by matching their descriptors. There are many research studies and many local descriptors have been proposed, including differential invariants [7], steerable filters [6], SIFT [2], PCA-SIFT [5] and SURF [8]. All these build a descriptor to model an image patch around some interest points. Over time, SIFT, PCA-SIFT and SURF are generally accepted to be superior. [9] gives a comparison of SIFT, PCA-SIFT and SURF. [10] also compares the performance between SIFT and SURF. In terms of speed, SURF gives the fastest runtime while the SIFT has the best performance in invariants, especially in scale, rotation and blurring. In our study, the run time is not a critical concern as BPs finding is performed offline. Therefore, we choose the SIFT descriptor for matching our video frames.

3.2 Review of the SIFT Algorithm

SIFT is a method to extract invariant features from images[2]. There are four major stages in SIFT:

1. Scale-space extrema detection: A Difference-of-Gaussian (DOG) function is used to find all potential interest points by searching over all scales and image locations.

2. Keypoint localization: A model is applied to all potential interest points to select keypoints based on their stability.

3. Orientation assignment: Based on the local image gradient directions, one or more directions are assigned to each keypoint.

4. Keypoint descriptor: A model is applied to each keypoint and its neighbours to transform them into a matrix form to represent the keypoint.

3.3 Turning Point Position and Turning Direction

Since the camera records at 30 frames per second, matching each frame with its next frames will lead to very high computational cost. Furthermore, the changes over too short a time interval may not be large enough to be detectable. Therefore, we choose only one in every $t = 10$ frames and compare it with the $d_1 = 40$ frames later. Figure $2(a)$ shows the number of matched interest points between the f_i and f_{i+d_1} as a function of time in the video. By applying a threshold T and a smoothing function, we obtain the binary graph shown in Figure $2(b)$, in which each zero region represents the a BP region during which the surveyor is making a turn.

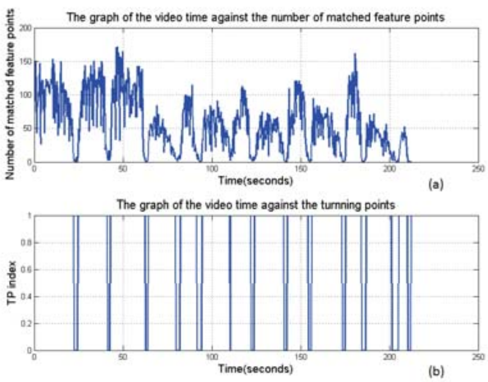


Fig. 2: BPs Finding. (a) Number of matched interest points as function of time. (b) Binarized Graph.

After finding a BP, the next step is to determine the turning direction of the surveyor at this BP. We assume that there can be only two possible turning directions (left L, right R), and compute the average shifting of the matched interest points inside each BP region in Figure $2(b)$. For a BP region covering N frames, for every $t = 10$ frame we sample a sample frame f_i and compute the average shifting of the frame $f_i + d_2$ where $d_2 = 5$. d_2 is smaller than d_1 so that shifting of shifting in the frames can be tracked. Assuming there are n matched interest points between frame f_i and f_{i+d_2}, the shift of frame f_{i+d_2} is computed as:

$$s_{i+d_2} = \frac{1}{n}\sum_{j=1}^{n}(g_{i+d_2}(x_j, y_j) - g_i(x_j, y_j)), \tag{1}$$

where $s_i(i + d_2)$ is for shift for frame $i + d_2$ and $g_i(x_j, y_j)$ is the coordinate of the $j - th$ interest point in the $i - th$ sample frame.

The shifting in a particular turning point region can be computed as:

$$S = \frac{1}{N/t}\sum_{i=1}^{\frac{N}{t}-1} s_{i+d_2}, \tag{2}$$

A negative S means the surveyor is turning to left while a positive S means the surveyor is turning to right.

4. Unique ALS and BP Identification

As described in the system overview, the segment between two intersection points is called an ALS. During the survey process, the surveyor go around the survey area randomly, and makes turn whenever an intersection is encountered, until most of the ALSs are passed through by the surveyor in the same direction more than one time. Identification of repeated ALSs will enable us to construct the floor map.

Assume K BPs are found by IFA, we divide the video trace into $K + 1$ individual video segments and label them as $(l_i; i = 1, ..., K + 1)$. Then, we determine if two video segments l_i and l_j are taken at the same ALS by computing the average number of matched interest points (A_m) between these two segments, as described in **Algorithm 1**. In the algorithm, a unified number of N_s frames are selected at a regular interval from each of the two segments for comparison. N_s is chosen as $min(C, N_{short}/30)$ where $C = 10$ is a constant and N_{short} is the number of frames in the shorter of l_i and l_j. This means at most 10 frames totally or one frame per second are used for the comparison. The value of A_m between all pairs of video segments are computed. We assume that the surveyor is traversing the same corridor segment in the same direction if A_m for the two corresponding video segments is larger than a threshold.

Algorithm 1 Function:TestVideoSeg

function TESTVIDEOSEG(l_i, l_j)
  $C = 10$
  $N_s = min(C, N_{short}/30)$
  $Am = \frac{\sum_{k=1}^{C} match(l_{i_k}, l_{j_k})}{C}$ ▷ where l_{i_k} and l_{j_k} are the k-th selected frames in trace l_i and l_j, and $match()$ is a function in SIFT
  if $Am > Threshold$ **then**
    $Tru = 1$ ▷ Tru is the Boolean type variable.
  else
    $Tru = 0$
  end if
  return Tru
end function

The video-based approach allows us to recognize BPs without ambiguity and to identify with very high accuracy video segments taken while surveyor is travelling in the same corridor in the same direction. However, if the surveyor traverses the same corridor segment in opposite directions, the images observed by the camera would be quite different and the two video segments will not be recognized as a match. To deal with this reverse path problem, we make use of the sequence of WiFi RSS values recorded and reuse the function TestHCP described in [1]. This function examines the correlation of two WiFi RSS trace segments and returns as output whether the two WiFi RSS trace segments are in the same ALS by detecting if there is a "high correlation pattern" in the correlation matrix that extends in either the +45 or -45 degree direction corresponding to two trace segments traversing the same corridor segment in the same or opposite direction.

With the addition input from WiFi, we can proceed to identify all the unique atomic location segments (ALS) contained in the video. We consider the $K+1$ video segment one by one, as described in **Algorithm 2**. If a segment l_i has not been identified as a match to any segment already examined, we assign a new ALS ID, and l_i will become the reference segment of this new ALS ID. The directionality of the reference segment is labelled as +1. For a segment l_j, if $TestVideoSeg(l_i, l_j)$ is true in **Algorithm 1**, it will be given the same ALS ID and assigned a directionality of +1. On the other hand, if $TestHCP(l_i, l_j)$ is true but $TestVideoSeg(l_i, l_j)$ is false in **Algorithm 1**, it will be given the same ALS ID but assigned a directionality of -1. After all video segments are examined, the result is a set of unique ALSs $E = \{E_1, ..., E_{N^e}\}$, where N^e is the number of unique ALSs identified. Knowing the set of unique ALSs, we can further identify the set of unique BPs which connects to the unique ALS. Let the set of unique BPs be $U = \{U_1, ..., U_{N^u}\}$. Further, we produce through **Algorithm 3** a vector $D = \{D_1, ..., D_{N^d}\}$ that describes the turning direction between any two ALSs that are connected as follows: D_i is a vector that includes the two connected ALSs (E_i and E_j), the BP (U_k) that connects them, and a turning direction indicator T_{ij} which is either L or R.

Algorithm 2 Unique ALS Identification

$e = 1$
for ($i = 1$,$i <= N+1$,$i++$) **do**
  if l_i `is unlabeled` **then**
    $l_i = +e$ ▷ The first direction found is label as positive.
    for ($j = i+1$,$j <= N+1$,$j++$) **do**
      if l_j `is unlabeled` **then**
        $V_s(i,j) = TestVideoSeg(l_i, l_j)$
        $cpc(i,j) = TestHCP(l_i, l_j)$
        if $V_s(i,j) = 1$ and $cpc(i,j) = 1$ **then**
          $l_j = +e$
        else if $V_s(i,j) = 1$ and $cpc(i,j) = 0$ **then**
          $l_j = -e$
        end if
      end if
    end for
    $e = e + 1$
  else if l_i `is labeled` **then**
    for ($j = i+1$,$j <= N+1$,$j++$) **do**
      if l_j `is unlabeled` **then**
        $V_s(i,j) = TestVideoSeg(l_i, l_j)$
        $cpc(i,j) = TestHCP(l_i, l_j)$
        if $V_s(i,j) = 1$ and $cpc(i,j) = 1$ **then**
          $l_j = l_i$
        else if $V_s(i,j) = 1$ and $cpc(i,j) = 0$ **then**
          $l_j = -l_i$
        end if
      end if
    end for
  end if
end for
return $N^e = e$

Algorithm 3 Turning Direction Vector Generation Algorithm

$j = 1$, $k = 1$
for ($i = 1$,$i < N^e$,$i++$) **do**
  $D(i,1) = E_i$
  $D(i,2) = E_j$
  $D(i,3) = U_k$
  $D(i,4) = T_{ij}$ ▷ T_{ij} is the turning direction either left or right.
  $j = j+1$, $k = k+1$
end for

5. Floor Map Construction

With the set of vertices U, the set of unique edges E and the set of turning vectors D, the floor map construction algorithm aims to create a planar graph $G = (E, U, D)$ that is more intuitive for human observers to read. There are three steps in the algorithm: (1) Depth First Block Search (DFBS), (2) path search, and (3) straight-line and turning direction embedding. Steps (1) and (2) follow [1] and step (3) incorporates information on turning directions to construct a floor map that would more closely resemble the actual physical map.

1. Depth First Block Search

The step of DFBS is based on Tarjan's DFS block search algorithm [16]. We also follow the notation in [1]. DFS starts from a vertex of G with the highest node degree and chooses an edge to follow. Traversing the edge leads to new vertex. If it reaches the end vertex in the path, it goes back to the preceding vertex and goes to another unexplored edge. It stops when all the edges are explored. DFS labels each vertex with a DFS number $DFSN(v)$ and create a spanning tree for the path search in step (2). For any vertices v and w in U, the spanning tree is constructed by a set of arcs $v \rightarrow w$, where $DFSN(v) < DFSN(w)$, and a set of fronds $v \dashrightarrow w$, where $DFSN(v) < DFSN(w)$.

Moreover, DFS assigns each vertex a number called the low point value (LPV) to determine whether a 'block' is a biconnected component of the graph or not. A biconnected component of graph G is a subgraph G_i. such that the remaining graph remains connected if the biconnected component is removed. For any vertex ($v \in U$), its low point value is defined as:

$$\begin{aligned} LPV(v) =& min(\{DFSN(v)\} \cup \{LPV(w) \mid v \rightarrow w\} \\ & \cup \{DFSN(w) \mid v \dashrightarrow w\}), \end{aligned} \quad (3)$$

Initially, the low point value in a vertex v is set equal to its $DFSN(v)$. After all of the low point values are calculated, we look for the vertices with $DFSN(v) \leq LPV(v)$, excluding the start vertex of the spanning tree. A path including these vertices is grouped into a sub-graph G_i called a block. Figure 3 shows an example of spanning tree. Each node is a vertex in the graph, and the numbers next to each node are the DFS number and low point value respectively. There are three blocks in the example. The largest block is the original graph with edges $(2, E)$ and $(1, S)$ excluded, and $(2, E)$ and $(1, S)$ each forms its own block, while S and E is the start and end BPs of the video path. The ALS ID of each edge can be identified by its start and end vertices and is labelled by E_1 to E_{18} in Figure 3.

2. Path Search

In path search, we search all blocks one by one. In the beginning, all the vertices and edges in the spanning tree are labeled as unexplored. We start from the vertex with the smallest DFS number in a block and mark it as explored. Then we extend the path to the next unexplored edge that is connected to the vertex until all of the vertices in the block are explored. The output of the search of each block is a path that covers all vertices in the block. The search step is completed when all the blocks are explored.

3. Planar Embedding

In this step, we use the paths identified in step (2) to draw the floor map. We assume that all edges/corridor segments are straight-lines. While [1] uses a planar layout algorithm based on [17], here we can simply make use of the direction relationships described in the matrix D. The result is a draft floor map that shows the relationship between vertices and edges. An example is shown in Figure 8.

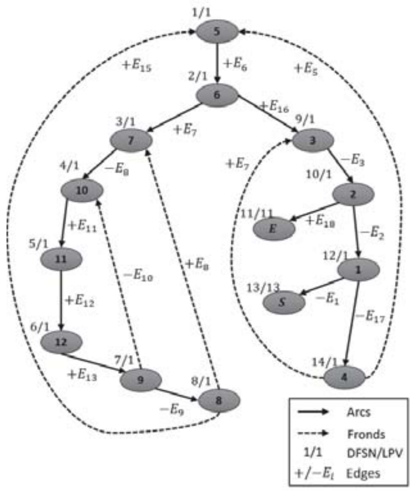


Fig. 3: A sample of spanning tree.

6. Room Number Finding Algorithm

Having a floor map that is labeled with room numbers is useful for navigation and other applications. In our Room Number Finding Algorithm, we make use of the WiFi RSS trace associated with clip 2 to reduce the search process in finding potential doorplate frames in the survey video. We have also applied different methods for preprocessing the image for character recognition.

6.1 Doorplate Frame Extracting

First, we need to find the video frames in which a doorplate may be present. To increase the searching speed, we get help from the WiFi RSS data. We compute a single-trace correlation (STC) which is the autocorrelation of the WiFi RSS trace to identify the times that the surveyor is pausing in front of a doorplate. The curve in Figure 4 shows the autocorrelation of the WiFi RSS sequence. Peaks in the autocorrelation curve reflect the potential times at which the surveyor is pausing in front of a doorplate. From the figure, we see that a video clip with more than 18000 frames can be reduced to about 80 frames for which the surveyor is potentially pausing. It needs to promise that there is a doorplate template in order to find the accurate doorplate frames inside the potential frames. The number of matching interest points between the template and each found frame will be calculated. The frame will be determined to be a doorplate frame if the number of matched interest points is larger than a threshold.

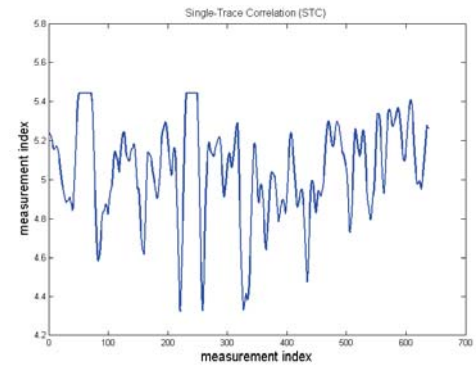


Fig. 4: The autocorrelation of the WiFi RSS measurements of a path.

6.2 Character Recognition

Character recognition is the final step of the Room Number Finding Algorithm. We use optical character recognition (OCR) [11] technique to perform the character recognition. There are standard steps in OCR for off-line characters recognition [14], [11], [13]:

1. Preprocessing: Before applying any feature extraction technique, the character images must be converted into a black-and-white image. The recognition performance is often highly dependent on this step, and we will discuss in greater details this later.

2. Location and segmentation: Segmentation is the segregation of words into individual alphabets/characters. The segmentation step typically works under the assumption that the characters are not connected together.

3. Feature Extraction, or template-matching and correlation method: There are two methods to complete the final step of OCR. The first method is to extract features contained in the character represented as a binary matrix. The second method is to compute the distance between the character with a set of templates to find the best match. We use the second method and the templates are provided in [12].

6.3 Binarization

The preprocessing, especially the binarization, is extremely important to the OCR performance [11]. Binarization methods can be classified as local or global. Global binarization methods include the Fixed Thresholding Method, Otsu Method and Kittler Method, while local binarization methods include the Niblack Method, Adaptive Method, Sauvola Method and Bernsen Method [15]. As [15] points out that the Otsu Method and Sauvola Method produce the best result for global binarization and local binarization respectively, we will compare only these two methods and choose one to perform the binarization step. Figure 5 shows that the Otsu Method gives a better binarization quality in our case. Hence, the Otsu Method is used as the binarization method.

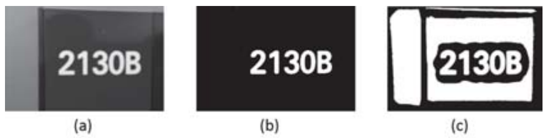


Fig. 5: The image under different binarization methods. (a)The original gray image. (b)The image under binarization with the Otsu Method. (c)The image under binarization with the Sauvola Method

7. Experiment and Result

The experiment took place in the lab area on the second floor of the academic building of our university. The actual floor map of the area is shown in Figure 6. In the experiment, a student equipped with an LG Nexus 5 walks at a relatively constant speed around the area. Two traces, one for the Intersections Finding Algorithm and the other for the Room Number Finding Algorithm, are recorded. Both videos are in $720p$ and WiFi RSS values are recorded along with them. The first trace starts from point A, and covers the area in a zigzag pattern until ending at point B. The second trace travel through the area randomly and stops for seconds to capture doorplates.

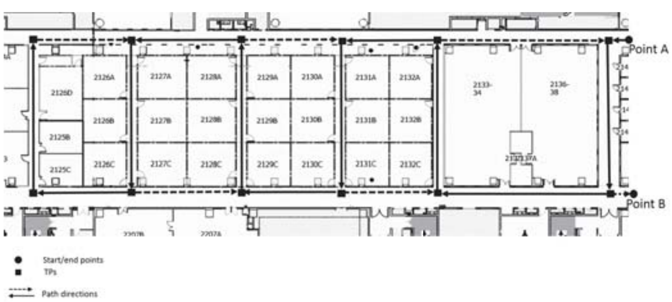


Fig. 6: Physical topology of corridors in survey area

7.1 Experimental Result for TPs Finding Algorithm

The BPs and turning directions in the trace are determined by the Intersections Finding Algorithm. As shown in Figure 7, the trace passes through 12 different BPs a total of 22 times, passing through 10 BPs twice. The result of IFA is shown in Figure 7. A threshold is applied into the curve in Figure $7(a)$ to obtain the curve in Figure $7(b)$. Figure $7(b)$ shows that all the BPs, each indicated by a zero region of the curve, have been found by the algorithm.

The turning time of the BPs, taken as the middle of the zero regions, are shown in Table 1. The turning directions at each BP is also shown in the table, where 1 indicates a left turn and 0 a right turn.

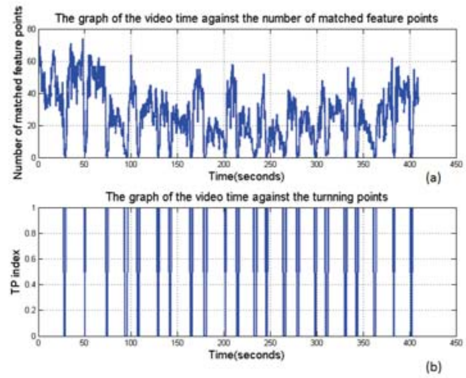


Fig. 7: BPs Finding. (a) Number of matched interest points as function of time. (b) Binarized Graph.

Table 1: The turning time and its turning direction in the trace

The n^{th} BP	Time (seconds)	Turning direction (1/0)
1	28.7	1
2	50.4	0
3	74.2	0
4	94.7	1
5	107.9	1
6	129.4	0
7	142.2	0
8	164.9	1
9	179.9	1
10	201.9	0
11	215.0	0
12	233.9	0
13	246.2	0
14	265.5	1
15	279.7	1
16	298.9	0
17	311.0	0
18	331.2	1
19	343.5	1
20	362.7	0
21	283.2	0
22	402.0	1

7.2 Experimental Result for Unique ALSs Labeling and Map Construction

Table 1 shows all the BPs in video clip 1 and table 2 shows the level of matching of the segments between the BPs. The high matching values are shaded, indicating that the two corresponding segments are the same ALS. Each unique ALS is assigned an ID. From the table, 5 pairs of segments are repeated segments so there are 18 unique ALSs. After identifying the unique ALSs, the data will be passed to the WiFi-RSS based system to construct the radio map. The resulting draft floor map shown in Figure 8 can also easily construct by followed the steps in floor map construction.

7.3 Experimental Result for Room Number Finding Algorithm

There are 34 rooms in the survey area, as shown in Figure 6. The surveyor goes around records all the doorplates in video clip 2. The result of the room number finding algorithm is shown in Table 3. There are 3 doorplates

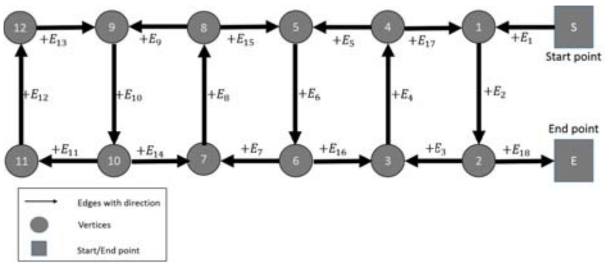


Fig. 8: The resulting graphical floor map

Table 2: The average matched interest point(A_m) between two trace segments

Segments	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1 (ALS_1)	-	0.6	0.5	0.4	1.4	0.8	0.6	0.5	1.2	0.4	1.2	0.9	0.7	0.1	0.3	0.5	1	0.4	0.4	0.7	0.3	0.3	0.5
2 (ALS_2)	0.6	-	2.1	2.2	1	0.9	1.1	1.3	0.3	1.5	1.7	1.5	1.6	0.9	0.3	2.1	0.8	0.8	0.6	1.1	0.9	47.6	0.6
3 (ALS_3)	0.5	2.1	-	0.5	0.9	0.1	1.8	0.9	0.6	0.8	2.4	1.1	2.5	0.3	0.5	0.7	1	0.3	0.8	1.1	1.8	2	0.6
4 (ALS_4)	0.4	2.2	0.5	-	0.9	1.4	0.3	3.2	0.5	0.7	0.6	1.9	1.1	1.5	0.4	2	0.2	1.4	0.2	20.5	0.7	1.2	0.5
5 (ALS_5)	1.4	1	0.9	0.9	-	0.1	0.5	0.4	3.8	0.4	0.4	0.7	1.4	0.5	0.7	0.1	0.7	0.4	0.8	0.7	0.4	0.8	0.4
6 (ALS_6)	0.8	0.9	0.1	1.4	0.1	-	1.7	3.6	0.6	4.2	0.9	1.5	1.2	6.6	0	5.6	0.4	58.1	0.2	1.6	1	2	0.3
7 (ALS_7)	0.6	1.1	1.8	0.3	0.5	1.7	-	0.5	1	0.6	1.3	1.1	1.6	0.1	0.7	0.8	1	0.2	0.6	0.4	0.8	1	0.7
8 (ALS_8)	0.5	1.3	0.9	3.2	0.4	3.6	0.5	-	0.5	2	0.2	1.8	1	2	0.4	39.2	0.9	2.5	0.6	2.4	0.3	1.7	0.4
9 (ALS_9)	1.2	0.3	0.6	0.5	3.8	0.6	1	0.5	-	0.3	0.5	1.2	1.1	0.1	0.7	0.1	0.4	0.1	1.7	0.2	0.3	0.7	0.3
10 (ALS_{10})	0.4	1.5	0.8	0.7	0.4	4.2	0.6	2	0.3	-	0.8	1.8	0.7	39	0.3	2.2	0.4	3.7	0.7	1.5	0.6	1.5	0.2
11 (ALS_{11})	1.2	1.7	2.4	0.6	0.4	0.9	1.3	0.2	0.5	0.8	-	1.3	1.1	1	0.3	0.8	1.7	0.3	0	1.6	0.3	1.2	1
12 (ALS_{12})	0.9	1.5	1.1	1.9	0.7	1.5	1.1	1.8	1.2	1.8	1.3	-	1.4	0.6	0.1	1.7	0.7	1	0.5	1.8	0.7	2	0.6
13 (ALS_{13})	0.7	1.6	2.5	1.1	1.4	1.2	1.6	1	1.1	0.7	1.1	1.4	-	0.2	0.1	0.6	2.6	0.5	0.5	0.3	0.5	0.9	0.3
14 (ALS_{10})	0.1	0.9	0.3	1.5	0.5	6.6	0.1	2	0.1	39	1	0.6	0.2	-	0.3	2.7	0.5	5.4	0.2	0.8	0.5	1.9	0.3
15 (ALS_{14})	0.3	0.3	0.5	0.4	0.7	0	0.7	0.4	0.7	0.3	0.3	0.1	0.1	0.3	-	1	1.1	0.4	2.6	0.7	0.7	1.3	0.7
16 (ALS_6)	0.5	2.1	0.7	2	0.1	5.6	0.8	39.2	0.1	2.2	0.8	1.7	0.6	2.7	1	-	0.7	3.4	0.2	2	0.2	1.5	0.2
17 (ALS_{15})	1	0.8	1	0.2	0.7	0.4	1	0.9	0.4	0.4	1.7	0.7	2.6	0.5	1.1	0.7	-	0.6	0.5	0.3	0.7	0.4	0.3
18 (ALS_6)	0.4	0.8	0.3	1.4	0.4	58.1	0.2	2.5	0.1	3.7	0.3	1	0.5	5.4	0.4	3.4	0.6	-	0.2	1.4	0.5	1.8	0.5
19 (ALS_{16})	0.4	0.6	0.8	0.2	0.8	0.2	0.6	0.6	1.7	0.7	0	0.5	0.5	0.2	2.6	0.2	0.5	0.2	-	0.5	0.6	0.6	0.4
20 (ALS_4)	0.7	1.1	1.1	20.5	0.7	1.6	0.4	2.4	0.2	1.5	1.6	1.8	0.3	0.8	0.7	2	0.3	1.4	0.5	-	0.6	0.6	0.4
21 (ALS_{17})	0.3	0.9	1.8	0.7	0.4	1	0.8	0.3	0.3	0.6	0.3	0.7	0.5	0.5	0.7	0.2	0.7	0.5	0.6	0.5	-	1	0.4
22 (ALS_2)	0.3	47.6	2	1.2	0.8	2	1	1.7	0.7	1.5	1.2	2	0.9	1.9	1.3	1.5	0.4	1.8	0.6	2.4	1	-	0.7
23 (ALS_{18})	0.5	0.6	0.6	0.5	0.4	0.3	0.7	0.4	0.3	0.2	1	0.6	0.3	0.3	0.7	0.2	0.3	0.5	0.4	0.4	0.4	0.7	-

discarded because of failed matching with the template. Two of these are because of low brightness in the environment, as the performance of SIFT is highly sensitive to differences in brightness. The third is because of a low number of matched interest points with the template.

Table 3: The number of doorplates under different stage of the algorithm

The stage of algorithm	Percentage of doorplate under the stage
Correct output doorplates	90.6%
Incorrect output doorplates	0%
Objective doorplates discards by matched with template	9.4%
Objective doorplates discards by STC	0%

8. Conclusion and Future Work

We have proposed to use video data to assist crowd-sourced WiFi RSS data in floor map and radio map construction for indoor WiFi localization applications. This paper shows that intersections and turning directions can be accurately determined from video data, by using SIFT to match interest points in successive video frames. This paper focuses on indoor areas that are grid-like, with corridor segments joining at intersected. The next step of our work will be to include different types of areas such as open flow and enclosed areas.

References

[1] X. Luo, A.K.S. Wong, M. Zhou, X.N. Zhang, and C.T. Lea,"Automatic Floor Map Construction for Indoor Localization,"*The Tenth Advanced International Conference on Telecommunications*, 2014, p.155–160.

[2] David G. Lowe,"Distinctive image features from scale-invariant key-points,"*International journal of computer vision*,vol. 60, pp. 91–110, 2004.

[3] Armin Gruen,"Adaptive least squares correlation: a powerful image matching technique,"*South African Journal of Photogrammetry, Remote Sensing and Cartography*,vol. 14, pp. 175–187, 1985.

[4] F. Zhao, Q.M. Huang, and W. Gao,"Image Matching by Normalized Cross-Correlation,"*IEEE International Conference on Acoustics, Speech and Signal Processing*,vol. 2, May 1985.

[5] Yan Ke, and R. Sukthankar,"PCA-SIFT: a more distinctive representation for local image descriptors,"*IEEE Computer Society Conference on Computer Vision and Pattern Recognition*,vol. 2, pp. 506–513, Jun.2004.

[6] W.T. Freeman, and E.H. Adelson,"The Design and Use of Steerable Filters,"*IEEE Transactions on Pattern Analysis and Machine Intelligence*,vol. 13, pp. 891–906, 1991.

[7] J.J. Koenderink, and A.J. van Doorn,"Representation of local geometry in the visual system,"*Biological Cybernetics*,vol. 55, pp. 367–375, 1987.

[8] H. Bay, A. Ess, T. Tuytelaars, and L.V. Gool,"Speeded-Up Robust Features (SURF)"*Computer Vision and Image Understanding*,vol. 110, pp. 346–359, 2008.

[9] L. Juan, and O. Gwun,"A comparison of sift, pca-sift and surf"*International Journal of Image Processing (IJIP)*,vol. 3, pp. 143–152, 2009.

[10] N.Y. Khan, B. McCane, and G. Wyvill,"SIFT and SURF Performance Evaluation against Various Image Deformations on Benchmark Dataset" *International Conference on Digital Image Computing Techniques and Applications (DICTA)*,pp. 501–506, Dec.2011.

[11] Eikvil Line,"Optical character recognition",1993.

[12] (2006) OCR (Optical Character Recognition) website. [Online]. Available: http://www.matpic.com/

[13] G. Vamvakas, B. Gatos, and S.J. Perantonis,"Handwritten character recognition through two-stage foreground sub-sampling"*Pattern Recognition*,vol. 43, pp. 2807–2816, 2010.

[14] N. Arica, and F.T. Yarman-Vural,"An overview of character recognition focused on off-line handwriting"*IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*,vol. 31, pp. 216–233, 2001.

[15] N.K. Garg,"Binarization Techniques used for Grey Scale Images"*International Journal of Computer Applications*,vol. 71, 2013.

[16] R. Tarjan,"Depth-first search and linear graph algorithms"*SIAM journal on computing*,vol. 1, pp. 146–160, 1972.

[17] P. Rosenstiehl, and R. Tarjan,"Rectilinear planar layouts and bipolar orientations of planar graphs"*Discrete & Computational Geometry*,vol. 1,pp. 343–353, 1986.

An Approach for the Emerging Ontology Alignment based on the Bees Colonies

Carlos Rangel[1], **José Aguilar**[1,2], **Mariela Cerrada**[1,3], **Junior Altamiranda**[1]
[1]CEMISID, Universidad de Los Andes, Mérida, Mérida, Venezuela
[2]Prometeo Researcher, Universidad Técnica Particular de Loja, Loja, Loja, Ecuador
[3]Prometeo Researcher, Universidad Politécnica Salesiana de Cuenca, Ecuador

Abstract - *This work is a research in the field of Ontologies Integration from the point of view of Ontology Mining based on Services. Specifically, the work focuses on an automatic suggestion of ontological alignments for users. The Ontology Mining area (OM) is very recent, due to the current trend of using ontologies as a mechanism for representing knowledge, which has created a wide field to explore and extract knowledge. The problem lies in the comparison of existing ontologies, in order to use them together, that is, finding their semantic equivalences. There are different techniques for ontologies alignment as a form of ontological comparison based on the matching of the concepts, which is a fundamental process in the ontologies integration. Each alignment technique uses different strategies; based on specific principles, which make it more adequate for a particular context. This paper proposes an automatic approach for comparison and selection of alignment techniques, given a group of ontologies, based on the ABC algorithm, which is inspire by bee colonies. The approach uses as comparison and selection criteria the execution time, the number of aligned concepts, and the number of times the colony chooses each technique (this is due to the stochastic approach of the ABC algorithm).*

Keywords: Ontologies Alignment, Alignment Techniques, Ontologies Integration, Artificial Bee Colonies, Ontology Mining.

1 Introduction

Data Mining (DM) allows the knowledge extraction from historical data. The new trend, giving semantic content to data, gives new branches in DM. Particularly, the Semantic Data Mining (SDM), which allows using DM techniques to extract knowledge from databases, in order to explore the semantic content of the web (called Semantic Web Mining (SWM)). Finally, explore or extract knowledge in the form of ontologies (called Ontology Mining (OM)).

Due to the big development of the ontologies, it is emerged the need for integrating them. In the literature, as a first step to compare the ontologies that will be integrated, it has been develop several ontologies alignment techniques. Some of these techniques are classes' structure, distance of names, and the name properties analysis, among others.

The problem is to determine which alignment technique should be used at a given context, which alignment technique is the best for the ontologies to be integrated. This paper proposes an ABC based technique, which automatically select the proper alignment technique based on the characteristics of the ontologies that are going to be integrated. This algorithm is a first step towards the automation of the process of integration of ontologies.

This article has five sections, the first one summarizes some aspects about Ontology Mining and Semantic Mining, and the second one presents the bases of the ABC algorithm. In the third, it is presented our ABC approach for the Emergent Selection of the Alignment technique, and finally, some conclusions are presented.

2 Ontology Mining (OM)

OM is known as the set of DM techniques for extracting behavior patterns, knowledge, among other features, in order to build or enrich ontologies. Currently, the growth in the amount of available ontologies on a given knowledge domain, has demanded to OM to explore techniques that can extract additional knowledge of a set of ontologies, particularly the integration patterns of several ontologies, in order to create a broader knowledge domain.

We particularly are interested in two integration mechanisms:

- Ontology linking: it is the process of finding relationships between entities belonging to different ontologies [4]. The results can be used to display maps, transforming a source into another (an ontology into another ontology), creating a set of relationships or rules between ontologies, or generating queries for the two ontologies to extract information.

- Ontology Merge: it is the process where multiple ontologies in the same domain must be joined (fused or merged) in order to standardize knowledge, to have full locally knowledge, among others. Ontology Merge is very important in the distributed systems, because it is costly carry out ontologies queries at different sites. These techniques of ontologies mixture is for the case where the ontologies handle the same knowledge, but with different representations, or having partial representations of such knowledge, such that ontologies may share certain concepts and others not. That requires the presence of an expert that has to be present at the time of the mixture, for decision-making.

These ontological integration mechanisms require a process of ontological comparison, which is usually called alignment [11]. The ontology alignment process usually needs the following elements: two ontologies O1 and O2, a set of parameters p, a set of resources r for alignment, and a function of alignment f, which returns a set of correlation A [6].

The function f defines the process that compares different ontological resources (concepts, relations, among others), in order to find correspondences between two concepts. In each O1 and O2 are analyzed each one of its elements: concepts, concepts properties, concept hierarchy, among others. The set p represents the requirements for alignment, p = {design language like OWL (Web Ontology Language), number of elements, vocabulary, among others}. The set of resources refers to the elements that are used to obtain the set of correlation (r = {similarity measures, algorithms}). The set A symbolizes all the semantic correlation found with the selected algorithm.

Some of the ontology alignment techniques proposed in the literature are [8]:

- Classes Structure: Graph-based techniques that consider as input the ontologies, including database schemas and taxonomies, as labelled graphs. Usually, the similarity comparison between a pair of nodes from the two ontologies is based on the analysis of their positions within the graphs. For that, they use algorithms based on the graph theory. The intuition behind this is that, if two nodes from two ontologies are similar, their neighbors must also be somehow similar [4].

- Distance of names: These techniques consider the names of concepts as sequences of letters in an alphabet. They are typically base on the following intuition: the more similar the strings, the more likely they are to denote the same concepts. Usually, distance functions map a pair of strings to a real number, such that a smaller value indicates a greater similarity between the strings. Some examples of string-based techniques that are extensively used in matching systems are prefix or suffix distances, and n-gram similarity [4].

- Name and properties analysis: it uses the distance of the names, and merges it with another string, which describes the properties of the concepts, to compare them.

This alignment technique has been implemented in the Alignment API [2].

3 ABC Algorithm

The algorithm based on the Colonies of Bees, called Artificial Bee Colony (ABC), has been defined by [6], motivated by the intelligent behavior of bees. It is as simple as Particle Space (PSO) and Differential Evolution algorithm (DE) [6].

The ABC algorithm can be used to solve multidimensional and multimodal optimization problems. The multimodal problems have more than one maximum or minimum. In the model, artificial bee colony consists of three groups of bees: employed, onlookers and scouts. Normally, artificial bees employed compose the first half of the colony and the second half includes the onlookers and scouts. For each food source, there is only one employed bee. In other words, the number of employed bees is equals to the number of food sources around the hive. When an employee has its food source exhausted, it becomes a scout [6].

ABC has parameters as the size of the colony and the maximum number of cycles. As optimization tool, it provides a search procedure based on a population in which possible solutions represent potential sources of artificial food. The aim of the bee colony is to discover the places of food sources with high amount of nectar (good solution). In the ABC system, artificial bees fly around of a multidimensional space of search. The employed and onlookers bees choose food sources based on the experience of them and their nest mates [6].

The scout bees fly and choose food sources randomly without using the experience. If the amount of nectar from a new source is greater or better than the previous one in its memory, the new position is memorized and the previous one is forgotten.

Thus, the ABC system combines local search methods, carried out by bees employed and onlookers, which is performed through the communication of the employed bees giving their expertise to the onlookers, with the methods of global search, which are managed by the scouts when they visit randomly food sources. In this way, this technique tries to balance the exploration and exploitation process [6].

The main steps of the algorithm are [6]:

Send the scouts bees to find food sources
  REPEAT
    Send the employed bees to identified food sources and determine their amounts of nectar.

Calculate the probability value of the sources (quality) with which the onlookers bees will prefer sources.
Send onlookers bees to food sources using a stochastic selection process based on the amount of nectar in each source.
Stop the process of exploitation of sources exhausted by bees.
Send scouts to the search area to discover new food sources randomly.
Save the best food source found so far.
UNTIL (the requirements are met)

The stochastic process in the selection of food sources is given by the probability Pc. In this work, this probability allows to modify the alignment technique stochastically (change of the food source).

4 Emergent Alignment by using our ABC Approach

The problem of the ontology alignment is to be able to decide which of the techniques of semantic alignment must be used. For it, it is used the ABC algorithm in order to let it choose automatically the technique to perform the alignment.

In our problem, the parameters/variables of the ABC algorithm are the following:

• Si: Service that can be utilizes to resolve a request (In our case, the alignment techniques). That is, each alignment technique is a source of nectar.

• Gi(Si): Profit, that is obtained by the use of the service Sj (one alignment technique), defined by the equation 1, which determine the quality of nectar (alignment technique).

• Sai(Si) Satisfaction of the Bee, when the service Sj is performed. It is also relate with the quality of nectar; in our case is the number of aligned nodes of the ontologies.

• CAi(Sj): Cost, it is represented in this work as the execution time of the service Sj to return a result (also affects the quality of nectar).

• Pc: Probability of preserving the food source.

The gain is calculated as follows, in the equation (1):

$$G_i(S_i) = \frac{Sa_i(S_i)}{CA_i(S_i)} Pc \qquad (1)$$

The main part of our algorithm is the procedure followed by the onlooker bees to select the food sources, based on the experiences of the rest of bees on the colony. The scout bees memorize the best food sources that find, and the employed bees determine the food sources to be studied. The onlookers bees change the decision about which food sources exploit using the following decision rules (see figure 1)):

• For each bee i with neighbor j:
If Gi(Si) < Gj(Sj) then
Si = Sj

If the satisfaction (Sai(Si) is the same, then the service that has the shortest time is selected. It is defined as follow:

• For each bee i with neighbor j:

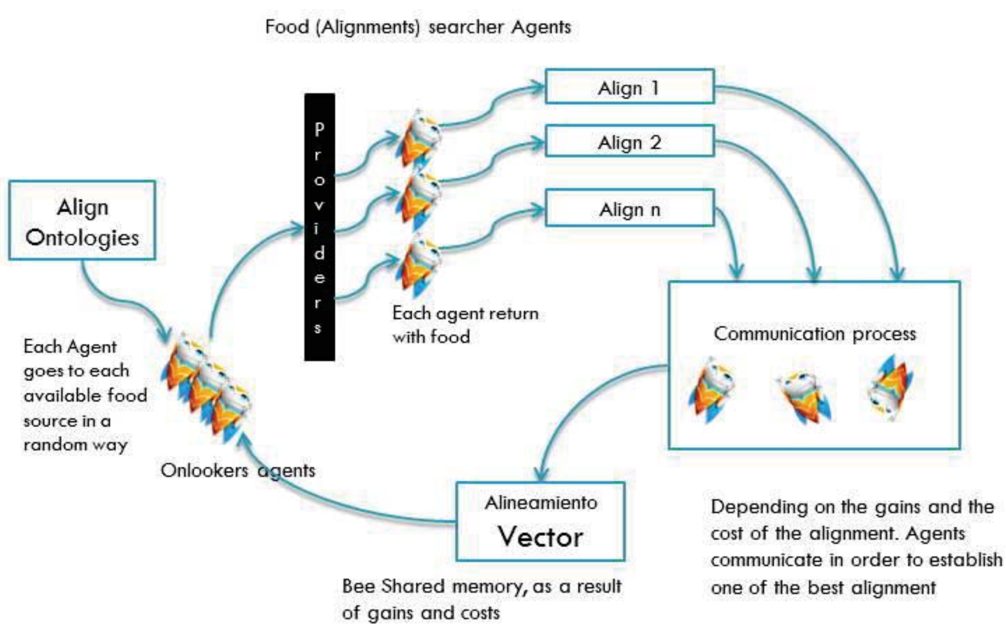


Figure 1. Emergent Alignment Process

If CAi(Si) > CAj(Sj) then
Si = Sj

The algorithm is iterative, and it is done for finite iterations to make several suggestions, it is not necessary that all the bees arrive at a same service (source of nectar); they may suggest various services. At the end of the iterations, we take the service more suggested by the bees (this is the consensus to which arrives the colony of bees).

5 Experiments

We use different pairs of ontologies to test our ABC algorithm: one pair about cars, another of the anatomy of the eye, and finally one pair about computers. These ontologies were taken from [8], where they analyze techniques to merge ontologies. Moreover, the Alignment API [2], is used to test the alignment techniques.

In Figure 2 we see the cars ontologies, which are described as taxonomies of cars, one has the brands of cars of Europe, and the other one has specifically brands of cars from German and Italy.

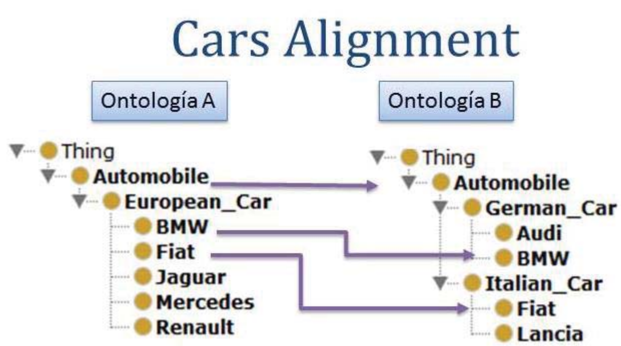


Figure 2. Ontologies to align

In Tables 1, 2, and 3, the parameters of the ABC algorithm are defined for the cars ontologies. These parameters are: the size of the colony, the maximum number of cycles (Maxcycle), the number of times that our algorithm is executed (called runtime, this to check how many times each technique is chosen as the best, among n executions of ABC). The output parameters considered for selection of the best alignment are the execution time, number of times each technique is selected, and the aligned nodes. Bold numbers in the fourth column is for indicating the best result (alignment technique).

The alignment techniques that are used by the ABC algorithm are (they determine the number of food sources, in our case 8) [8]:

a) Class Structure
b) Distance edited name
c) Distance edited subclass name
d) Name and properties
e) Same names
f) Distance SMOA name (A String Metric for Ontology Alignment) [9]
g) Distance chains
h) Sub structures distance

Table 1. Cars Ontologies. Colony size is twice the size of food sources

Maxcycle	Runtime	Execution Time	Times that each Alignment Technique is chosen	# Aligned nodes
250	30	- 6:32 6:44 6:33 6:32 6:33 6:33 7:01	a) 0 **b) 6** c) 3 d) 5 **e) 6** f) 5 g) 5 h) 0	0 3 3 3 3 3 3 3
100	30	- 2:46 2:40 2:49 2:46 2:47 2:49 2:49	a) 0 b) 4 c) 4 d) 3 **e) 6** f) 5 g) 4 h) 4	0 3 3 3 3 3 3 3
25	30	- 0:43 0:50 0:43 0:42 0:46 0:45 0:49	a) 1 b) 5 c) 1 **d) 6** **e) 6** f) 5 g) 4 h) 2	0 3 3 3 3 3 3 3

Table 2. Cars Ontologies. Colony size equal to the size of food sources

Maxcycle	Run time	Execution Time	Times that each Align Technique is chosen	# Aligned nodes
250	30	- 3:26 3:26 3:25 3:23 3:25 3:30 3:29	a) 0 b) 5 c) 5 d) 4 **e) 9** f) 5 g) 0 h) 2	0 3 3 3 3 3 3 3

100	30	-	a) 0	0
		1:26	b) 5	3
		1:24	c) 4	3
		1:22	d) 9	3
		1:23	**e) 9**	3
		1:25	f) 5	3
		1:25	g) 4	3
		1:30	h) 0	3
25	30	-	a) 1	0
		0:24	b) 6	3
		0:28	c) 3	3
		0:27	d) 2	3
		0:23	**e) 7**	3
		0:24	**f) 7**	3
		0:26	g) 2	3
		0:27	h) 2	3

Table 3. Cars Ontologies. Colony size is the half of the size of food sources

Maxcycle	Run time	Execution Time	Times that each Align Technique is chosen	# Aligned nodes
250	30	-	a) 0	0
		1:44	**b) 7**	3
		1:46	c) 5	3
		1:45	d) 4	3
		1:46	e) 4	3
		1:46	f) 5	3
		1:56	g) 3	3
		1:55	h) 2	3
100	30	-	a) 0	0
		0:49	b) 3	3
		0:44	**c) 6**	3
		0:45	d) 5	3
		0:44	e) 5	3
		0:48	f) 4	3
		0:48	g) 4	3
		0:47	h) 4	3
25	30	-	a) 1	0
		0:15	b) 5	3
		0:15	c) 4	3
		0:17	d) 4	3
		0:16	e) 4	3
		0:13	**f) 8**	3
		0:18	g) 2	3
		0:20	h) 2	3

In tables 1, 2, and 3, we show the algorithm performance in case of varying the size of the colony. This variable has important influence on the runtime and search process. It determines how many food sources can test at one time, and it requires a random search to exploit the food sources that are not been used in a given moment (it is the case of table 3). Particularly, the results show that the runtime performance can be significantly improved, without putting the amount of bees so high. Table 3 shows that with the half of the size of food sources, it is sufficient to have a good result (aligned nodes) regarding the case of having a twice as food sources (see table 1).

The number of bees should not be so large because the algorithm balances very well the process of exploration and exploitation. This is due to the ability of exploration of the technique.

Another parameter that is important to analyze is the number of cycles. We see that this parameter has a great influence regarding the techniques that are been more selected. With this parameter, the ability of the collective learning of our ABC approach is exploited in order to obtain the best techniques (b, d, e, and f). We can see that the bad technique is eliminated when we use more cycles (a). That is because the set of selected techniques changes depending on the number of cycles that the ABC algorithm leaves the colony to choose food sources. As part of the learning process the algorithm determines the set of best techniques, but with few cycles it is a random process.

The behavior of our algorithm for the rest of studied ontologies (the eye anatomy and the computers ontologies) is similar, that means the type of ontology have not influence in the quality of the search process (exploration and exploitation).

Table 4. Eye anatomy Ontologies. Case where the colony size is the half of the size of food sources for 250 cycles

Times that each Align Technique is chosen	# Aligned nodes
a) 0	0
b) 0	5
c) 0	5
d) 12	5
e) 0	5
f) 12	5
g) 0	5
h) 6	5

Table 5. Computer Ontologies. Case where the colony size is the half of the size of food sources for 250 cycles

Times that each Align Technique is chosen	# Aligned nodes
a) 0	0
b) 0	10
c) 0	10
d) 15	10
e) 0	10
f) 11	10
g) 0	10
h) 4	10

In the case of the eye anatomy ontologies, our algorithm determines the best alignment techniques are d and f, on the other hand, with computer ontologies the best alignment technique is d. We highlight they are different techniques regarding the best ones in case of the cars ontologies (normally e).

In all of cases of study, normally the analyzed techniques align the same number of concepts (except a, which is eliminated by the learning process of our ABC approach), then, the performance is evaluated in term of the execution time. In this way, the execution time of the different alignment techniques is different for each pair of ontologies to be aligned, and it is detected by our approach when selects them

6 Conclusions

As a major contribution to the field of emerging knowledge engineering, we have proposed an ABC approach to allow an automatic selection of alignment techniques. This process is adapted to the pair of ontologies that we would like to integrate for future uses. Our approach exploits very well the learning and exploration capabilities of the ABC algorithm.

This work is focuses on showing the alignment as an emergent process within an bigger process of ontology integration, when we aim to automate this process.

We have started with the assumption that there are different ways or techniques in the literature for ontology alignment, and there are no a priori criteria to decide which technique to use when we have two ontologies. We propose an equation for calculating the gain as a main criteria to propose a technique, based on the number of aligned nodes.

The gain value is used in the algorithm, and it is penalized by the execution time. Based on this value, our algorithm is capable for finding the best alignment techniques, by using the exploration and exploitation capabilities of the bees' colonies, which share information to arrive to a stable opinion, when all the bees or most of them reach a similar opinion.

Acknowledgments

This work has been developed under the funding FONACIT through the PEII program, research project No. 2994. Dr Aguilar and Dr. Cerrada have been partially supported by the Prometeo Project of the Ministry of Higher Education, Science, Technology and Innovation of the Republic of Ecuador.

7 References

[1] Alma-Delia Cuevas and Adolfo Guzman-Arenas, Automatic Fusion of knowledge stored in Ontologies. *Journal Intelligent Decision Technologies*, 4 (1): 5-19, 2010.

[2] Jérôme David, Jérôme Euzenat, François Scharffe and Cássia Trojahn dos Santos. The Alignment API 4.0, Semantic web journal 2(1):3-10, 2011

[3] Mathieu d'Aquin, Gabriel Kronberger and and María Suárez. Combining Data Mining and Ontology Engineering to Enrich Ontologies and Linked Data. In *Proceedings of the First International Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data*, Greece, pages 19-24, May 2012.

[4] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer-Verlag, Berlin, 2013.

[5] Melanie Hilario, Phong Nguyen, Huyen Do, Adam Woznica and Alexandros Kalousis. Ontology-based meta-mining of knowledge discovery workflows. In *Meta-Learning in Computational Intelligence*, pp. 271-315, Springer, 2011.

[6] Dervis Karaboga and Brune Basturk. On the performance of artificial bee colony (ABC) algorithm, *Applied Soft Computing*, 8: 687–697, 2008.

[7] Ammar Mechouche, Nathalie Abadie and Sébastien Mustière. Alignment-Based Measure of the Distance between Potentially Common Parts of Lightweight Ontologies. In *Proceeding of the 9th International Semantic Web Conference (ISWC'10)*, Shanghai, Novembe 2010.

[8] Salvatore Raunich and Erhard Rahm, Towards a Benchmark for Ontology Merging. *Lecture Notes in Computer Science*, 7567: 124-133, 2012.

[9] Giorgos Stoilos, Giorgos Stamou and Stefanos Kollias. A String Metric for Ontology Alignment. *Lecture Notes in Computer Science*, 3729: 624-637, 2005.

[10] Hongbo Sun, Wenhui Fan, Weiming Shen; Tianyuan Xiao. Ontology fusion in HLA-based collaborative product development. 2010 In *Proceeding of International Conference on Systems Man and Cybernetics*, Turkey, pages 2526-2532, October 2010.

[11] Roberto Zagal. Alineación de ontologías usando el Método Boosting. *Tesis de grado de Maestría en Ciencias de la Computación*. Instituto Politécnico Nacional, Mexico, 2008.

Brain Based Control of Wheelchair

Rahib H.Abiyev, Nurullah Akkaya, Ersin Aytac, Irfan Günsel, Ahmet Çağman
Applied Artificial Intelligence Research Centre, Near East University, Lefkosa, Mersin-10, North Cyprus, Turkey

Abstract - *This paper presents a brain based control of the wheelchair for physically impaired users. The design of the system is focused on receiving electroencephalographic (EEG) signals from the brain, processing and turning the system and then performing control of the wheelchair. The number of experimental measurements of brain activity has been obtained using human control commands of the wheelchair. The obtained data including EEG signals and control commands are used to design brain based control mechanism in training mode. The classification of brain signals has been done using a Support Vector Machine (SVM) and neural networks. The training data is used before using the system under real conditions. Then test data is applied to measure the accuracy of the control. The system designed in this paper is adjusted to control a wheelchair with five commands: move forward, move backward, stop, turn left and turn right in real conditions. The provided approach allows reducing the probability of misclassification and improving control accuracy of the wheelchair.*

Keywords: Brain Based Control, Brain–Computer Interface, Wheelchair Control, SVM, neural networks.

1 Introduction

The use of human brain signals to control devices and software in order to interact with the world is an important problem in bioengineering. The solution to this problem includes two stages. The first stage is the development of the interface between the brain and computer. The second stage is the design of brain based control of devices. The basic aim of the BCI is related to the design of communication channel for disabled people. A BCI system provides communication between computer and mind of pupils. This communication can be based on muscular movements during brain activity or the changes of the rhythms of brain signals. These brain activities can be estimated with electroencephalographic (EEG) signals. Since the brain signals are very weak we need to apply some spatial and spectral filters and amplifiers to the EEG to extract characteristic features of signals. Several EEG signals can be detected, resulting in different types of BCI. These signals are based on change of frequencies, change of amplitudes. For example during voluntary thoughts the frequencies of signals are modified, during movement a synchronization/ desynchronization of brain activity which involves μ rhythm amplitude change. This relevant characteristic makes rhythm based BCI suitable to be used.

Recently some research works have been done to develop many applications of BCI for wheel chairs. BCI is a control interface that translates human intentions into appropriate motion commands for the wheelchairs, robots, devices, etc. BCI allows improving the life quality of disabled patients and letting them interact with their environment. [3] considers the application of BCI and control of wheelchair in an experimental situation. The research considers the driving of a simulated wheelchair in a virtual environment (VE) before using BCI in a real situation. The virtual reality (VR) decreases the number dangerous situations by using train and test applications. [4,5] describe a BCI system which control the wheelchair that moves in only one direction- move forward. In [6] a simulated robot is designed that performs two actions- 'turn left then move forward', or 'turn right then move forward'. [7,8] uses three possible commands turn left, turn right and move forward. In [9] BCI is designed using EEG signal captured by eight electrodes. Wavelet transform was used for feature extraction and the radial basis networks was used to classify the predefined movements. In [10] controller based on the brain-emotional-learning algorithm is used to control the omnidirectional robot. [11] presents the design of an asynchronous BCI based control system for humanoid robot navigation using an EEG. [12] apply BCI to robot control. [13] considers a non-invasive EEG-based Brain Computer Interface (BCI) system to achieve stable control of a low speed unmanned aerial vehicle for indoor target searching. [14-17] consider the design of brain controlled wheelchair. The constriction of viable brain–actuated wheelchair that combines brain computer interface with a commercial wheelchair, via a control layer is considered. Combining the BCI with a shared control architecture [13] allows to dynamically produce intuitive and smooth trajectories.

Another problem in brain based control is the obtaining of high classification accuracy. In brain based wheelchair control, a classification error (a wrong command) can cause dangerous situations, so it is crucial to guarantee a minimum error rate to keep the users safe. In this paper, the design of BCI and efficient brain based control of wheelchair is presented.

2 BCI system architecture

Figure 1 depicts BCI based control of the wheelchair. BCI system consists of an Emotiv headset connected to a computer where classification algorithms are run which is connected to a microcontroller that controls the movement of motors. A BCI based control system is usually composed of four main units: signal acquisition unit, signal preprocessing unit, classification unit and action unit. Figure 2 presents the structure of the system. The brain signals are measured by emotive sensors using 14 different channels. These input signals are sent to the signal processing unit. The signals after preprocessing are entered to the classification system. The output signals of the classification block are motor signals (clusters) that are sent to the wheelchair.

Figure 1. The computer brain–actuated wheelchair

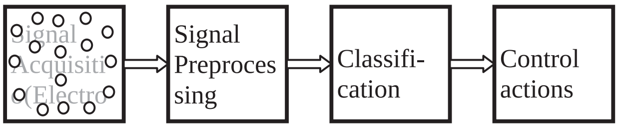


Figure 2. Structure of BCI.

In signal acquisition block the EEG signals are captured using the Emotiv headset. Emotiv EPOC is an EEG Headset which supplies 14 channels EEG data (Figure 3) and 2 gyros for 2 dimensional controls. Its features are adequate for a useful BCI in case of resolution and bandwidth. Our system uses upper face gestures for actuation commands since most Emotiv sensors are located in the frontal cortex they are the most reliable signals to detect.

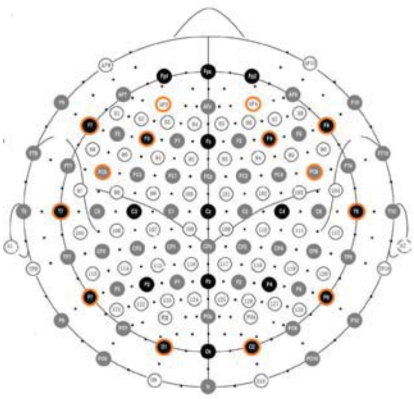


Figure 3. Emotiv's sensor Layout compared to standard 72 sensors layout. The distribution of EEG electrodes. Fourteen channels are marked for data acquisition.

Two different approaches can used for processing of the input sensor signals: Fast Fourier Transform (FFT) and without FFT. In this paper FFT approach is used to process input signal. The use of FFT allow to decrease the size of the input data. Here the input signal received from the headset is divided into windows having 2 sec. time interval with 50% overlap (Figure 4). The use of overlapping windows allows us to increase the accuracy of the classification. Each two seconds window corresponds to 256 samples of data. Each second corresponds to 128 data samples. The obtained signals from the channels, stored as windows, are then sent to normalization block. Each channel is normalized in order to center each channel on zero by calculating the mean value of each channel for the window, then subtracting it from each of the data points in the channel. After normalization, Hamming window is applied to each channel in the window. EEG signals do not generally repeat exactly, over any given time interval, but the math of the Fourier transform assumes that the signal is periodic over the time interval. This mismatch leads to errors in the transform called spectral leakage. Hamming window is used to mitigate this problem. Then fast Fourier transform (FFT) is applied to each channel in the window to find out the frequency components of the signal. Each frequency component is used as a feature, which results in 64x14 features. In order to increase the performance of the classification, the features are ranked by evaluating the worth of a frequency by measuring the information gain with respect to the class.

$$\text{InfoGain(Class,Frequency)}=H(\text{Class}) - H(\text{Class} \mid \text{Frequency})$$

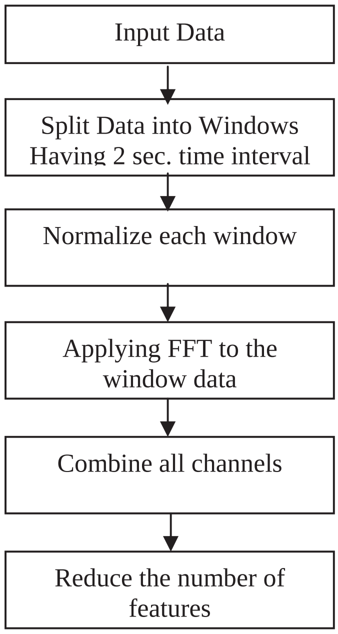


Fig.4. Signal Preprocessing unit.

After frequency representation, the combination of all channel signals is performed. The filtering operation is applied in order to select important features of the brain signals. These features are used for classification purpose. The whole signal preprocessing stages are shown in Figure 4. In the second approach the acquired brain signal after windowing, normalization and combining operations are used for classification purpose.

These signals are input for classification block. The signals are processed and classified. Output of classification system is used to activate the wheelchair. Even though during training system reports 100% success rate in real world conditions it does misclassify, a state machine is used to further increase safety and reduce misclassification. As an example the system won't transition from forward motion to backward motion without stopping in neutral. Output of the state machine drives the microcontoller which controls the motors on the wheelchair. The number of classes is equal to the number of control actions.

3 Classification

3.1 Support Vector Machines

The features extracted from the EEG signals are used for classification and determining control action. For this purpose in the paper classification techniques, such as SVM and neural networks are applied. The SVM method was invented by Vapnik, and the current standard improvement was proposed by Cortes and Vapnik [18].

Support vector machine tries to find out a hyperplane that has best separation which can be achieved by largest distance to the nearest training data point of any class. Let assume a binary classification have a data points (x_i,y_i), where $x_i \in R^p$ data points, $y_i \in \{-1,1\}$ classes. Each (x_i) is a vector. It needs to find the maximum-margin hyperplane that divides the points into two class. It can be described as:

$w.x + b = 1$ and $w.x + b = -1$.

where w is the normal vector to the hyperplane. It needs to minimize $\|w\|$ to prevent data points from falling into the margin, it needs to add the following constraint: for each i either $w.\, x_i – b \geq 1$ for x_i of the first class or $w.\, x_i – b \leq -1$ for x_i of the second class. As a result, it can be written as $y_i\, (w.\, x_i – b) \geq 1$, $1 \leq i \leq n$. The samples along the hyperplanes are called Support Vectors (SVs) and separating hyperplane with largest margin can be defined by M=$\frac{2}{\|w\|}$ that specifies support vectors means training data points closets to it. Taking into account the mentioned we can obtain the optimization quadratic problem:

$$\begin{aligned} &\text{Minimize } \|w\| \\ &\text{Subject to: } y_i\, (w.\, x_i – b) \geq 1, \\ &\qquad \text{for any } i=1,\ldots,n. \end{aligned} \tag{1}$$

The main goal in SVM is the maximization of the margine of separation and minimization of training error. The above problem can be transformed into Lagarange formulation.

$$\begin{aligned} &\max imize \quad L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \\ &subject\ to \quad \sum_{i=1}^{n} y_i \alpha_i = 0 \\ &\qquad \alpha_i \geq 0, \quad i = 1, \ldots, n. \end{aligned} \tag{2}$$

where $K(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$ is a kernel function that satisfies Mercer theorem. Based on Karush-KuhnTucker(KKT) complementarity conditions the optimal solutions α∗ ,w∗ ,b∗ must satisfy the following condition.

$$\alpha_i^* [y(w^* \varphi(x_i) + b^*) - 1] = 0, \quad i = 1, \ldots, n.$$

where the α_i^* are the solutions of the dual problem. The resulting SVM for function estimation becomes

$$f(x) = \text{sgn}(\sum_{i=1}^{m} \alpha_i^* y_i K(x_i, x_i^*) + b) \tag{3}$$

where m is the number of support vectors. SVM technique is a powerful widely used technique for solving supervised classification problems due to its generalization ability. In essence, SVM classifiers maximize the margin between training data and the decision boundary (optimal separating hyperplane), which can be formulated as a quadratic optimization problem in a feature space.

3.2 Neural Network

Feed-forward neural network is applied for classification of brain signals. The used NN include input, hidden, and output layers. The sigmoid activation function is used in the neurons of hidden and output layers. Once the neurons for the hidden layer are computed, their activations are then fed to the next layer until all the activations finally reach the output layer. Each output layer neuron is associated with a specific classification category. In a multilayer feed-forward network (Figure 5), each neuron of previous layers is connected the neurons of next layers by using weight coefficients. In computing the value of each neuron in the hidden and output layers one must first take the sum of the weighted sums and the bias and then apply activation function f(sum) (the sigmoid function) to calculate the neuron's activation [19].

The extracted features of the anthemia diseases are inputs of neural networks. In this structure, $x_1, x_2, \ldots, x_m$ are input features that characterize the anthemia diseases. The j-th output of two layer neural networks is determined by the formula

$$y_j = f_k(\sum_{j=1}^{h} v_{jk} \cdot f_j(\sum_{i=1}^{m} w_{ij} x_i))$$
$$where \quad f(\Sigma) = \frac{1}{1+e^{-\Sigma}} \tag{4}$$

where w_{ij} are weights between the input and hidden layers of network, v_{jk} are weights between the hidden and output layers, f is the sigmoid activation function that is used in neurons, x_i is input signal. Here k=1,..,n, j=1,..,h, i=1,..,m, m, h and n are the numbers of neurons in input, hidden and output layers, correspondingly.

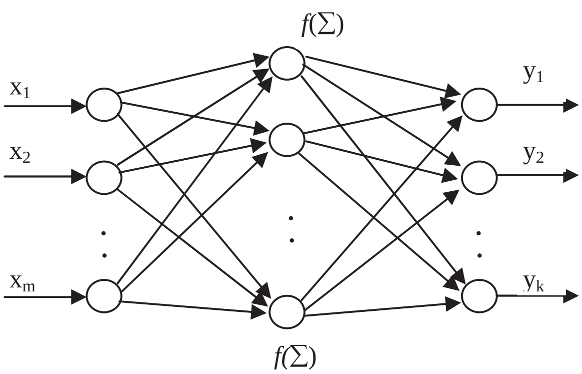


Figure 5. Multilayer feed-forward network

After activation of neural network, the training of the parameters of NN starts. NN is trained using anthemia data set taken from UCI library. During learning the 10-fold cross validation is used for evaluation of classification accuracy. There should be set of experiments in order to achieve required accuracy in the NN output. The simulation is performed using different number of neurons in hidden layer. The number of output neurons was 8 which was equal to the number of classes. The backpropagation algorithm is applied for training of NN. Neural network training consists of minimizing the usual least-squares cost function:

$$E = \frac{1}{2}\sum_{p=1}^{O} (y^d - y)^2 \tag{5}$$

where O is the number of training samples for each class, y^d and y is the desired and current outputs of the p input vector.

The training of the NN parameters has been carried out in order to generate a proper NNs model. The parameters w_{ij}, v_{jk} $(i=1,...,m$, $j=1,...,h,$ $k=1,\ldots,n)$ of NNs are adjusted using the following formulas.

$$w_{ij}(t+1) = w_{ij}(t) + \gamma \frac{\partial E}{\partial w_{ij}}; \quad v_{jk}(t+1) = v_{jk}(t) + \gamma \frac{\partial E}{\partial v_{jk}} \tag{6}$$

where, γ is the learning rate, $i=1,...m;$ $j=1,...h;$ $k=1,\ldots,n;$ m, h, n are the number of inputs, hidden and output neurons of the network.

The whole process includes the following steps:

1) In the first step, the weights of neurons are initialized in the interval of [0-1].
2) Input data are fed to NN input (forward propagation).
3) Outputs of neurons of hidden layer are computed (Feedforward process)
4) The outputs of the hidden layer are fed to the inputs of output layer of NN and the outputs of NN are computed.
5) The error between current outputs and desired outputs (target) is computed.
6) Error is propagated back to the previous layer in order to update the weight coefficients of the neurons of the network. The back propagation of error signal is continued until the update of all weight coefficients in the layers is performed.
7) Repeating the steps 2 to 6 until the error becomes an acceptable small value.

4 Experiments and Results

The BCI system is simulated and used in real life application. The EEG signals are measured with Signal acquisition unit- the Emotiv EPOC headset. In the experiments, we have used 14 channels for measuring EEG signals. The measured EEG signals have different rhythms within the frequency band. The experiments show that measuring brain signals is difficult so we have tested our system using brain muscle signals. The signals obtained from 5 sample channels are shown in Figure 6. Figure 6(a) depicts a neural pose, patient relax not doing anything. Figure 6(b) depicts a positive gesture. As shown in figures, the EEG signals with positive gesture pose are changing more frequently than neutral pose. After preprocessing the important features of these signals are extracted and used for classification purpose.

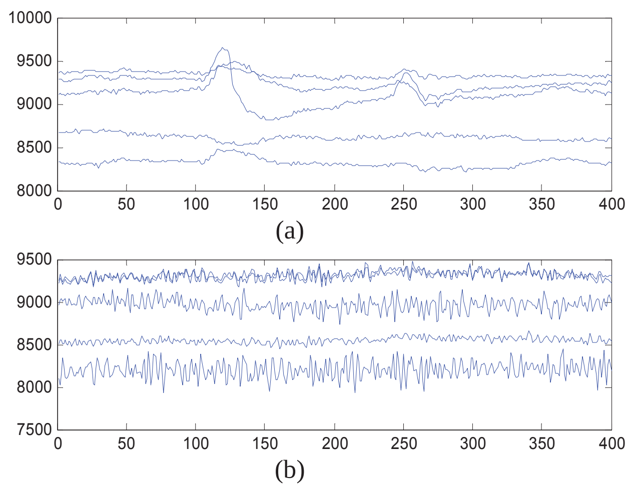


Figure 6. EEG signals for five channels: a) neutral pose, b) positive gesture pose

Five clusters are used in the experiment: Move Backward, Move Forward, Stop, Turn Left, and Turn Right. For each cluster, the system recorded 10 seconds of data. The classification of the signals is performed using SVM with the polynomial kernel and neural networks. 10 fold cross validation is used for separation the data into training and testing set. For comparison purpose we test the system using different classification techniques. In the result of classification the following results are obtained (Table 1).

Table 1. Classification results

Method	Correctly Classified Instances	Incorrectly Classified Instances	Mean absolute error	Root mean squared error
SVM	50 / 100%	0	0.24	0.3162
MLP (NN)	50 / 100%	0	0.0309	0.0884
Bayesian	47 / 94%	3 / 6%	0.024	0.1549
Random tree	37 / 74%	13 / 26%	0.104	0.3225

As shown using SVM and Neural networks the classification rate is achieved as 100%. These clusters activate the corresponding control signal which is then used to actuate the motors of the wheel chair.

5 Conclusion

The paper presents design of brain based control system for the wheelchair. The emotional and muscular states of the user are evaluated for classification and control purpose. The design of BCI has been done to drive brain controlled wheelchair using five mental activities of the user: Move Backward, Move Forward, Stop, Turn Left and Turn Right. For classification of EEG signals, SVM and neural networks with 10 fold cross validation data set are used. The obtained 100% classification results prove that the used techniques are a potential candidate for the classification of the EEG signals in the design of brain based control system. In the future, we are going to improve the number of commands for control wheelchair and decrease detection time of the EEG signal used for measuring brain activities and design efficient brain controlled wheelchair.

6 References

[1] Roman Gonzalez. Communication Technologies Based on Brain Activity. 2010 World Congress in Computer Science, Computer Engineering and Applied Computing-WORLDCOM 2010. BIOCOMP 2010: 864-869

[2] Francisco Velasco-Álvarez and Ricardo Ron-Angevin. Asynchronous Brain-Computer Interface to Navigate in Virtual Environments Using One Motor Imagery. J. Cabestany et al. (Eds.): IWANN 2009, Part I, LNCS 5517, pp. 698–705, 2009. © Springer-Verlag Berlin Heidelberg 2009

[3] Galán, F., Nuttin, M., Vanhooydonck, D., Lew, E., Ferrez, P.W., Philips, J., de Millán,J.R.: Continuous brain-actuated control of an intelligent wheelchair by human EEG. In: Proceedings of the 4th International Brain-Computer Interface Workshop and Training Course, TU Graz/Büroservice, Graz, pp. 315–320 (2008)

[4] Leeb, R., Settgast, V., Fellner, D., Pfurtscheller, G.: Self-paced exploration of the Austrian National Library through thought. International Journal of Bioelectromagnetism 9(4), 237–244 (2007)

[5] Leeb, R., Friedman, D., Müller-Putz, G.R., Scherer, R., Slater, M., Pfurtscheller, G.: Self-Paced (Asynchronous) BCI Control of a Wheelchair in Virtual Environments: A Case Study with a Tetraplegic. Computational Intelligence and Neuroscience. Article ID 79642 (2007)

[6] Tsui, C.S.L., Gan, J.Q.: Asynchronous BCI Control of a Robot Simulator with Supervised Online Training. In: Yin, H., Tino, P., Corchado, E., Byrne, W., Yao, X. (eds.) IDEAL 2007. LNCS, vol. 4881, pp. 125–134. Springer, Heidelberg (2007)

[7] Scherer, R., Lee, F., Schlögl, A., Leeb, R., Bischof, H., Pfurtscheller, G.: Towards elfpaced Brain-Computer Communication: Navigation through virtual worlds. IEEE Transactions on Biomedical Engineering 55(2), 675–682 (2008)

[8] Anas Fattouh, Odile Horn, Guy Bourhis. Emotional BCI Control of a Smart Wheelchair. IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 3, No 1, May 2013

[9] Vijay Khare, Jayashree Santhosh, Sneh Anand, Manvir Bhatia. Brain Computer Interface Based Real Time Control of Wheelchair Using Electroencephalogram. International Journal of Soft Computing and Engineering (IJSCE) SSN: 2231-2307, Volume-1, Issue-5, November 2011.

[10] Maziar A. Sharbafi, Caro Lucas, and Roozbeh Daneshvar. Motion Control of Omni-Directional Three-Wheel Robots by Brain-Emotional-Learning-Based Intelligent Controller. IEEE Trans. on Systems, Man, and Cybernetics—Part C: Applications and Reviews, Vol. 40, No. 6, 2010

[11] Yongwook Chae, Jaeseung Jeong, *Member*, *IEEE*, and Sungho Jo. Toward Brain-Actuated Humanoid Robots: Asynchronous Direct Control Using an EEG-Based BCI. IEEE Trans. On Robotics, Vol. 28, No. 5, 2012

[12] Simon Grude, Matthew Freeland, Chenguang Yang and Hongbin Ma. Controlling mobile Spykee robot using Emotiv

Neuro Headset. In Proceed. İn Chinese Control Conference. 26-28,July, 2013, Xian, China

[13] Tianwei Shi, Hong Wang, Chi Zhang. Brain Computer Interface system based on indoor semi-autonomous navigation and motor imagery for Unmanned Aerial Vehicle control. Expert Systems with Applications 42 (2015) 4196–4206

[14] Tom Carlson, Robert Leeb, Ricardo Chavarriaga and Jos´e del R. Mill´an. The Birth of the Brain–Controlled Wheelchair. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems October 7-12, 2012. Vilamoura, Algarve, Portugal, pp. 5444-5445

[15] 16. B. Rebsamen, C. Guan, H. Zhang, C. Wang, C. Teo, M. H. Ang, Jr., and E. Burdet, “A brain controlled wheelchair to navigate in familiar environments,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 6, pp. 590–598, Dec. 2010

[16] I. Iturrate, J. M. Antelis, A. Kubler, and J. Minguez, “A noninvasive brain-actuated wheelchair based on a p300 neurophysiological protocol and automated navigation,” IEEE Trans. Robot., vol. 25, no. 3, pp. 614–627, Jun. 2009.

[17] G. Vanacker, J. del R. Mill´an, E. Lew, P. W. Ferrez, F. G. Moles, J. Philips, H. V. Brussel, and M. Nuttin, “Context-based filtering for brain-actuated wheelchair driving,” Comput. Intell. Neurosci., vol. 2007, pp. 1–12, May 2007

[18] Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." Machine learning 20, no. 3 (1995): 273-297.

[19] Koch, Christof, and Idan Segev, eds. Methods in neuronal modeling: from ions to networks. MIT press, 1998.

[20] Rahib H.Abiyev, Nurullah Akkaya, Ersin Aytac, Irfan Günsel, Ahmet Çağman. Development of Brain Computer Interface for Wheelchair. The International Biomedical Engineering Congress 2015 (IBMEC-2015), 12-14 March 2015, Girne, North Cyprus

eJADE-S: Encrypted JADE-S for Securing Multi-Agent Applications

Basit Ali[1], Umar Manzoor[2], and Bassam Zafar[2]
[1]Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Pakistan
[2]Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
umarmanzoor@gmail.com, bzafar@kau.edu.sa

Abstract – *In the last two decades, Multi Agent System (MAS) has been used in diverse areas because of its tremendous capabilities; however, security and reliability issues are two major hurdles in adaptation of this paradigm in real distributed applications. Java Agent Development Framework (JADE) is one of the most popular frameworks used for the implementation of MAS, however, JADE has many security issues. To overcome these security issues Jade-S is proposed, however, it still has many security weaknesses. In this paper, we have proposed Encrypted Jade-S (eJade-S) for securing Multi-agent based applications. Cryptography new algorithm using two s-box for encryption / decryption is integrated in JADE-S to overcome the security issues. We have tested the proposed solution on large number of multi-agent based applications; experimental results show the effectiveness of the proposed solution in securing multi-agent based applications.*

Keywords: Securing MAS, JADE, JADE-S, Multi-Agent Systems, Encrypted JADE-S

1 Introduction

Agents are software programs that capable of performing autonomous action on the behalf of the user [6, 7]. The central concept of autonomy in agent means it does not have to receive instruction from the user and can decide the action itself according to the situation. When these agents work together to perform some common task(s) such a system is known as Multi Agents System. Agents within Multi-agent based application make can make virtual society for negotiation / obtaining their common goals [13].

In the last two decades, Multi Agent System (MAS) has been used in diverse areas such as (health sciences, bioinformatics, software modeling, distributed systems, biomedical engineering, parallel computing, autonomic computing [10, 11, 14, 15] etc) because of its tremendous capabilities; however, security and reliability issues are two major hurdles in adaptation of this paradigm in real distributed applications. In the last two decades, many frameworks for development / implementation of multi-agent system have been proposed; Java Agent Development Framework (JADE) [8] is one of the better and most often used frameworks for the implementation of MAS. Is JADE a secure framework for developing multi-agent based application? The answer is not yet however several steps have been taken for making JADE secure. The architecture layer of JADE is unsecure which implies that any application developed with JADE is also unsecure and can easily get infected by different attacks. JADE-S [1] has been design to overcome the security weakness of JADE; JADE-S is the security add-on incorporated in the JADE version 3.2 and provides authorization, authentication, message encryption, message signature and security policy features for the agents.

Although JADE-S aims to provide the security goal, however, it still has many security weaknesses [2, 3]. JADE-S saves the information related to security policy / authentication in plain text files, which can easily be modified / updated. In this paper, we have proposed Encrypted JADE-S (eJADE-S) for securing Multi-agent based applications. Cryptography new algorithm using two s-box for encryption / decryption is integrated in JADE-S to overcome the security issues.

The paper is organized as follows. The first section discusses the background related to multi-agent system, possible attack on agent based applications and possible solutions. This section is followed by the discussion of the Encrypted JADE-S (eJADE-S) which provides mechanism for securing multi-agent based applications. In Section 4 performance analyses of proposed solution on different test cases is presented. At the end conclusion is drawn and we outline some questions for future research.

2 Background

Because of tremendous capabilities of Agent technology, it has been used in diverse areas especially distributed systems. Many efficient solutions have been proposed using agent paradigm, as a result, agent development has moved from researcher communal to the practical fields. Java Agent Development Framework (JADE) is one of the most often used framework for the development / implementation of multi-agent based applications. Foundation for Intelligent

Physical Agents (FIPA) [9] is responsible for standard stipulations for multi-agent expertise and JADE supports (FIPA) stipulations.

In 2005, JADE proposed a plug-in named as JADE-S which provides the security features for the agent development framework (i.e. JADE). By extending the security model of Java, JADE-S used customizable sandbox and provided security modules which are essential for the development of secure multi-agent system.

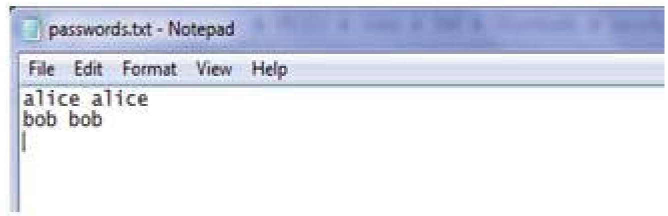
Mobile Agents [4, 5] are usually distributed over different hosts, to provide security in such a distributed and open environment JADE-S presented the idea of multi-user system where all the activities are performed by the authenticated agents. The administrator of the system assign privilege to these agents. Moreover, each agent owned a public and private key through which it can encrypt and sign messages.

2.1 Authentication – JADE-S

The authentication process is the base of whole system and all other services that are used by this JADE-S. The authentication of the JADE-S module is supported by the security service which is composed of two elements of JAAS API (Java Authentication and Authorization Service) namely [6]:

- A Callback Handler that force the user to provide username and password.
- A Callback Handler that authorize the user.

Every user that start / create the container or agent must authenticate itself by providing username / password to the system, if the authentication is successful, the user becomes the owner of the container. This implicates that the user who starts the platform, by executing the main container also own the AMS and Directory Facilitator (DF). With the help of Callback Handler the user send their username and password to the local containers which passes the same to the main container. After receiving the information, Main container verifies the same with the login module which matches the user name / password to the corresponding user name / password stored in the password text file as shown in figure 1. If the login is successful, the authenticated agent and its container may start execution in the form of platform otherwise the system will generate error message and exits.



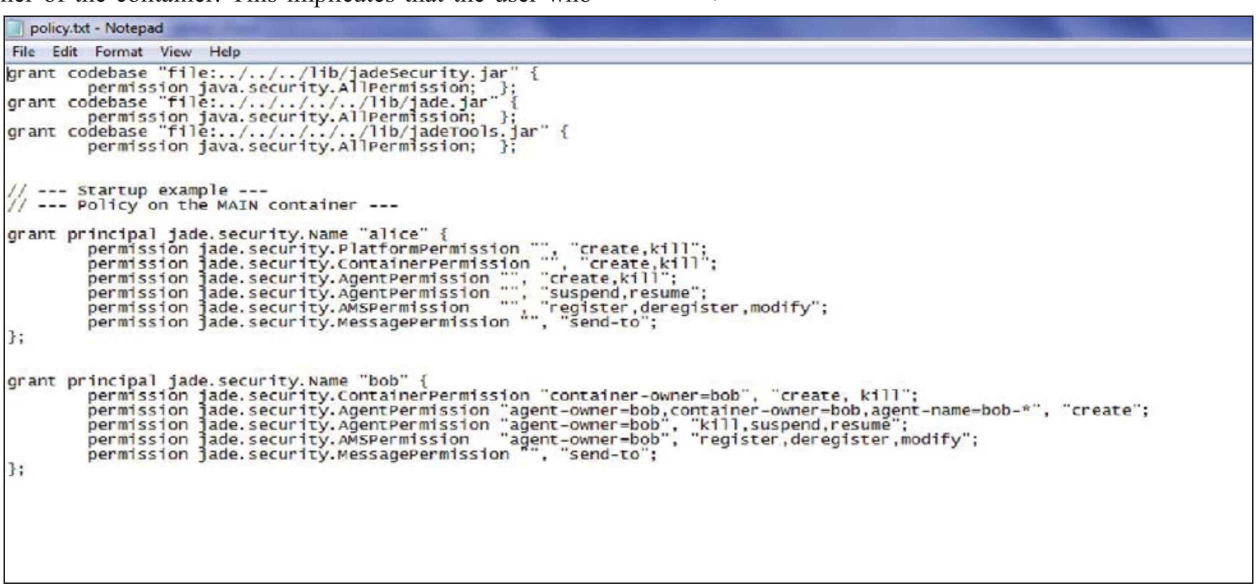
```
passwords.txt - Notepad
File Edit Format View Help
alice alice
bob bob
```

Figure 1. Sample Password.txt file

2.2 Authorization – JADE-S

The Authentication module of JADE-S assigns permissions to the agents (i.e. all possible actions and operations which the agent can perform are listed). With the help of these rules the action are permitted and denied by JADE-S. These set of rules or access control list is saved in the policy file usually named as policy.txt. This file follows the default JAAS syntax, any action that is not defined in the policy is denied forcefully by the system. The following rights can be issued to the agents:

Platform Permission: Right to create / kill Main Container of the system, Create container / agent, Kill agent / container etc. These permissions are normally assigned to the administrator of the system.



```
policy.txt - Notepad
File Edit Format View Help
grant codebase "file:../../../lib/jadeSecurity.jar" {
        permission java.security.AllPermission;  };
grant codebase "file:../../../../../lib/jade.jar" {
        permission java.security.AllPermission;  };
grant codebase "file:../../../../../lib/jadeTools.jar" {
        permission java.security.AllPermission;  };


// --- Startup example ---
// --- Policy on the MAIN container ---

grant principal jade.security.Name "alice" {
        permission jade.security.PlatformPermission "", "create,kill";
        permission jade.security.ContainerPermission "", "create,kill";
        permission jade.security.AgentPermission "", "create,kill";
        permission jade.security.AgentPermission "", "suspend,resume";
        permission jade.security.AMSPermission   "", "register,deregister,modify";
        permission jade.security.MessagePermission "", "send-to";
};

grant principal jade.security.Name "bob" {
        permission jade.security.ContainerPermission "container-owner=bob", "create, kill";
        permission jade.security.AgentPermission "agent-owner=bob,container-owner=bob,agent-name=bob-*", "create";
        permission jade.security.AgentPermission "agent-owner=bob", "kill,suspend,resume";
        permission jade.security.AMSPermission   "agent-owner=bob", "register,deregister,modify";
        permission jade.security.MessagePermission "", "send-to";
};
```

Figure 2. Sample Policy.txt file

- Container Permission: Right to create / kill a container.
- Agent Permission: Right to create / kill an agent in the local container.
- AMS Permission: Right to register and de-register agent in the AMS.
- Message Permission: Allows agent to send messages to other agents.

2.3 Encryption and Signature – JADE-S

Agents communicate with each other through messages on same or different platform, signature mechanism ensure the data reliability and non-manipulation whereas encryption assurance confidentiality. The signature and encryption mechanism of JADE-S is assured by assigning asymmetric public and private key to each agent. The signing and verifying operations are assured by the signature whereas encrypt and decrypt operation are handled by Encryption. These operations always operates on the platform in order to protect the sensitive data / information.

Agents call the appropriate security function in order to achieve the security and confidentiality of messages. Before sending message the Security Helper class function is applied on ACL Message. The sending agent has no concern with the decryption and verification of signature. These services are performed on the receiving end. In case of any failure during decryption or signature verification process, ACL Failure Messages is send to the sender.

2.4 IMTP over SSL – JADE-S

This is last and most important security feature provided by the JADE-S which includes privacy, data reliability and authenticated connection between the agents' Container Transport Protocol (IMTP) over SSL (i.e. RMI over SSL). The container-to-container mechanism of JADE-S is slightly different from the agent-to-agent where each container is awarded with a certificate. The certificate of all containers in the platform is saved in the trusted block of container. The security algorithms like mutual authentications, encryption and signature are performed by the TSL/SSL Protocol. Every container must share his owns certificate to the other communicating container and verifies the one presented to it using the trusted store. After Successfully verification, the TLS/SSL protocol continues with encryption and signature of all information exchanged between the containers.

3 Encrypted JADE-S

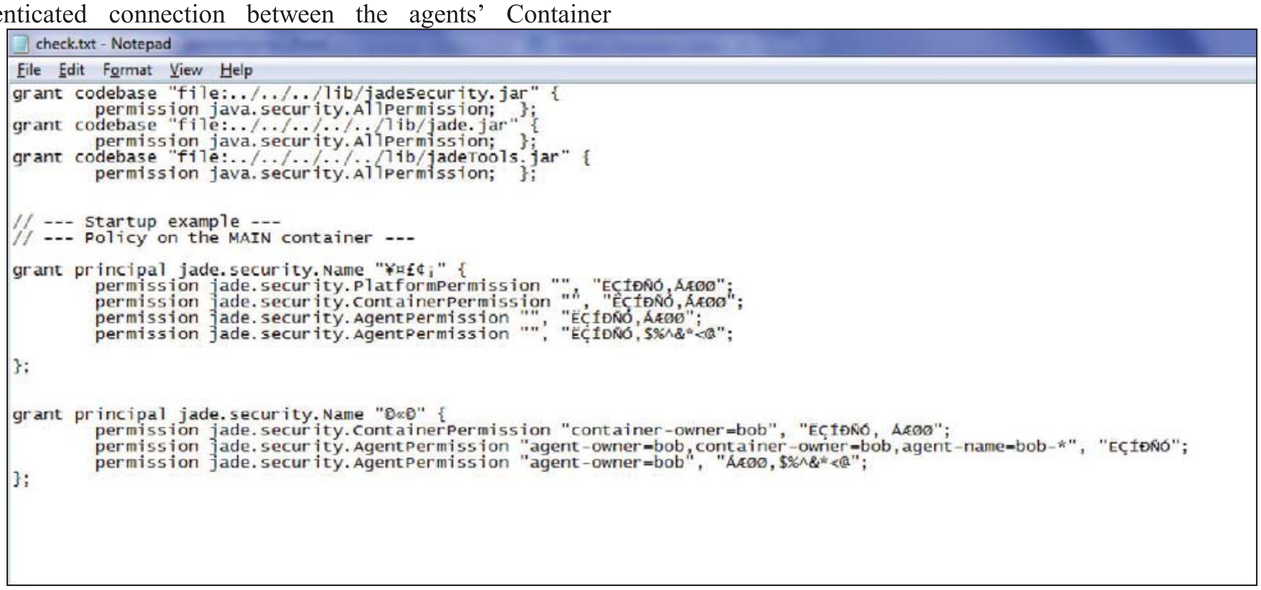
Jade-S creates some configuration files during installation in the root directory of the system. These files contain necessary credentials of user i.e. login username and password, privacy and policies of the system, agent roles etc are defined in the same configuration files.

```
e-jade.txt - Notepad
File Edit Format View Help
¥¤£¢¡ ¥¤£¢¡

Ð«Ð Ð«Ð
```

Figure 3. Encrypted JADE-S Password File

These files are simple text files as shown in figure 1 and 2 respectively. These configuration files are usually placed at the project root, and any mobile agent running on the host (where these files are stored) can access / change these configurations. For example, adding more privileges to itself by making changes to the configuration files.



```
check.txt - Notepad
File Edit Format View Help
grant codebase "file:../../../lib/jadeSecurity.jar" {
        permission java.security.AllPermission;  };
grant codebase "file:../../../../../lib/jade.jar" {
        permission java.security.AllPermission;  };
grant codebase "file:../../../../../lib/jadeTools.jar" {
        permission java.security.AllPermission;  };


// --- Startup example ---
// --- Policy on the MAIN container ---

grant principal jade.security.Name "¥¤£¢¡" {
        permission jade.security.PlatformPermission "", "ËÇÍÐÑÓ,ÁÆØØ";
        permission jade.security.ContainerPermission "", "ËÇÍÐÑÓ,ÁÆØØ";
        permission jade.security.AgentPermission "", "ËÇÍÐÑÓ,ÁÆØØ";
        permission jade.security.AgentPermission "", "ËÇÍÐÑÓ,$%^&*<@";

};


grant principal jade.security.Name "Ð«Ð" {
        permission jade.security.ContainerPermission "container-owner=bob", "ËÇÍÐÑÓ, ÁÆØØ";
        permission jade.security.AgentPermission "agent-owner=bob,container-owner=bob,agent-name=bob-*", "ËÇÍÐÑÓ";
        permission jade.security.AgentPermission "agent-owner=bob", "ÁÆØØ,$%^&*<@";
};
```

Figure 4. Encrypted JADE-S Policy File

In this paper, we have proposed Encrypted JADE-S (eJADE-S) by incorporating a new cryptographic algorithm in JADE-S to secure the configuration files (i.e. secure the login credentials, agent roles, and policy etc) generated by JADE-S. Our proposed techniques uses multi-substitution box in data encryption which makes it nearly impossible to change / modify the configuration settings. In eJade-S the password file is renamed to e-jade.txt, if any agent locates / open this file, it can't understand the contents because they are encrypted using our proposed algorithm discussed later in the section.

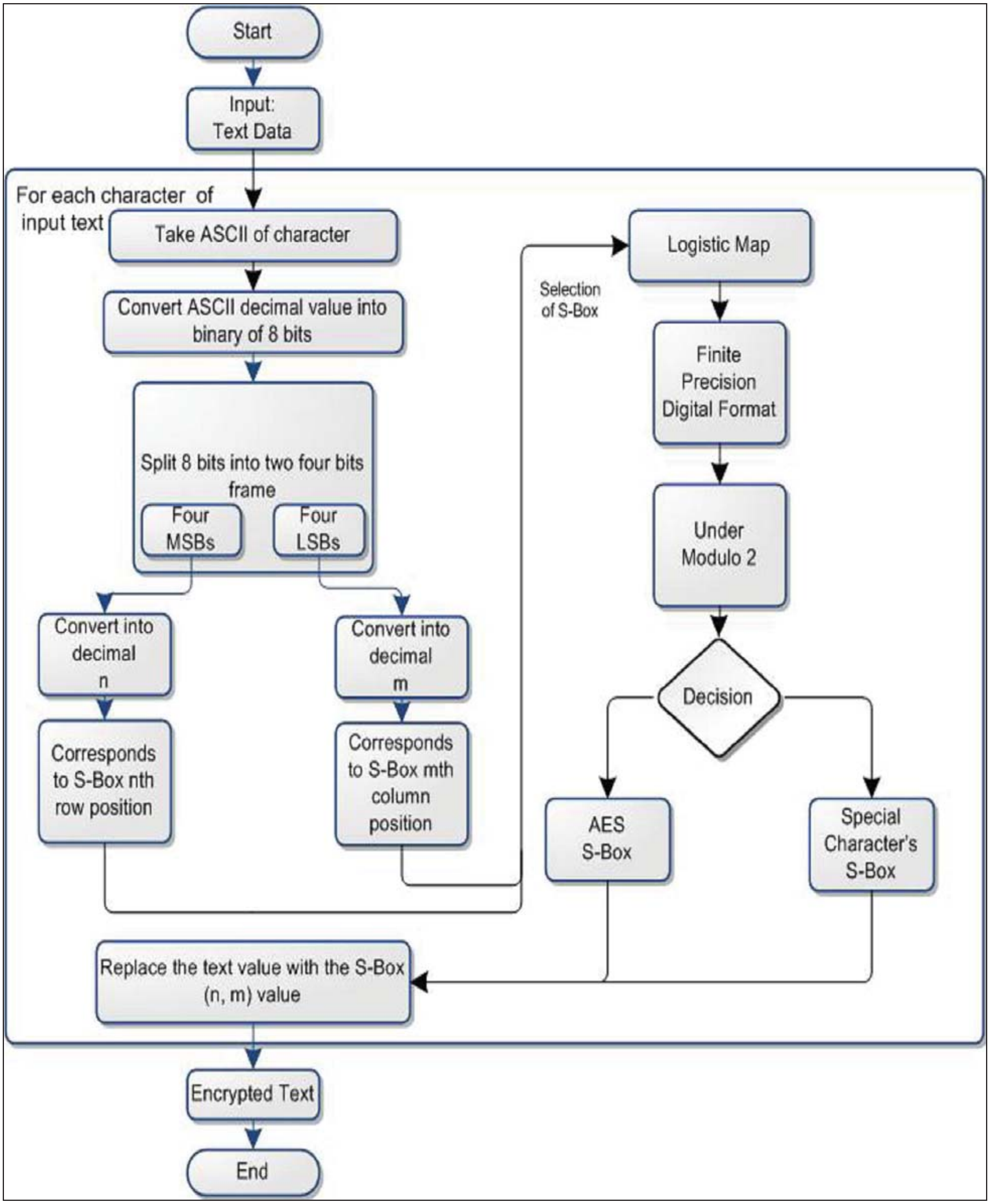


Figure 5. Proposed Substitution algorithm

After the encryption is applied on the login information, the username and password of the each agent is encrypted as shown in figure 3. Similar to password file, policy file which includes the rules / rights of the agents is also encrypted using the same algorithm; therefore, it becomes really hard for the unauthorized agent to understand which right is given to what particular agent. Once these configuration files become encrypted, entire application becomes more secure than before.

3.1 Encryption Algorithm

The encryption algorithm uses multiple S-box instead of single S-Box in substitution method.

Input: Simple text that need to encrypt.

Output: Encrypted Text.

```
While characterarray[]! = null do
   Ascii number = characterarray[i]Asciivalue
   Binarynumber = convert Ascii number to binary
   Binaryarray[] = Binarynumbersplitby4digits:
   Rowvalue = Binaryarray[0]
   Columnvalue = Binaryarray[1]
   Sum = Rowvalue + Columnvalue
   Mod = Sum/2
   If Mod == 0 then
      Encrpytedvalue = S0[Rowvalue][ Columnvalue]
      characterarray[i] = Encrpytedvalue
   else
      Encrpytedvalue = S1[Rowvalue][ Columnvalue]
      characterarray[i] = Encrpytedvalue
   end if
End While
```

3.2 Decryption Algorithm

Decryption Algorithm follows the same step in reverse order to convert the encrypted data into original text.

Input: Encrypted Text Data.

Output: Original Simple text data.

```
While decryptedlettersarray[]! = nulldo
   Find the location of decrypted character in both S-boxes. Once the letter found save the row and column index of that character.
   Row = decryptedlettersarrayrownumber:
   Col = decryptedlettersarraycolnumber
   Rowbinarynumber = Convert Row number to binary number
   Colbinarynumber = Convert Col number to binary number
   Binary = Add both Rowbinarynumber and Colbinarynumber as a string to become 8 or 16 digit numbers.
   ASCII = Convert Binary to decimal number.
   Orignaltext = Get the character value against the ASCII number.
End While
```

4 Experimental Analysis

eJade-S is design to secure the Multi Agent System, the end user can upload the multi-agent based application as input and the proposed framework yields secure MAS application as output. The GUI of application is shown in Figure 6. The end users select the MAS project that needs to be secured by clicking on the Upload Project. All the agents in the uploaded project are listed in the list box which is on the left side of the window. The user must select the Agent from the agent list and assign the roles / right by clicking on the check box which are on the right side of the window. The users must repeat the same step to assign the roles to each and every agent individually. Once the roles are assigned, the user needs to click the secure project Button which yields secure multi-agent application as output.

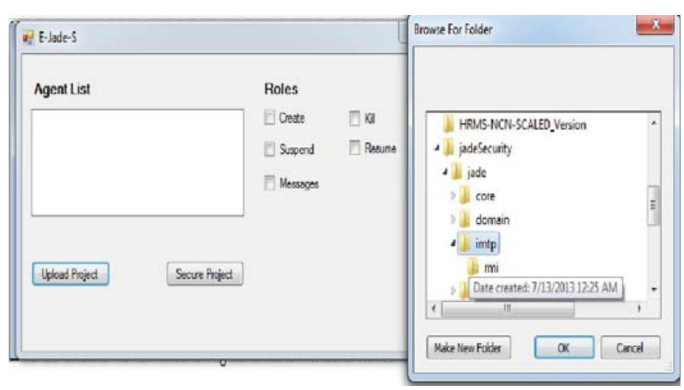


Figure 6. GUI of Encrypted JADE-S

We have evaluated the proposed solution on large number of test cases; in this section we have discussed one test case which is multi-agent based application proposed by Manzoor et al [12] for activity or resources monitoring over the networks. An agent based system for activity monitoring on network (ABSAMN) is fully autonomous and manage the resources with the help of mobile agents. The ABSAMN uses XML configuration files for managing / monitoring the network, these configuration files are unsecure as any agent can amend or modify them. Furthermore, no authentication / authorization feature is included in the system which means any agent can access any application module and perform unauthorized operation. Seven agents are being used in ABSAMN and the administrator has to assign rights / permission to each one using eJADE-S inferface.

No	Agent Name	Rights / Permissions
1	Master Controller Agent	Create, Kill, Send Messages, Suspends, File Sharing
2	Controller Agent	Create, Send Messages, File Sharing
3	Monitor Agent	Task performing, File Sharing
4	Action Agent	Start Or Kill Process, Receive Messages
5	Statistical Agent	Task Performing
6	Information Agent	Store Information and Send Messages
7	Messaging Agent	Send Receive Messages

Table 1. Shows the rights / permissions assigned to ABSAMN agents.

Once the administrator assigns the rights to each ABSAMN agent, eJADE-S configuration files (policy and password) are generated which contains encrypted contents. After that eJADE-S library files are added to ABSAMN project and secured ABSAMN jar file is generated. After securing the ABSAMN, agents needs to authenticate / authorize with the system and can only perform the actions which are assigned to them by the administrator.

5 Conclusion

Java Agent Development Framework (JADE) is one of the better and most often used frameworks for the implementation of MAS. The architecture layer of JADE is unsecure which implies that any application developed with JADE is also unsecure and can easily get infected by different attacks. JADE-S [2] has been design to overcome the security weakness of JADE; however, it still has many security weaknesses. In this paper, we have proposed Encrypted JADE-S (eJADE-S) for securing Multi-agent based applications. Cryptography new algorithm using two S-box for encryption / decryption is integrated in JADE-S to overcome the existing security issues. The proposed framework has been tested on larger number of applications, the experimental results shows the efficiency and effectiveness of the same.

6 References

[1] Salvatore Vitabile, Vincenzo Conti, Carmelo Militello, Filippo Sorbello "An extended JADE-S based framework for developing secure Multi-Agent Systems", Computer Standards & Interfaces, Volume 31, Issue 5, September 2009, Pages 913-930.

[2] X. Vila, A. Schuster, A. Riera "Security for a Multi-Agent System based on JADE", Computers & Security, Volume 26, Issue 5, August 2007, Pages 391-400.

[3] S. Venkatesan, C. Chellappan, T. Vengattaraman, P. Dhavachelvan, Anurika Vaish "Advanced mobile agent security models for code integrity and malicious availability check" Journal of Network and Computer Applications, Volume 33, Issue 6, November 2010, Pages 661-671.

[4] T. Sander and C. F. Tschudin, “Protecting mobile agents against malicious hosts” Mobile Agents and Security, Lecture Notes in Computer Science, Vol. 1419, pp. 44-60, 1998.

[5] M. Loulou, Sfax ENIS , M. Tounsi, A.H. Kacem, M. Jmaiel, "A Formal Approach to prevent Attacks on Mobile Agent Systems" The International Conference on Emerging Security Information, Systems, and Technologies (SecureWare), Pages 42-47, 2007.

[6] Umar Manzoor, Samia Nefti, Yacine Rezgui "Categorization of malicious behaviors using ontology-based cognitive agents" Data & Knowledge Engineering, Volume 85, May 2013, Pages 40-56

[7] Umar Manzoor, Samia Nefti "iDetect: Content Based Monitoring of Complex Networks using Mobile Agents", Applied Soft Computing, Volume 12, Issue 5, May 2012, Pages 1607-1619

[8] Java Agent DEvelopment Framework (JADE), http://jade.tilab.com/

[9] Foundation for Intelligent Physical Agents (FIPA), http://www.fipa.org/

[10] K.A Karasavvas, R Baldock, A Burger "Bioinformatics integration and agent technology" Journal of Biomedical Informatics, Volume 37, Issue 3, June 2004, Pages 205-219

[11] Chuan-Jun Su, Chia-Ying Wu "JADE implemented mobile multi-agent based, distributed information platform for pervasive health care monitoring" Applied Soft Computing, Volume 11, Issue 1, January 2011, Pages 315-325

[12] Umar Manzoor, Samia Nefti "An agent based system for activity monitoring on network – ABSAMN" Expert Systems

with Applications, Volume 36, Issue 8, October 2009, Pages 10987-10994

[13] Umar Manzoor, Samia Nefti "QUIET: A Methodology for Autonomous Software Deployment using Mobile Agents" Journal of Network and Computer Applications, Volume 33, Issue 6, November 2010, Pages 696-706

[14] David Isern, David Sánchez, Antonio Moreno "Agents applied in health care: A review" International Journal of Medical Informatics, Volume 79, Issue 3, March 2010, Pages 145-166

[15] Jiannong Cao, Yudong Sun, Xianbin Wang, Sajal K. Das "Scalable load balancing on distributed web servers using mobile agents" Journal of Parallel and Distributed Computing, Volume 63, Issue 10, October 2003, Pages 996-1005.

Effect of Demand-Side Management in Electricity Price/Load Forecasting in Smart Grids

H. Shayeghi
Department of Electrical Engineering
University of Mohaghegh Ardabili
Ardabi, Iran

A. Ghasemi
Department of Electrical Engineering
University of Mohaghegh Ardabili
Ardabi, Iran

H. A. Shayanfar*
Department of Electrical Engineering
College of Technical and Engineering
South Tehran Branch
Islamic Azad University
Tehran, Iran

hshayeghi@gmail.com, ghasemi.agm@gmail.com, hashayanfar@gmail.com

Abstract- ***Electricity price and load forecasting are two important problems for market participants and independent system operators (ISO) in smart grid environments. Most existing papers predict price and load separately, while, the aggregate reaction of consumers can potentially shift the demand curve in the market, resulting in prices that may differ from the initial forecasts. In this regards, demand-side management (DSM) constructs the customers responsible for improving the efficiency, reliability and sustainability of the power system. In this paper, we proposed a new multi-input multi-output (MIMO) system which can consider the interaction between load and price. Therefore, proposed Least Squares Support Vector Machine (LSSVM) to model the nonlinear pattern in price and load. Also, used discrete wavelet transform (DWT) to make valuable subsets. Moreover, proposed feature selection to select best input candidates. Finally, the MIMO-based LSSVM parameters are optimized by artificial bee colony (ABC) algorithm. Simulations carried out NEPOOL region (courtesy ISO New England) electricity market data, and showing that the proposed algorithm has good potential for simultaneous forecasting of electricity price and load in smart grids.***

Keywords: ***Electricity Market, price and load forecast, smart grid, DSM, LSSVM.***

I. INTRODUCTION

Growing recognition of the electricity grid modernization to enable novel electricity consumption and generation models has found expression in the infrastructure of the smart grid concept. In fact, the current electricity grid performs rather stable, but, issues such as improving the energy efficiency, the purpose of large-scale renewable energy integration and the reduction in environment emission need a new grid model which is called smart grid [1] as shown in Fig 1. However, it has been already recognized that the Smart Grid is a new electricity network, which highly integrates the advanced sensing and measurement technologies, information and communication technologies (ICTs), analytical and decision-making technologies, automatic control technologies with energy and power technologies and infrastructure of electricity grids [2].

Furthermore, smart grids makes a two-way flow exchange with customers, providing advanced information and options, power export capacity, customers and improved energy efficiency. Note that there are great correlations between these participants in smart grid. Therefore, all services in smart grid environment have a complexly compared to current grid services. Therefore, there is a strong requirement in the electric power markets for accurate and robust tools that properly forecast electricity price and load signals in simultaneous form, opposite to available methods which only forecast price and load separately [3].

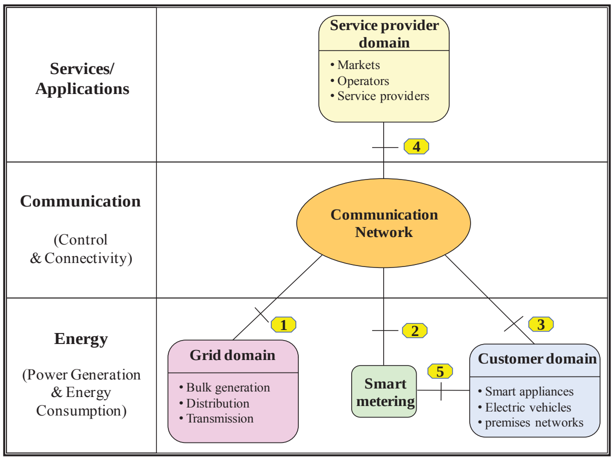


Figure 1. Simplified smart grid domain model

In recent years, some papers have been dedicated to forecasting in electricity markets with this correlation between load and price. In [4] proposed a framework that used MIMO forecasting engine for joint price and demand prediction with data association mining (DAM) algorithms. In [5] a two-stage integrated price and load forecasting framework is developed. At each stage, a hybrid time-series and adaptive wavelet neural network (AWNN) model is used, in which multivariate autoregressive integrated moving average catches the linear relationship of price and load log return series, generalized autoregressive conditional heteroscedastic unveils heteroscedastic character of residuals and AWNN presents non-linear impacts. In [6] proposed a hybrid algorithm based on LSSVM-MIMO+ wavelet packet transform (WPT)+Quasi-Oppositional Artificial Bee Colony (QOABC) to forecast price and load in smart grid with their correlations. In [7], initial next-day load forecasts are generated using a multi-layer perceptron in the first step. In the second step, it is assumed the next-day electricity prices are known and the variation of prices with respect to the previous day's price is quantified. Fuzzy systems are used to extract

* Corresponding Author. E-Mail: hashayanfar@gmail.com (H. A. Shayanfar)

price-load variation patterns. Finally, the identified patterns are employed to modify and improve initial load forecasts. The load forecasting method in [8] is similar to that of [7] except that RBF neural networks and Adaptive Neural Fuzzy Inference Systems (ANFIS) are employed to generate the load forecasts. A mixed price and demand forecasting method is presented in [9], in which, price and demand are iteratively predicted, and the forecasts are considered in the candidate input set of the subsequent predictors. For forecasting both demand and price, historical price and demand information are considered as input in [9].

In this paper, LSSVM based on MIMO model is proposed which is able to explain linear problems quicker with a more straight forward technique. Moreover, it is necessary to find the optimal LSSVM parameters. In order to achieve this purpose, we used ABC algorithm. Also, to get better the forecast accuracy, the DWT model has been utilized because it is a power tool for noise diminution.

II. Problem Expression

2.1. DWT

DWT is a powerful tool for noise reduction without destroying the dynamics of the original price series. DWT are discretely translatable and scalable by

$$\psi_{j,k}(x) = a_0^{j/2}\psi(a_0^j x - kb_0) \tag{1}$$

where b_0 and a_0 are refereed to translation and scale parameters, respectively, and k and j are two integer numbers. Therefore, the DWT can be defined by

$$DWT(j,k) = \int_{-\infty}^{+\infty} x(t).\psi_{j,k}(x)dx \tag{2}$$

According to these explanations, a DWT consist of two parts; decomposition and reconstruction. In the decomposition phase, the low-pass filter removes the higher frequency components of the signal and high-pass filter picks up the remaining parts. Then, the filtered signals are down-sampled by two and the results are called approximate coefficients and detail coefficients [10]. The reconstruction is just a reversed process of the decomposition and for perfect reconstruction filter banks. Figure 2 depicts the corresponding wavelet decomposition.

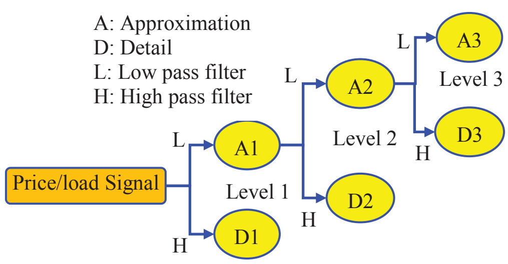


Figure 2. Discrete wavelet decomposition tree

2.2. LSSVM

Given a training set of N data points $\{y_k, x_k\}_{k=1}^{N}$, where $x_k \in R^n$ is the k-th input pattern and $y_k \in R$ is the k-th output pattern, the classifier can be constructed using the support vector method in the form:

$$y(x) = sign\left[\sum_{k=1}^{N} \alpha_k y_k K(x, x_k) + b\right] \tag{3}$$

Where, α_k are called support values and b is a constant. The $K(\cdot,\cdot)$ is the kernel, which is $K(x, x_k) = \exp\{-\|x - x_k\|_2^2 / \sigma^2\}$ (RBF), where κ, θ and σ are constants. For instance, the problem of classifying two classes is defined as:

$$\begin{cases} w^T \phi(x_k) + b \geq +1 & if\ y_k = +1 \\ w^T \phi(x_k) + b \leq -1 & if\ y_k = -1 \end{cases} \tag{4}$$

This can also be written as:

$$y_k[w^T \phi(x_k) + b] \geq 1, \quad k = 1,\dots,N \tag{5}$$

Where, $\phi(\cdot)$ is a nonlinear function mapping of the input space to a higher dimensional space. LS-SVM classifier:

$$\min_{w,b,e} J_{LS}(w,b,e) = \tfrac{1}{2} w^T w + \gamma \tfrac{1}{2} \sum_{k=1}^{N} e_k^2 \tag{6}$$

Subjects to the equality constraints:

$$y_k[w^T \phi(x_k) + b] = 1 - e_k, \quad k = 1,\dots,N \tag{7}$$

The Lagrangian is defined as:

$$L(w,b,e;\alpha) = J_{LS} - \sum_{k=1}^{N} \alpha_k \left\{ y_k[w^T \phi(x_k) + b] - 1 + e_k \right\} \tag{8}$$

With Lagrange multipliers $\alpha_k \in R$ (called support values). The conditions for optimality are given by:

$$\begin{cases} \frac{\partial L}{\partial w} = 0 & \rightarrow & w = \sum_{k=1}^{N} \alpha_k y_k \phi(x_k) \\ \frac{\partial L}{\partial b} = 0 & \rightarrow & \sum_{k=1}^{N} \alpha_k y_k = 0 \\ \frac{\partial L}{\partial e_k} = 0 & \rightarrow & \alpha_k = \gamma e_k \\ \frac{\partial L}{\partial \alpha_k} = 0 & \rightarrow & y_k[w^T \phi(x_k) + b] - 1 + e_k = 0 \end{cases} \tag{9}$$

For $k = 1,\dots,N$. After elimination of w and e one obtains the solution:

$$\begin{bmatrix} 0 & Y^T \\ Y & ZZ^T + \gamma^{-1} I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1_v \end{bmatrix} \tag{10}$$

With:

$$\begin{aligned} &Z = [\phi(x_1)^T y_1; \dots; \phi(x_N)^T y_N], Y = [y_1; \dots; y_N], \\ &1_v = [1;\dots;1], e = [e_1;\dots;e_N], \alpha = [\alpha_1;\dots;\alpha_N] \end{aligned} \tag{11}$$

Mercer's condition is applied to the matrix $\Omega = ZZ^T$ with:

$$\Omega_{kl} = y_k y_l \phi(x_k)^T \phi(x_l) = y_k y_l K(x_k, x_l) \tag{12}$$

The kernel parameters, i.e. σ for RBF kernel, can be optimally chosen by optimizing an upper bound on the VC dimension [11]. The support values α_k are proportional to the errors at the data points in the LSSVM case, while in the standard SVM case many support values are typically equal to zero.

2.3. Feature selection

The proposed feature selection can be summarized as follows:

Stage I: Mutual information (MI) between $CI(t) \in \{X(t), Y_1(t), Y_2(t), \ldots, Y_m(t)\}$ (candidate input) and $x(t)$ (target feature), that is $MI[CI(t), x(t)]$, is calculated based on the binomial distribution method illustrated in [13]. $MI[CI(t), x(t)]$ with high value means $CI(t)$ is a more relevant feature for forecasting $x(t)$. Consequently, the candidate input features of $\{X(t), Y_1(t), Y_2(t), \ldots, Y_m(t)\}$ are sorted based on their mutual information with the target feature such that a higher $MI[CI(t), x(t)]$ value results in a higher rank. Afterward, the candidate inputs with $MI[CI(t), x(t)]$ value bigger than TH_1 (relevancy threshold) are retained as the relevant features of the forecast process and the other candidate inputs are filtered out. This stage is irrelevancy filter.

Stage II: Let $S_1 \subset \{X(t), Y_1(t), Y_2(t), \ldots, Y_m(t)\}$ denote a subset of candidate inputs chosen in pervious stage. Higher value of MI between two selected candidates $CI_k(t) \in S_1$ and $CI_m(t) \in S_1$, i.e. $MI[CI_k(t), CI_m(t)]$, means more common information between the candidate inputs $CI_k(t)$ and $CI_m(t)$ and so these candidates have a higher level of redundancy. The following redundancy criterion $RC[.]$ calculates the redundancy of each selected feature $CI_k(t) \in S_1$ with the other candidate inputs of S_1:

$$RC\left[CI_k(t)\right] = \max_{CI_m(t) \in S_1} \left(MI\left[CI_k(t), CI_m(t)\right]\right) \quad (13)$$

We can rank the candidate inputs of S_1 based on the redundancy measure of (13) so that a higher value of $RC[CI_k(t)]$ means $CI_k(t)$ is a more redundant feature or equivalently a less informative candidate input. If $RC[CI_k(t)]$ becomes greater than a redundancy threshold $TH2$, $CI_k(t)$ is considered as a redundant candidate input and so between this candidate and its partner, one feature should be filtered out. For instance, suppose that

$$\arg \max_{CI_m(t) \in S_1} \left(MI\left[CI_k(t), CI_m(t)\right]\right) = CI_r(t), \quad CI_r(t) \in S_1 \quad (14)$$

In fact, $CI_r(t)$ has the highest mutual information with $CI_k(t)$ among the features of S_1. Between $CI_k(t)$ and its partner $CI_r(t)$, one variable should be eliminated. For this purpose, the relevancy factors of these features, i.e. $MI[CI_k(t), x(t)]$ and $MI[CI_r(t), x(t)]$, are considered and the feature with less relevancy factor (less relevant feature or less effective feature for the forecast process) is filtered out. It is possible that more than two features be redundant such that only one of them is eligible to be retained. Therefore, the redundancy filtering process is repeated for all input features of S_1 until no redundancy calculates of (13) become greater than TH_2.

The redundancy filter, as described above, constitutes the second stage of the proposed two stage feature selection technique. The subsets of feature $S_2 \subset S_1$ that pass the redundancy filter are finally selected candidate inputs by the proposed technique. The candidate features of S_2 are considered as the inputs of the forecast engine. TH_1 and TH_2 are the adjustable parameters of the feature selection technique, which are fine-tuned by the cross-validation method.

2.4. ABC algorithm

In this section, the standard ABC is briefly reviewed [12]. The process of the ABC algorithm is presented as follows:

Step 1. Initialization: generate random population and calculate their fitness values. This population and fitness values called employed bees and nectar amounts, respectively.

Step 2. Move the onlookers: an onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source with a probability related to its nectar amount by "Eq. (15)", this method, known as roulette wheel selection method. The movement of the onlookers follows:

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_i} \quad (15)$$

Where, p_i and SN are probability of selecting the i^{th} employed bee and number of employed bees, and fit_i is the fitness value of the solution.

$$x_{ij}(t+1) = \theta_{ij}(t) + \varphi[\theta_{ij}(t) - \theta_{kj}(t)] \quad (16)$$

Where, $k \in \{1,2,...,BN\}$ and $j \in \{1,2,...,D\}$ are randomly chosen indexes and x_i, t, θ_k and $\phi()$are the position of the i^{th} onlooker bee, the iteration number, the randomly chosen employed bee and random variable between (-1,1), respectively. D is the number of dimension of optimization problem. BN is number of onlooker bee.

Step 3. Move the scouts: when selected a food source, all the employed bees associated with it abandon the food source, and become scout. The scouts are moved by:

$$\theta_{ij} = \theta_{ij,\min} + r(\theta_{ij,\max} - \theta_{ij,\min}) \quad (17)$$

Where, r denotes a random factor between 0 and 1. θ_{ijmax} and θ_{ijmin} are lower and upper boundary of x_i, respectively.

Step 4. Update the best food source found so far: Memorize the best food source found so far.

Step 5. Termination checking: checking termination criteria satisfied, if it is satisfied then stop algorithm otherwise go to step 2.

2.5. DSM model

Power system distribution networks are designed for peak loads. For optimum utilization of network capacity, utilities employ DSM with objective of minimum possible peak load. DSM ensures maximum load factor and thus maximizing total profit of utilities. For better insight of different DSM techniques, load distribution over a day is represented by a normal distribution curve as in Fig 3.

A. Valley Filling: The loads during off peak hours, i.e., region III, are increased to achieve flatter profile.

$$[\uparrow p_3] \Leftrightarrow [load\ addition] \quad (18)$$

B. Load Shifting: The shift able loads during peak hours are shifted to off-peak hours, resulting lower peak of curve and a flatter profile i.e. region 1 are shifted to region 2 and 3.

$$[\uparrow p_3 \cap \uparrow p_2] \Leftrightarrow [\downarrow p_1] \quad (19)$$

C. Peak Clipping: The load from peak hours (region 1) is reduced like scheduled power cuts.

$$[\downarrow p_1] \Leftrightarrow [load\ removal] \quad (20)$$

D. Energy Conservation: This applied when reduction in load is required all over the load curve.

$$[\downarrow p_1 \cap \downarrow p_3 \cap \downarrow p_2] \Leftrightarrow [load\ removal] \quad (21)$$

E. Load Building: It employed when increased energy consumption is required due to surplus production.

$$[\uparrow p_1 \cap \uparrow p_3 \cap \uparrow p_2] \Leftrightarrow [load\ addition] \quad (22)$$

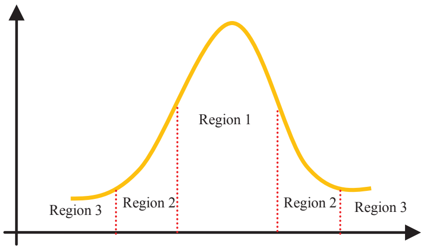


Figure 3. Representation of load profile; P_1: Probability of load lying in region 1, P_2: Probability of load lying in region 2, P_3: Probability of load lying in region 3.

III. Effect of DSM in Forecasting

Currently let's see what happens if a component of DSM has the ability to response to the price variations. This state has been shown in Fig 4. In this case, the demand curve has two parts [4]. The first one is that which cannot response to the electricity price variations (price taking part). This part is shown as a vertical line. The second part is related to the price responsive part of demand and has a minus slope in Fig 4. In fact, this part represents the DSM of the system. As can be seen in Fig. 4, proper DSM in the system can protect the demand-side from price spikes. Therefore the price taking part of the demand will also benefits from the DSM of the system. As mentioned before, by emerging the smart grids the response of consumers to the price signals of the electricity market will not be weak and the spot electricity price and demand will mutually affect each other.

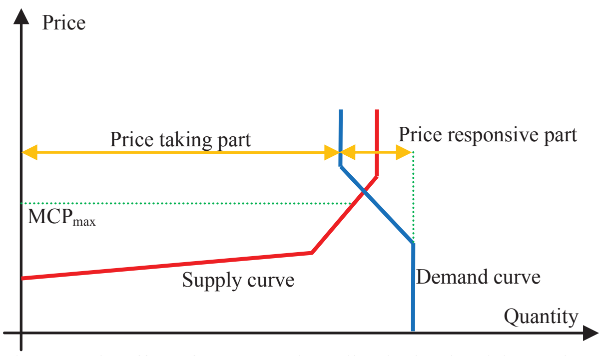


Figure 4. The effect of DSM on price spikes in the electricity markets.

Therefore in this new electricity grid, load forecasting and price forecasting cannot be implemented independently. In fact, the penetration level of smart grid technologies like advanced metering infrastructure (AMI) systems, smart monitors and smart controllers and also the arrangement of different kinds of consumers (i.e. the ratio of industrial, commercial, and residential consumers) will affect the overall response of the demand-side to the price signals. Although, in long-term these factors will become fixed and the overall response of the demand-side will be stable, but they have great effect on DR in the transition period. The structure of the forecaster is also likely to be changed due to the dynamic relation between load and price. Fig 5, schematically shows possible inputs and outputs of the future load and price forecasting systems.

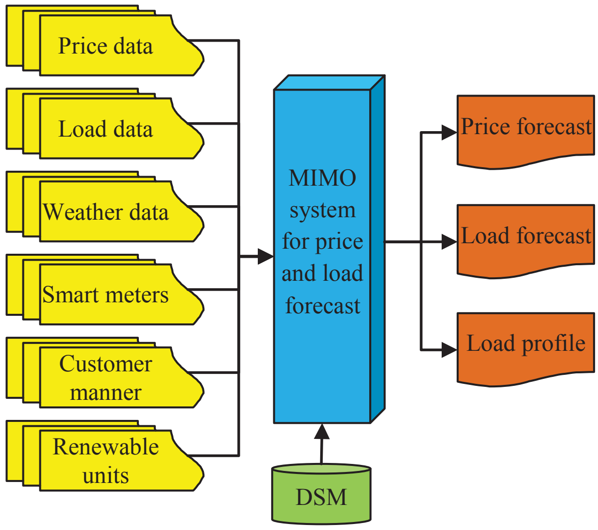


Figure 5. Structure of price and load forecast in smart grid with DSM.

IV. Proposed Price and Load Forecasting Based MIMO Channel

In rest of this section, the proposed forecast is describe,

Stage 1: Firstly, the historical data for price and load are sorted as input data. Now, with DWT system, these signals individually are decomposed in detail (price; D_p and load; D_L) and approximate (price; A_p and load; A_L) subsets. Then, proposed feature selection is used to select best data with more relevancies and least redundancy.

Stage 2: When the Multi-Input Multi-Output (MIMO) is launched with $\{x_{p1}, x_{p2},\ldots, x_{pn}\}$ and $\{x_{l1}, x_{l2},\ldots, x_{lm}\}$ vectors then the training process will be starting. This performance about learning is shown in Fig 6 with more detail.

Stage 3: This stage used inverse DWT transform to guess the hourly prices for d^{th} day by means of the estimates d^{th} day of the constitutive subsets. In other words, this model is used in turn in order to reconstruct the estimate signal for price and load, i.e.,

$$W^{-1}(\{a_h^{est}, b_h^{est}, c_h^{est}, d_h^{est}; \\ h = T+1,\ldots,T+24\}) = P_h^{W,est} \quad (23)$$

The objective function is calculated by the following equation,

$$Obj = \frac{1}{N}\left(\sum_{i=1}^{N}\left(\frac{P_{act_i} - P_{for_i}}{P_{act_i}} + \frac{L_{act_i} - L_{for_i}}{L_{act_i}}\right)\right) \quad (24)$$

Where, P_{act} and P_{for} are the forecast and actual values of price signal, L_{act} and L_{for} are the forecast and actual values of load signal, respectively.

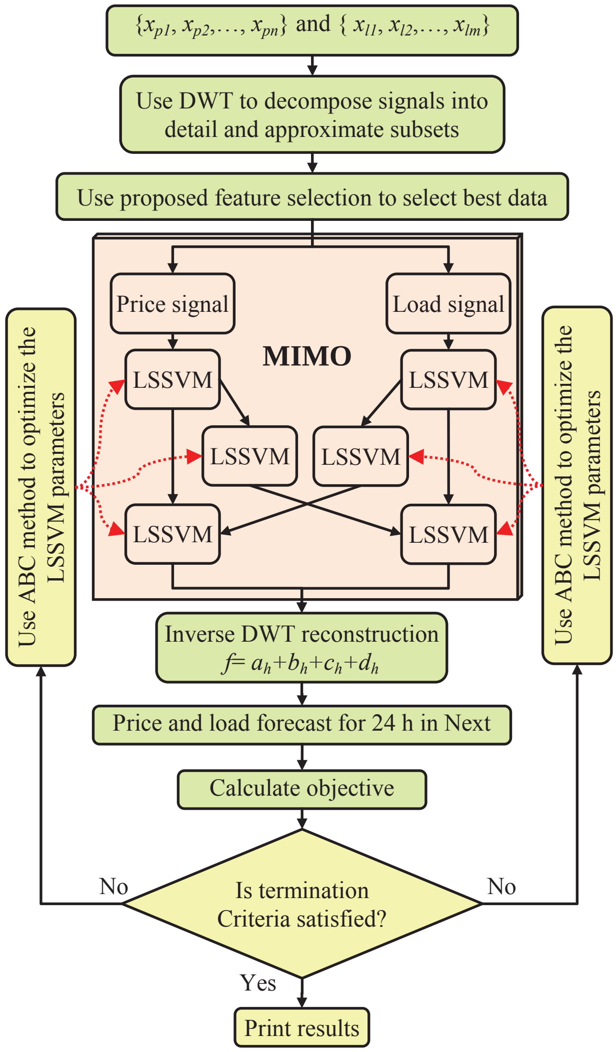


Figure 6. Schematic diagram of the proposed hybrid forecasting methodology

Stage 4: The aim of this stage is error minimizing, in fact, we done this process by adjusting the LSSVM parameters using forecasted and actual values. If the maximum number of training process ($iter_{Max}$) is reached, then finish. If not, go to *stage 2*.

V. Simulation Results and Discussion

The model are trained on hourly data from the New England Power Pool (NEPOOL) region (courtesy ISO New England) from 2004 to 2007 and tested on out-of-sample data from 2008 [14]. The most widely used criterion to measure forecasting error is the Mean Absolute Percentage Error (MAPE), the Forecast Mean Square Error (FMSE), and the Error Standard Deviation (ESD). This accuracy is calculated as a function of the actual prices that occur. The daily MAPE can be given by:

$$MAPE_{day/week} = \frac{1}{N}\sum_{i=1}^{N}\frac{|P_{iACT} - P_{iFOR}|}{P_{AVE-ACT}} \tag{25}$$

$$P_{AVE-ACT} = \frac{1}{N}\sum_{i=1}^{N} P_{iACT} \tag{26}$$

Where, P_{iFOR} and P_{iACT} are referred to forecasted and actual values, and $P_{AVE\text{-}ACT}$ is the average value of P_{iACT} as given in Eq. (26) to avoid the adverse effects of prices close to zero. Furthermore, the FMSE is the square root of the average of 24 (daily) squared differences between the actual prices and the forecasted ones;

$$FMSE_{day/week} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_{iACT} - P_{iFOR})^2} \tag{27}$$

The ESD index, one of the important performance criteria, is given by:

$$ESD_{day/week} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(E_i - E_i^{Ave}\right)^2},$$
$$E_i = P_{iACT} - P_{iFOR}, E_i^{Ave} = \frac{1}{N}\sum_{i=1}^{N} E_i \tag{28}$$

The hourly electricity price and load signals of the NYISO electricity market are shown in Figs. 7 and 8, respectively. Also, to the reader's convenience, Fig. 9 shows the relation between load and price signals. Figure 10 shows the weekly forecast load signal, and, similarly, Fig 11 is referred to price signal in this period.

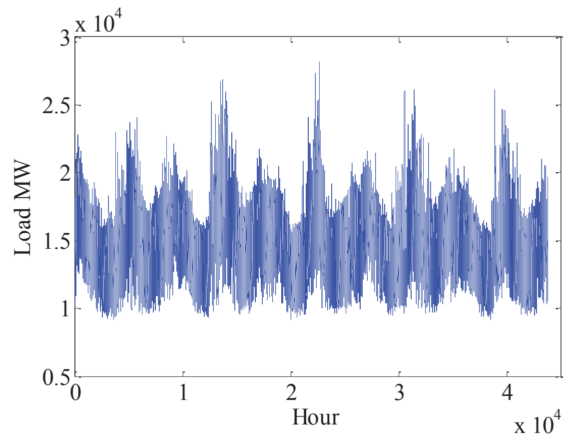


Figure 7. Load signal in the NEPOOL electricity market.

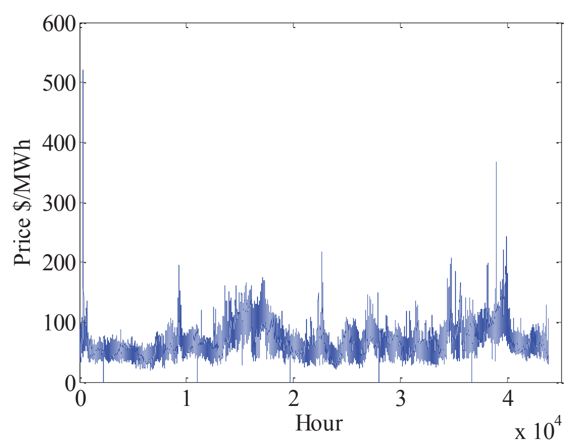


Figure 8. Price signal in the NEPOOL electricity market.

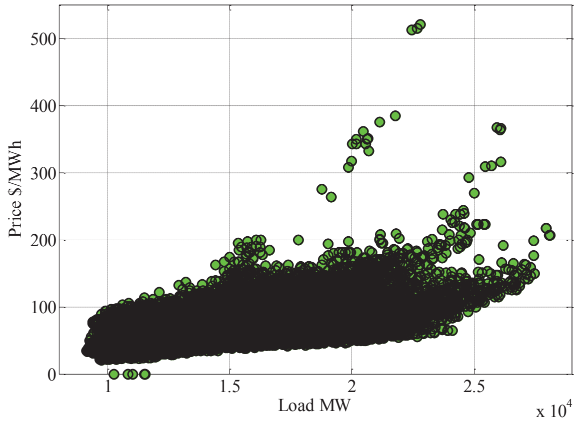


Figure 9. Relation between load price signals in the NEPOOL electricity market.

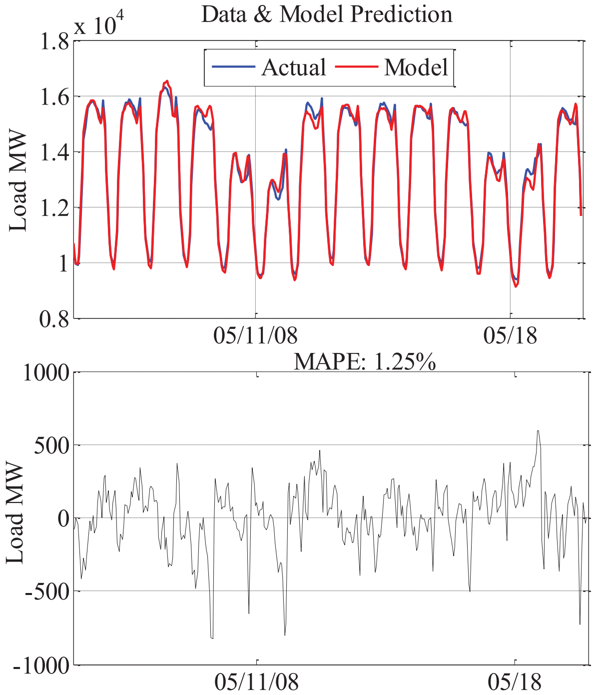


Figure 10. Weekly forecasted electricity load result for the NEPOOL electricity market.

The minute differences between the target value and predicted value may be due to measurement errors or special occasion day or severe weather conditions in present year. Variations due to seasonal changes are taken care by the inputs x_p and x_l viz. load and price of previous day and average load and price of previous week, as seasonal changes are gradual. Load and price on same day of previous week caters for weekend loads and prices. Effect of previous year's variations is avoided by using normalized data; however one limitation of model is that it would not be able to do prediction for holiday, rainy day or severe weather conditions. The network performance curve, between MSE and epochs is shown in Fig 12. The best performance achieved for validation is at epoch 67 with MSE of 3.28×10^{-3}. The regression plot between network response and target is shown in Fig. 13. The R (correlation coefficient value) values for training, validation and testing are 0.9943, 0.9903 and 0.9939 respectively. The overall R value is 0.9942 resulting in very close prediction.

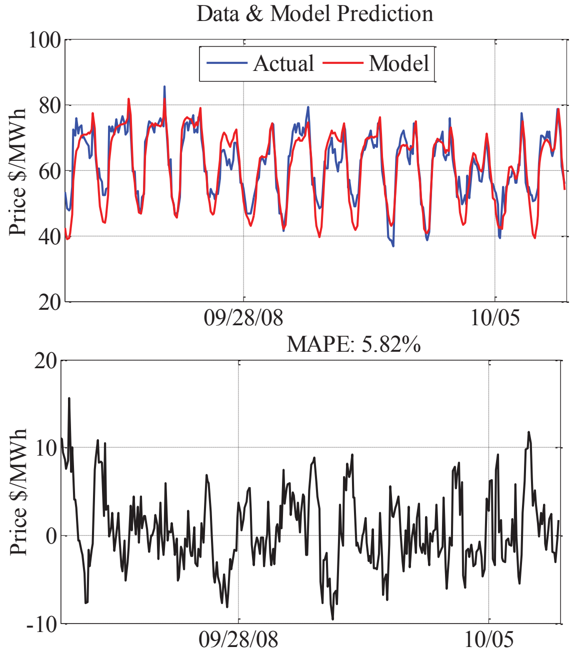


Figure 11. Weekly forecasted electricity price result for the NEPOOL electricity market.

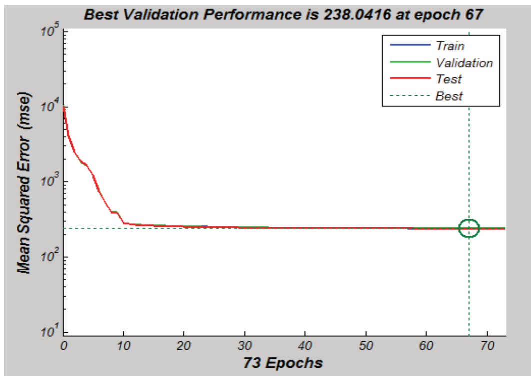


Figure 12. Performance curve for LSSVM training.

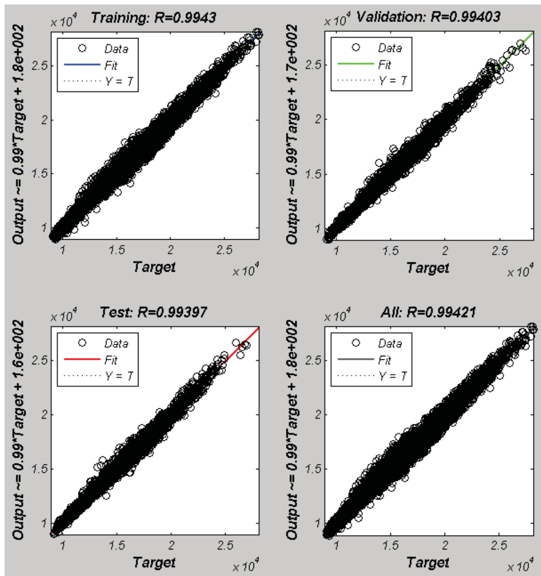


Figure 13. Regression plot.

VI. CONCLUSIONS

A hybrid short-term load and price forecast with a new feature selection framework is proposed in this paper to enhance the forecast accuracy in smart grid. The correlation between load and price signals is modeled with a LSSVM learning algorithm. In other words, a hybrid forecasting framework is proposed which takes into account the bidirectional price-demand relationships when forecasting electricity market price and demand. Hybrid models combine the capabilities of different modeling approaches and are reported to improve forecasting performance in previous applications. The New England hourly price and load data are used to evaluate the performance of the feature selection algorithm. Simulation results demonstrate the efficiency of the selected input sub-series to improve the forecast accuracy. The results also show that the forecast errors for the special days are significantly decreased by including all the selected sub-series in the input set. The selected sub-series are used as the inputs to the individual LSSVMs and the outputs are combined using the determined weighting coefficients to provide the load and price forecast.

REFERENCES

[1] P. Siano, "Demand response and smart grids-A survey," Renewable and Sustainable Energy Reviews, 2014;30:461–478.

[2] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou,G. Kalogridis, M. Sooriyabandara, et al, "Smart grid communications: overview of research challenges, solutions, and standardization activities," IEEE Commun Surv Tutorials, 2013;15 (1):21–38.

[3] J. Aghaei, M.I. Alizadeh, "Demand response in smart electricity grids equipped with renewable energy sources: a review," Renewable Sustainable Energy Rev, 2013;18:64–72.

[4] M. Amir, Z. Hamidreza, D.R. William, "Electricity Price and Demand Forecasting in Smart Grids," IEEE Trans. smart grid, 2012;3(2):664–674.

[5] L. Wu, M. Shahidehpour, "Hybrid model for integrated day-ahead electricity price and load forecasting in smart grid," IET Generation, Transmission & Distribution, 2014;8(12):1937–1950.

[6] H. Shayeghi, A. Ghasemi, M. Moradzadeh, M. Nooshyar, "Simultaneous day-ahead forecasting of electricity price and load in smart grids," Energy Conversion and Management, 2015;95:371–384.

[7] A. Khotanzad, E. Zhou, and H. Elragal, "A neuro-fuzzy approach to short-term load forecasting in a price-sensitive environment," IEEE Trans. Power Syst., 2002;17(4):1273–1282.

[8] Z. Yun, Z. Quan, S. Caixin, L. Shaolan, L. Yuming, and S. Yang, "RBF neural network and ANFIS-based short-term load forecasting approach in real-time price environment," IEEE Trans. Power Syst., 2008;23(3):853–858.

[9] N. Amjady, A. Daraeepour, "Mixed price and load forecasting of electricity markets by a new iterative prediction method," Electr. Power Syst. Res., 2009; 79(9):1329–1336.

[10] H. Shayeghi, A. Ghasemi, "Day-ahead electricity prices forecasting by a modified CGSA technique and hybrid WT in LSSVM based scheme," Energy Conversion and Management, 2013;74:482–491.

[11] VN. Vapnik, "The nature of statistical learning theory," New York: Springer; 1995.

[12] H. Shayeghi, A. Ghasemi, "A modified artificial bee colony based on chaos theory for solving non-convex emission/economic dispatch," Energy Conversion and Management, 2014;79:344–54.

[13] N. Amjady, F. Keynia, Day-ahead Price Forecasting of Electricity Markets by Mutual Information Technique and Cascaded Neuro-Evolutionary Algorithm, IEEE Trans. Power Syst., 2009;24(1):306-318.

[14] http://www.iso-ne.com/.

Hossein Shayeghi received the B.S. and M.S.E. degrees in Electrical and Control Engineering in 1996 and 1998, respectively. He re ceived his Ph.D. degree in Electrical Engineering from Iran University of Science and Technology, Tehran, Iran in 2006. Currently, he is a Professor in Technical Engineering Department of University of Mohaghegh Ardabili, Ardabil, Iran. His research interests are in the application of robust control, artificial intelligence and heuristic optimization methods to power system control design, operation and planning and power system restructuring. He has authored and co-authored of five books in Electrical Engineering area all in Farsi, two book chapters in international publishers and more than 180 papers in international journals and conference proceedings. Also, he collaborates with several international journals as reviewer boards and works as editorial committee of three international journals. He has served on several other committees and panels in governmental, industrial, and technical conferences. He was selected as distinguished researcher of the University of Mohaghegh Ardabili several times. In 2007 and 2010 he was also elected as distinguished researcher in engineering field in Ardabil province of Iran. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE. Currently, he is head of Ardabil Technology Incubation Center (ATIC) at University of Mohaghegh Ardabili since 2008.

Ali Ghasemi received the B.Sc. and M.Sc. (Honors with first class) degree in electrical engineering from Isfahan University of Technology (IUT), Isfahan, and University of Mohaghegh Ardabili (UMA), Ardabil, Iran, in 2009 and 2011, respectively. Currently he is pursuing the Ph.D. degree in the electrical engineering and computer science of UMA. His research interests are application of forecast methods, operation adaptive and robust control of power systems, Planning, Power System Restructuring and applications of heuristic techniques. He has authored of a book in Electrical Engineering area in Farsi, and more than 120 papers in reputable international journals and conference proceedings. Also, he collaborates as editorial committee and reviewer of 20 international journals. He is a member of the Iranian Association of Electrical and Electronic Engineers (IAEEE). In 2012 and 2013, He received the award of the 4^{th} *Electric Power Generation Conference* and *UMA* for his M.Sc. thesis. He is the recipient honor M.Sc and Ph.D student Award of *UMA*, 2014. He received the 2013 best young researcher award of the *Young Researcher and Elite Club*.

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems, Smart Grids and Reactive Pricing in Deregulated Power Systems. He has published more than 497 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Maximization of the Resource Production in RTS Games using Planning and Scheduling

Thiago F. Naves, Carlos R. Lopes
(tfnaves@ufu.br) (crlopes@ufu.br)
Faculty of Computing
Federal University of Uberlândia
Avenida João Naves de Ávila 2121, Uberlândia (MG), Brazil, ZIP 38400-902

***Abstract*—Real time Strategy Games are one of the areas of AI that most present challenges related to planning and constraints satisfaction. In these games, the player needs to produce resources to develop his army in order to attack and defend against enemies. The choice of production goals has a direct impact in the strength of the army. In a previous work, we developed an approach based on Simulated Annealing that achieves quality goals by maximizing resource production. In this paper we extend our previous work by introducing a scheduling component. By intercalating planning and scheduling we were able to find better results than our previous approach. The experiments showed that our system was capable of competing with a human player without any restriction.**

Keywords*—Real-Time Strategy Games, Resources, Goals, Search, Planning.

I. Introduction

Real-time strategy (RTS) games are one of the most popular categories of computer games. "StarCraft 2" is a representative game title of this category. These games have different character classes and resources, which are employed in battles. During the battles, a stage of resources production stands out, where the player must develop the maximum of these in a given interval of time so that he can attack and defend himself from enemies. To produce resources is a complex task since each resource has a series of prerequisites that must be satisfied prior to its production. Thus, maximizing the production of resources during a match is important for victory in the game.

The resources in RTS games are all kinds of raw materials, basic construction, military units and civilization. To obtaining a desired set of resources it is necessary to carry out actions. In general we have three sorts of actions related to resources: actions that produce resources, actions that consume resources and actions that collect resources. A sequence of actions that can be executed in the game is called a plan, and this reaches a certain amount of resources, i.e, a goal within the game. One of the most common problems in maximizing resource production is the choice of goal to be achieved. It is also a problem in most of the studies that involve RTS games. These works focus on finding the necessary actions to achieve a goal of resources, however the goals are determined randomly or without quality assurance for planning.

In a previously work, we developed an approach to solve the problem of choice of goal in RTS games through the maximization of resource production [14]. This make it possible to determine a goal quality along with the actions required to achieve it, in a given time interval in the game. The work was proposed using a sequential architecture in relation to a plan of action and execution of this in the game. In this paper, we present a new version of the choices of goal approach, now using an architecture with scheduling of actions, which improves and increases resource maximization.

The domain used for this work is StarCraft, which is considered the game with the highest number of constraints on planning tasks [5]. In our approach we have to generate an initial plan of action to achieve an arbitrary amount of resources. This initial plan is developed using scheduling in its actions. In order to do this, we have developed a planner system able to plan and schedule the necessary actions to achieve a random amount of resources. The initial plan is the basis for exploring new plans in order to find the one that has largest amount of resources.

In this paper, we focus on the task of finding the goal that maximizes resource production, as we stated earlier. Once we have an initial plan, we use Simulated Annealing [1] (SA) to maximize the amount of resources to be achieved. However, our approach is not only used to specify the goal. We developed a consistency checker capable of handling and validating the changes made in the plan, now also maintaining the scheduling characteristic between the actions of the plan. Thus, our approach when specifying a goal to be achieved develops the plan with all the actions to achieve it within the game.

Our work is motivated by the results obtained in [7] and by gaps in the approaches of [4] and [2] with respect to the choice of goal to be achieved during the production of resources. Thus, this research can go toward a new approach to resource production, and can be extended to other applications. The results obtained are encouraging and show the efficiency of the method to find good solutions. We understand that this research is interesting for the AI planning field because it deals with a number of challenging problems such as concurrent activities and real-time constraints.

The remainder of this paper is organized as follows. In section III related work is presented and discussed. Section II lists the characteristics of the problem. Section IV presents the scheduling architecture developed here, along with the two main techniques. Section V discusses the results of the experiments. In section VI we make final considerations about the new version of the approach.

II. Characterization of the Problem

The planning problem in RTS games is to find a sequence of actions that leads the game to a goal state that achieves a certain amount of resources. This process must be efficient. In general, the search for efficiency is related to the time that is spent in the execution of the plan of action (makespan). However, in our approach we seek for actions that increase the amount of resources that raise the army power of a player. This is achieved by introducing changes into a given plan of action in order to increase the resources to be produced, especially the military units, without violating the preconditions between the actions that will be used to build these resources.

To execute any action you must ensure that its predecessor resources are available. The predecessor word can be understood as a precondition to perform an action and the creation of the resource as a effect of its execution. Resources can be labeled in one of the following categories: $Require$, $Borrow$, $Produce$ and $Consume$. Figure 1 shows the precedence relationship among some of the main resources of the domain. These resources are based on StarCraft, which has three different character classes: Zerg, Protoss and Terran. This work focuses on the resources of the class Terran.

```
resource CommandCenter
resource Barracks
resource Scv
resource Firebat
resource Minerals
resource Gas

action build-commandcenter :duration 75 sec.
 :borrow 1 Scv :consume 400 Minerals
 :produce 1 CommandCenter
action build-barracks :duration 50 sec.
 :require 1 CommandCenter :borrow 1 Scv
 :consume 150 Minerals :produce 1 Barracks
action build-scv :duration 13 sec.
 :borrow 1 CommandCenter :consume 50 Minerals 1 Supply
 :produce 1 Scv
action build-firebat :duration 15 sec.
 :require: 1 Academy :borrow 1 Barracks
 :consume 50 Minerals 25 Gas 1 Supply
 :produce 1 Firebat
action collect-minerals :duration 45 sec.
 :require 1 CommandCenter :borrow 1 Scv
 :produce 50 Minerals
action collect-gas :duration 20 sec.
 :require 1 Refinery :borrow 1 Scv
 :produce 25 Gas
```

Fig. 1. Some of the main actions of the resource domain of Strarcraft.

For example, according to Figure 1 to execute an action that produces a $Firebat$ resource you must have a $Barracks$ and $Academy$ available, a certain amount of $Minerals$ and Gas is also required. The $Firebat$ is a military resource. Thus, the action *build-firebat* involves: $Require$ *(1 Academy, 1 Barracks)*, $borrow$ *(1 Barracks)*, $consume$ *(50 Minerals, 25 Gas)* and $produces$ *(1 Firebat)*. The $Barracks$ will be $borrow$, i.e, it is not possible to perform another action until *build-firebat* is finished. The time to build a $Firebat$ is 15 seconds (sec). However, you also need to perform all actions in the shortest time possible, by executing in parallel those that are possible. This situation describes the challenges that surround the planning of actions.

In our approach each plan of action has a time limit, army points, feasible and unfeasible actions. The time limit constrains the maximum value of makespan of the plan of action. The army points are used to evaluate the strength of the plan in relation to its military power. Each resource has a value that defines its ability to fight the opponent. Feasible actions in a plan are those actions that can be carried out, i.e, they have all their preconditions satisfied within the current planning. Unfeasible actions are those that can not be performed in the plan and are waiting for its preconditions to be satisfied at some stage of planning. Unfeasible actions do not consume planning resources and do not contribute to the evaluation of the army points of the plan.

One example of operations in a plan, supposing that a time limit of 170 seconds was imposed to find a goal and the initial plan was set to perform *1 Firebat*. In this case, the completed plan of action has the following actions: *(8 collect-minerals, 1 collect-gas, 1 build-refinery, 1 build-barrack, 1 build-academy, 1 build-firebat)* with 160 makespan and 16 army points. There are several operations that could be made in this plan and that would leave unfeasible actions. For example, exchange one of the actions *collect-minerals* that is a precondition of $Barracks$ by any other action would leave the resource unfeasible. In consequence the resources $Firebat$ and $Academy$ would also be unfeasible, since these have $Barracks$ as one of their predecessors. With this, the plan would have 0 army points.

Some specific combinations of operations can make the plan elevate its resource production and thus its army strength. For example, if it were made the insertion of an *build-marine* action in place of the action collect-gas. The new action will be feasible in the plan, as it will use the minerals left by the action *build-firebat* that became unfeasible due to lack of gas. Now, if in the next two operations the actions of *collect-minerals* and *collect-gas* are inserted in the plan, the resource $Firebat$ comes back to being feasible. This happens because the action of minerals would be made by a resource Scv which is idle and the scheduling process puts its execution time as early as possible in the plan. The gas action would be performed at the same time before leaving the plan, also by the scheduling process. Thus, the final goal now has 28 points without exceeding the time limit, still having the same makespan of 160 sec. The native state of resources when starting a match in StarCraft is *4 Scv, 1 CommandCenter*.

Results, as in the example presented above, are motivators for this research.

III. Related Work

There is little research in maximizing goals of resources production in RTS games. One of the reasons is the complexity involved in searching and managing the state space, which makes it difficult to attend the real-time constraints.

The work developed by [7] remains as our approach in a certain sense. It also uses Simulated Annealing to explore the state space of the StrarCraft. But in this case to balance the different classes in the game by checking the similarity of plans in each class. In our work Simulated Annealing is used to determine a goal to be achieved that maximizes resources production, given the current state of resources in the game and a time limit for the completion of actions.

[4] developed a linear planner to generate a plan of action given an initial state and a goal for production of resources, which is defined without the use of any explicit criterion. The generated plan is scheduled in order to reduce the makespan. Our work now uses a planner system with scheduling. Our approach deals with dynamic goals for resources production. This is necessary in order to discover a goal that maximizes the strength of the army. To the contrary, [4] works with a goal with fixed and unchangeable resources to be achieved.

Work developed by [2] has the same objective as the one pursued by [4]. These approaches differ in the use of the planning and scheduling algorithms. [2] developed their approach using Partial Order Planning and SLA* for scheduling. [2] also works with a goal with fixed and unchangeable resources to be achieved and can not be adapted to our problem.

The work of [5] presents an approach based on FF algorithm [10] to develop a plan of action that achieves a specific goal in the shortest possible time (minimum makespan). The goals to be used are expert goals, recovered from a database created from analyses of games already played. In this approach, heuristics and efficiency strategies to reduce the search space and achieve better results. In this work the amount of goals and the number of produced resources is limited those goals that are at the database, at some point the game a more appropriate goal cannot be considered by not being at the database or not produce the correct amount of resources. The approach proposed in this paper aims to select and building goals through plans of actions without resource constraints and various possibilities of productions.

The work of [12] and [17] uses an approach with Case Based Reasoning (CBR) and Goal Driven Autonomy (GDA). CBR is used to select expert goals from a database and GDA allows a goal to be discarded during its executionand a new goal is chosen that takes into account the game situation. This approach has similarities with that proposed in this article, especially in relation to planning goals within the game, however this approach also has a limit on the number of goals that are considered, which are present in a database.

We also surveyed some already existing planners that could be applied. Approaches described in [6], [9] and [11] were considered for our problem at hand, but both have a different approach and would have to be modified to adapt to our goal. In short, most of the approaches surveyed have different objectives and the techniques used are not efficient for the domain that we are exploring.

IV. Scheduling Architecture

This architecture has been developed and adapted so that the choice of goal approach could operate with the actions executed in parallel, which increases the ability to maximize resources and get even better results. The results are very similar to those which are obtained by experienced players of RTS games, in particular StarCraft. Experienced players tend to parallelize most actions executed during a match.

As in the sequential architecture, there are two techniques that integrate the SA allowing the maximization of the amount of resources, which are the system planner and scheduled consistency checker. The parameters of the SA and its algorithm and way of operating were not altered when using the architecture with scheduling. Thus, the relevant details to SA will not be discussed here and emphasis will be given to new architecture. The details about the operation of the SA in choice of goal approach are detailed in [15].

A. Planner System

Planner system uses the concept of partial order planning (POP) [13] to generate a plan of action and perform the scheduling task during this process. Its development was supported by need of a planner that can generate a plan of action as input to the SA, enabling the search for solutions in with parallelism of actions. Another need was to find an integrated solution that reduces the time spent during the process of planning and scheduling.

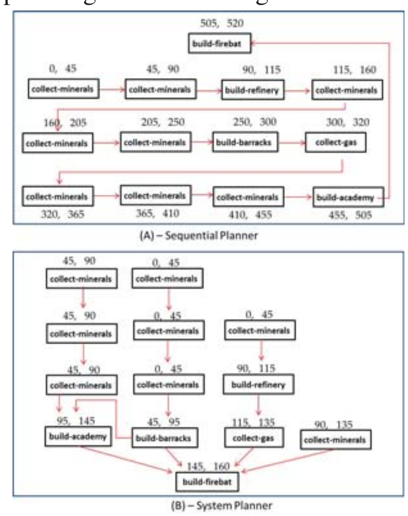


Fig. 2. Comparison between the plans obtained using planner system and sequential planner.

The main motivation for constructing the planner is in the significant gain that the production resources have in relation to the sequential planning, since the choice of a quality goal occurs by maximizing this production. Figure 2 illustrates this advantage in the production of resources. In this figure, the plan (A) illustrates the actions required to build the $Firebat$, where these had execution times assigned and are executed in the linear order indicated by the arrows. This plan takes 520 seconds to perform the actions. In the plane (B), the planner determines the actions necessary to build resources and schedule them. In this plan, the arrows represent the causal links between actions. These indicate the restraining order and resources that will be produced and used between actions. This order specifies only that an action should be executed before another action of which it is a precondition, but how much earlier is indifferent. Thus, the order of execution of actions does not follow a fixed ordering as the sequential planner. Figure 3 shows the execution order of the actions.

The architecture of the planner is composed of two levels. In the first, the processes of the constraint satisfaction problem

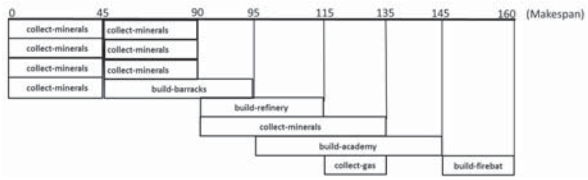


Fig. 3. Order of execution of the plan of action generated in Figure 2.

for the composition of the plan are made, in the second the actions defined by this process are subjected to scheduling. These two levels intercalate its executions and are described in Figure 4. With this architecture, the algorithm has the concept of strong coupling [8] where planning and scheduling problems are reduced to a uniform representation. If any inconsistency is found in the scheduling step it is possible check it out and satisfy it in the planning stage being performed intercalated with that. Thus, it is not necessary to interrupt the process for such verification. In this coupling, the planner is essential for successfully achieving this representation, because the causal links defined between the actions in the planning stage have the temporal order of constraints among the actions, which also assists in the production and use of resources. With that, the scheduling step only finds the best time for the action to be performed, leaving planning tasks where resources information and constraints between actions are checked and set, to be used in the next stage.

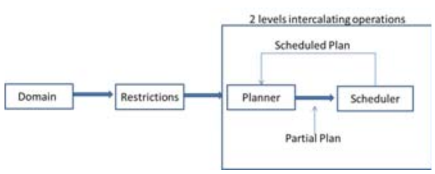


Fig. 4. Representation of the strong coupling intercalating POP and scheduling.

POP allows the coupling in a more intrinsic level to be reached, if used, for example, the sequential planner such representation would be decomposed into two separate sequential problems. Finding a sequential plan of action and since it has been finished would start the scheduling process, similarly to the works of [4] [2]. Algorithm 1 describes the pseudo-code of the planner system.

The planner receives as parameters the list where the actions that will make the plan will be inserted $Plan$, the resource that is the current goal R_{goal} and the time limit for completion of the plan of action T_{limit}. A initial plan of action that achieves an amount of resources in an interval, for example, 180 sec can produce many more resources than the sequential planner. At the beginning of the operation, the planner calls the method $LinkBuild()$ (line 1), responsible for defining all the actions that will make the plan and its respective causal links. This method then sets the links of partial order planning, inserts the first action in the plan from the goal resource and from its predecessor resources fills the plane with the remaining actions necessary.

The algorithm between lines 2 and 5 verifies if the action is not the type of production unit, i.e, whether the action is

Algorithm 1 Planner($Plan$, R_{goal}, T_{limit}))

1: $Plan \leftarrow LinkBuild(Plan, R_{goal})$
2: **for each** Action $Act \in Plan$ **do**
3:     **if** $Act.unity = false$ **then**
4:         $times \leftarrow Quantity(Act)$
5:     **end if**
6:     **while** $i < times$ **do**
7:         **for each** Action $ActRs \in Plan$ **do**
8:             **if** $ActRs = Act.rbase$ **then**
9:                 $Schedule(Plan, ActRs, Act)$
10:             **end if**
11:         **end for**
12:         $contr \leftarrow Constructs(Plan, Act)$
13:         **if** $contr = false$ **then**
14:             $Exit()$
15:         **end if**
16:         $i \leftarrow i + 1$
17:     **end while**
18: **end for**
19: **return** $Plan$

the mineral or gas type, but only resources that are not of type unit. If so, the function $Quantity()$ (line 5) is called to check how many times the action should be executed. For example, if the action *build-minerals* has a link to a *build-academy* that requires 150 $Minerals$, three executions of action are necessary because each execution generates 50 $Minerals$.

All resources of the game have a resource base that is responsible for building the resource within the game. For example, the Scv is base resource of $Barracks$, once it is the Scv that builds the $Barracks$ in the game. Whenever the action corresponding to a resource base is found, the method $Schedule()$ (line 9) is used. It traverses the list that contains actions executed by the resource base and tries to find a time interval between these so that the current action can be executed. The scheduler method uses a strategy to find the shortest execution time between the base resources that are in the plan, and can also execute the action.

After setting the starting time of the action through the scheduling process, the planner will now finish setting its attributes, again using the planning stage. The method $Constructs()$ (line 12) is responsible for this task. This sets the starting and ending execution time of the action, inserting new actions of renewable resources if necessary, modifying some link action if necessary due to the scheduling that can change these links, also confers there is not any threat in these (according to the concept of POP), and checks to see if the time limit has not been exceeded in the plan. The final plan that is built by the planner, contains the actions that reach a random amount of resources, as well as having all actions.

B. Scheduled Consistency Checker

With the search for goals toward plans of actions that have quality in army points and yet scheduled actions, verification and validation of new plans becomes even more important. The scheduled consistency checker developed is used in this step. Algorithm 2 shows its pseudo-code.

The algorithm 2,receives as parameters the current solution of the SA $Plan$, the operation that will be performed in the plan opt, the time limit of the solution will be generated T_{limit} and finally the variable $penalty$ for its value be defined. Initially the method $Invib()$ is called (line 1) responsible for

Algorithm 2 Consistency($Plan$, opt, T_{limit}, $penalty$)

```
1: Invib(ActsInv, opt, penalty)
2: for each Action Act ∈ ActsInv do
3:     if Act.feasi = true then
4:         for each Action Actl ∈ Act.link do
5:             UndoL(Actl, ActsInv, penalty)
6:         end for
7:         for each Action Actl ∈ Act.Clink do
8:             ReleaL(Actl, ActsInv, penalty)
9:         end for
10:    end if
11: end for
12: Rearrange(Plan, ActsInv, opt)
13: while cont = true do
14:     for each Action Act ∈ ActsInv do
15:         feasible ← Gather(Plan, Act, penalty)
16:         if feasible = true then
17:             for each Action ActRs ∈ Plan do
18:                 if ActRs = Act.rBase then
19:                     Schedule(Plan, ActRs, Act)
20:                     cont ← Constructs(Plan, Act)
21:                 end if
22:             end for
23:         end if
24:     end for
25: end while
26: return Plan
```

checking what actions will be unfeasible due to operation. The method places the first actions in the unfeasible action list $ActsInv$, which are necessary for the algorithm to find out which others will also become unfeasible.

The algorithm between lines 2 and 11 finds the other actions that will be unfeasible. For each one that still has the attribute $feasi$ valid (line 3), i.e, every action that has just been added to the list of unfeasible, the algorithm calls the methods responsible for removing its links and finding out which other actions will also be unfeasible. The method $UndoL()$ (line 5) is called to put each resource that receives a link of action that was unfeasible Act within the list of actions unfeasible. This is done as Act which is a precondition of such actions, which will no longer run. For example, if an action *build-refinery* become unfeasible this means that the *Refinery* does not exist in the game anymore, so all actions *collect-gas* receiving a link of this resource will also be also unfeasible.

The method $ReleaL()$ (line 8) is called just after $UndoL()$, it is responsible for releasing the actions that were serving as precondition for the action that became unfeasible, thus eliminating the links from other actions that came to it. This frees the resources used by the action to be used by other actions. After the list of actions unfeasible has been completely filled, the algorithm calls the $Rearrange()$ (line 12). It performs the operation on the plan so that it becomes a new neighbour plan. From the line 12 onwards, the algorithm focuses on what actions can becomes feasible again, configures its attributes and schedules it. The function $Gather$ (line 15) is used with every action unfeasible. It seeks the preconditions of the action that is being checked Act, to see if the available actions can be used to execute it.

When the preconditions of an action are found within the plan, the algorithm schedule its and sets its attributes, as they will become feasible. It made a search for base resources of the action (line 17) to define which among those available would execute the action. When a base resource is found the scheduling algorithm is called $Schedule()$ (line 19). The algorithm is the same as used in the planner. It performs the schedule for each action returns to be feasible.

Finally, the function $Constructs()$ (line 20) is called to set the attributes of the action that returned to be feasible. It sets up the execution times of actions, builds all the links between them and their preconditions, removes the action from the list of unfeasible and checks to see if the time limit has not been reached. If this time is reached, the action does not return to feasible and the verifier stops the process returning the plan with the last action that became feasible as new solution for SA.

V. Experiments and Discussion of Results

The experiments were conducted on a computer intel core i7 1.6 GHz CPU with 4 GB of ram running on a windows operating system. The API Bwapi [3] allows the control of StarCraft units and and information retrieval such as the number of cycles and time of the game. It was used as an interface for the experiments with a human player and for evaluating the SA performance.

The procedures used in the experiments are: $SA(S)$, $Player(A)$ and LT. $SA(S)$ refers to our approach, with SA plus planner system and scheduled consistency checker. $Player(A)$ is a human player experience level in the game. To be classified as an experienced player a 5-year experience with StarCraft was considered. LT is a A.I agent player (bot) called LetaBoot [16], who participated in the 2014 AIIDE StarCraft Competition [1], getting in the top three of the competition.

In all tests, the used approaches tried to achieve the best goal with resource production, i.e, a goal focused on produce resources that allow a player to advance against the enemy bases and overcome it in a confrontation. These goals produce various types of resources that increase the amount of army points. Producing resources with ability to fight enemy leads to the production of other types of resources present in the game, since its are required to enable more powerful resources. In fact, a feature observed in various players when played matches of the game were analysed is the trend to produce offensive resources along the match and only produce other types of resources when these are needed to enable the coming of new resources that will be used to attack and resist to the enemy's army. Thus, the approaches will be compared according the amount of resources that can produce within a time limit.

TABLE I. Results of Experiment 1

Time Limit	**Player(A)**	**SA(S)**	***Makespan* of SA(S)**	**Runtime of SA(S)**
150 sec.	29	**32**	150 sec.	26,1 sec.
300 sec.	140	**141**	299 sec.	48,2 sec.
450 sec.	**206**	182	450 sec.	98,02 sec.
600 sec.	**306**	245	598 sec.	141,1 sec.
750 sec.	**437**	302	748 sec.	197,0 sec.
900 sec.	**601**	400	899 sec.	254,9 sec.
1050 sec.	**781**	511	1047 sec.	301,6 sec.

[1] StarCraft Competition is a competition between Starcraft bots, this competition is an event that is part of AIIDE (AAAI Conference on Artificial Intelligence and Interactive Digital Entertainmen). Available at http://webdocs.cs.ualberta.ca/ cdavid/starcraftaicomp/.

In Experiment I, we compared the performance of experienced player with SA(S). The army points (below each procedure in table) obtained in this experiment are much higher than those obtained in experiments made with the sequential architecture. This stresses the importance of scheduling in this approach. The algorithm was able to overcome the player on three occasions and was overrun by another four. It is interesting to note that the algorithm achieved the best results in the mean time intervals (up to 400 sec). In these results the $SA(S)$ runtime was low enough so that the approach was tested inside the game. This is due to the causal link structure of the POP approach, which helps reducing the search for actions and preconditions during management. However, with the growth of time limit intervals, the runtime of SA(S) increases dramatically. This is due to the rise in the number of actions, which begins in the generation of the initial solution by the planner. When solutions are generated by $SA(S)$, the complexity of validating these increases along with the number of actions that are present.

In order to use our approach under real-time constraints we used a strategy that takes into account the fact the best results have been achieved considering time intervals that range from 150 to 300 seconds. This strategy consists of decomposing a resource production goal to be achieved in time intervals superior to 300 sec into smaller resource production subgoals to be achieved in time intervals in which the algorithm can find the best results. As soon as a subgoal is achieved its plan of actions is carried out. While executing the actions that satisfies a subgoal, the algorithm keeps working in order to figure out a plan of action that satisfies the next subgoal, and so on. For example, suppose a 600 sec time limit for resource production. In this case, the algorithm splits the resource production goal for this time interval into three resource production subgoals considering a 200 sec time limit for each one. Whenever a goal is divided into subgoals, these have time intervals between 150 and 200 seconds.

TABLE II. RESULTS OF EXPERIMENT 2

SA		**LT**	
Runtime	Army Points	Army Points	**Execution**
41,1 sec.	**112**	94	**1**
43,0 sec.	92	**101**	**2**
41,9 sec.	**106**	93	**3**
44,3 sec.	**109**	94	**4**
42,8 sec.	**98**	90	**5**
44,2 sec.	**112**	94	**6**
41,6 sec.	94	**101**	**7**
41,1 sec.	**106**	39	**8**
43,5 sec.	**104**	90	**9**
44,9 sec.	**102**	94	**10**

Experiment 2 compares the resource production of SA and the bot LT. LT bot was chosen because it is a bot with open code and also by the use of the resources of Terran class in Starcraft, the same used by our approach. To accomplish this experiment tests were performed to determine the maximum time that LT was able to produce resources without using their resources on direct attacks within the game. This was considered the time that both approaches would have to perform their resource production. At the end, the quality of both in terms of army points was evaluated. LT also was chosen for having an offensive production resources strategy, similar to that described at the beginning of this section. During each experiments execution the bot concentrated only on achieving goals that increase its ability to confront and be confronted by enemies. The time was set in 270 seconds and ten executions were carried out with this time.

In the table II, the SA overcame LT in eight executions and lost two others in relation to resource production. The LT has achieved 101 army points in three executions when 3 Wraith (Terran resource) was produced. In other runs, the agent showed a pattern of army points that varies between 90 and 94 points. The military resource production of LT follows a pattern where the resource Wraith is the most produced, already SA seeks to produce resources that add up more army points to the goal being determined, without following a pattern of production. The LT strategy has a maximum amount of resources that can be built in a time interval and thus the values of army points achieved by it were often similar. $SA(S)$ does a search for the best combination of resources that maximize production and raise the power of the army produced.

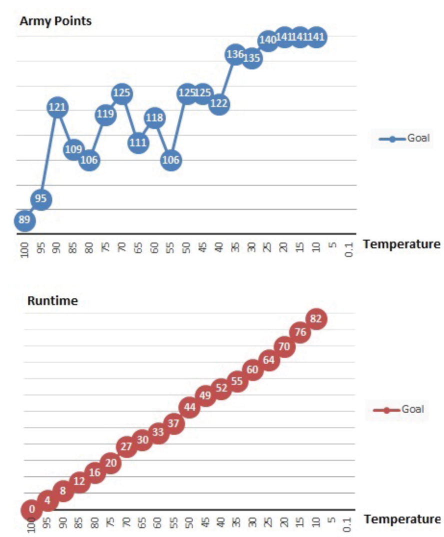


Fig. 5. Graphs of army points and runtime for one of the runs of the experiment I.

In Figure 5 we depict the graphs for the army points and runtime with time interval of 600 sec in Experiment 1. In relation to the army points, $SA(S)$ oscillates between solutions at low temperatures, so that in some cases it even considers plan of action with lower value to the initial plan. This is due to the high amount of changes that occur in the plan when made an operation, even a simple operation. At low temperatures the algorithm follows the normal trend of accepting mostly better solutions than the current. In relation to runtime, the algorithm maintains the tendency to work more quickly when the temperature is in medium/high values. At lower temperatures it spends more time.

The Experiment III, shows the performance analysis of the $SA(S)$. In this, we also made 15 runs of the algorithm for three different time intervals. The algorithm has its performance evaluated in average performances that are close to the optimum value. Good results were obtained with the average intervals. With a limit of 450 sec we obtained solutions almost

TABLE III. RESULTS OF PERFORMANCE TESTS OF SA(S)

Time Limit	200 sec.	450 sec.	750 sec.
Optimum value for army points	55	209	442
Number of runs over 95% of optimum	82%	37%	8%
Number of runs over 90% of optimum	12%	25%	28%
Average value	77	179	301
Average CPU runtime	27,5 sec.	97,3 sec.	199,7 sec.
Number of Runs	15	15	15

exclusively to 90% and 95% of the optimum. With intervals of 750 good results were also achieved. With 1050 sec the algorithm had more difficulty maintaining the results as in the previous experiment, due to the increase number of actions.

VI. CONCLUSION AND FUTURE WORK

In this paper we presented a new approach for improving the maximizing the production of resources in a RTS game. We extend our previous work by developing a system capable of intercalating planning and scheduling of actions. An important component in our system is a consistency checker that enables the generation of new plans with parallel actions.

The objective of this new architecture was to achieve satisfactory results regarding the quality of the solutions, as we achieved in a previous work. With this new approach the results were even better, since the approach increased the maximization of resources while competing with human players without any restriction and maintaining runtime compatible with the game.

As future work, we aim to implement different scheduling algorithms in order to improve the system performance. Also, we intend to integrate our algorithm for resource production with algorithms that implement strategies for attack and defence to build a complete AI player.

ACKNOWLEDGMENTS

This research is supported in part by Research Foundation of the State of Minas Gerais (FAPEMIG) and Faculty of Computing (FACOM) from Federal University of Uberlândia.

REFERENCES

[1] E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimisation and Neural Computing.* John Wiley e Sons, Inc. New York, NY, USA 1989, 1989.

[2] A. Branquinho, C. R. Lopes, and T. F. Naves, "Using search and learning for production of resources in rts games," *The 2011 International Conference on Artificial Intelligence*, 2011.

[3] Bwapi, "Bwapi - an api for interacting with starcraft : Broodwar," available in http://code.google.com/p/bwapi/ - Last visited on 09/04/2014.

[4] H. Chan, A. Fern, S. Ray, N. Wilson, and C. Ventura, "Online planning for resource production in real-time strategy games," in *ICAPS*, 2007.

[5] D. Churchil and M. Buro, "Build order optimization in starcraft," *AIIDE 2011: AI and Interactive Digital Entertainment Conference*, 2011.

[6] M. B. Do and K. S., "Sapa: A scalable multi-objective metric temporal planner," *Journal of AI Research*, vol. 20, pp. 63–75, 2003.

[7] T. Fayard, "Using a planner to balance real time strategy video game," *Workshop on Planning in Games, ICAPS 2007*, 2007.

[8] E. Fink, *Changes of Problem Representation: Theory and Experiments*. Springer, 2003.

[9] A. Gerevini, A.; Saetti and I. Serina, "Planning through stochastic local search and temporal action graphs in lpg," *Journal of Artificial Intelligence Research 20:239-230*, 2003.

[10] J. Hoffmann and B. Nebel, "Ff: The fast-forward planning system," *Journal of Artificial Intelligence Research*, pp. 253–302, 2001.

[11] S. Kecici and S. S. Talay, "Tlplan-c: An extended temporal planner for modeling continuous change," *The International Conference on Automated Planning and Scheduling (ICAPS)*, 2010.

[12] M. Klenk, M. Molineaux, and D. Aha, "Goal-driven autonomy for responding to unexpected events in strategy simulations," *Computational Intelligence*, pp. 187 – 203, 2013.

[13] S. Minton, J. Bresina, and M. Drummond, "Total-order and partial-order planning: A comparative analysis," *Journal of Artificial Intelligence Research*, vol. 2, pp. 227–262, 1994.

[14] T. F. Naves and C. R. Lopes, "Maximization of the resource production in rts games through stochastic search and planning," *IEEE international conference on systems, man and cybernetics - SMC*, pp. 2241– 2246, 2012.

[15] ——, "Stochastic search and planning for maximization of resource production in rts games," *International Conference on Artificial Intelligence - ICAI*, pp. 2241– 2246, 2012.

[16] S. Ontanon, G. Synnaeve, A. Uriarte, F. Richoux, D. Churchill, and M.Preuss, "A survey of real-time strategy game ai research and competition in starcraft," *IEEE Transactions on Computational Intelligence and AI in Games*, 2013.

[17] B. G. Weber, P. Mawhorter, M. Mateas, and A. Jhala, "Reactive planning idioms for multi-scale game ai," *In Proceedings of IEEE CIG*, 2010.

SESSION

PATTERN RECOGNITION AND SUPPORTING ALGORITHMS + MACHINE LEARNING AND APPLICATIONS + LEARNING METHODS (SUPERVISED AND UNSUPERVISED) AND DATA MINING

Chair(s)

TBA

Using Google Glass and Machine Learning to Assist People with Memory Deficiencies

Thomas Way, Adam Bemiller, Raghavender Mysari and Corinne Reimers
Applied Computing Technology Laboratory
Department of Computing Sciences
Villanova University, Villanova PA 19085
thomas.way@villanova.edu

Abstract – *Memory deficiencies may occur naturally with age and for a variety of reasons including Alzheimer's disease, depression, side-effects of drug use, stroke, and traumatic brain injury. Because memory loss can significantly interferes with daily life, many external memory aids have been developed, though most use a passive approach. In this paper, we report on the design and prototype development of a dynamic, wearable system, called ELEPHANT, to assist those with memory deficiencies. Our design uses the Google Glass platform and a machine learning approach to intelligently retrieve stored photographic "memories" annotated with location, date, time, and activity information to enhance the memory of the user. Motivating background, system design philosophy, results of an initial prototype implementation and plans for future enhancements are presented.*

Keywords: Machine learning, Google Glass, memory deficiency, wearable technology, assistive technology.

1 Introduction

Our memories define who we are. Memory deficiencies affect people of any age and are associated with Alzheimer's disease, depression, side-effects of drug use, stroke, and traumatic brain injury, including the resulting side-effects of playing in the NFL, and a variety of other causes [2]. Memory loss significantly interferes with daily life, robbing a person of his or her independence, happiness and sense of self. Over the years, many external memory aids have been developed to assist those with memory loss, from low-tech note cards to a variety of high-tech electronic devices [3,8,10,14,15], though most use a passive approach requiring that they be checked periodically.

Fig. 1. Imagined use of ELEPHANT memory assistive system for elephants with memory deficiencies.

For people with memory loss, not being able to remember to check a notebook or electronic device for information can itself be a barrier to the very assistance that is needed. Recently, research has reinforced past results and demonstrated that using active reminders that automatically notify the user via television assisted prompting [12] and even Google Calendar [13] can successfully help to compensate for memory loss. The successful demonstration of an active approach based on television display or Google Calendar opens up an intriguing line of research and development possibilities, and neatly points at a Google Glass solution.

Rather than requiring manual entry of items into Google Calendar, a system that can automatically prompt entry of reminders could provide a valuable alternative for those with memory deficiency. Our proposed design, the Electronic Localization, Elucidation and PHotographic Assistive Notification Technology system, or ELEPHANT, has the goal of detecting or discovering new events and supporting the prompted, visual learning of an individual's daily routine. Using machine learning techniques, an assistive memory store of annotated visual data is gathered and analyzed for later retrieval and use, providing dynamic reminders and general memory assistance. Annotated photographic information has been demonstrated to provide effective memory assistance to amnesiacs and Alzheimer's patients [7,15,16], with benefits such as improved memory accuracy, the ability to carry on a conversation, and increased quality of social interactions with others.

In this paper, an approach that uses machine learning techniques on the Google Glass platform is presented and discussed. First, motivations from a healthcare needs perspective is shown and the suitability of wearable technology to support those needs is given. Next, justification for our proposed update to the "memory wallet" concept [7] is provided to support the approach and as a foundation for the initial ELEPHANT prototype application. Finally, our plans for continued development and future work are presented.

2 Healthcare Needs

According to a recent survey of adults in the U.S. aged 60 or over, 12.7% of respondent's self-reported increased confusion or memory loss in the preceding 12 months [1]. Of these respondents, 35.2% reported functional difficulties as a result of these issues, resulting in significant negative impact to their normal activities and social interactions.

As of 2011, over 16 million in the U.S. were suffering from some form of cognitive impairment [21], with an estimated 5.1 million of those over the age of 65 suffering from Alzheimer's disease, resulting in more than 10 million family members providing assistive care. By 2014, over 20 million in the U.S. were affected by some similar form of significant cognitive decline [6]

Thus, the challenges and costs to affected individuals, families and caregivers are substantial. However, adopting a technological solution can reduce costs and improve outcomes for those affected [11], and it is long-established that technology can be used to successfully assist people with cognitive impairment [8].

3 Wearable Technology in Healthcare

Wearable technology, wearable devices, or simply "wearables," refer to electronic devices that are incorporated into clothing or accessories worn on the body [19], and are increasing in mainstream popularity [18]. Wearable devices appear to have the potential to improve care for a variety of patients while reducing costs for healthcare organizations [17]. There also appears to be large and underserved market among the chronically ill who could benefit from wearable devices [9], and there is an increasingly widespread belief that wearable devices will become commonplace and perform such helpful tasks as automatic facial recognition to assist with remembering names [17]. Thus, it appears wearable technology has a place in healthcare in general, and in memory assistance specifically.

Fig. 2. Google Glass™. [5]

One device that may serve in that capacity is Google Glass, a type of "smart" glasses. Glass is composed of a frame that holds a computer, battery and a high resolution display [5]. It is equipped with, Bluetooth and Wi-Fi connectivity, a bone conducting transducer for audio I/O, 12 GB of memory, a rechargeable battery, and a camera that can take pictures and videos. Software applications designed to use Glass are called "Glassware," and make use of APIs for the local device and web-based interactions [4]. Some initial trials of using Google Glass in the healthcare industry are underway to assist with patient care, with indications that wearable computing increasingly will have a role in healthcare [17].

4 System Design

The overall design (Fig. 3) of our Google Glass based, machine learning, assistive memory system consists of five principle components: the Recollection Triggers module manages user interactions and queries, a Classification Filter manages the memory database, the Memory Item Database where memories are stored, a Recall Evaluator that assesses query results and guides the user to further actions as needed, and the Prism Display which includes all Google Glass display, I/O and networking support.

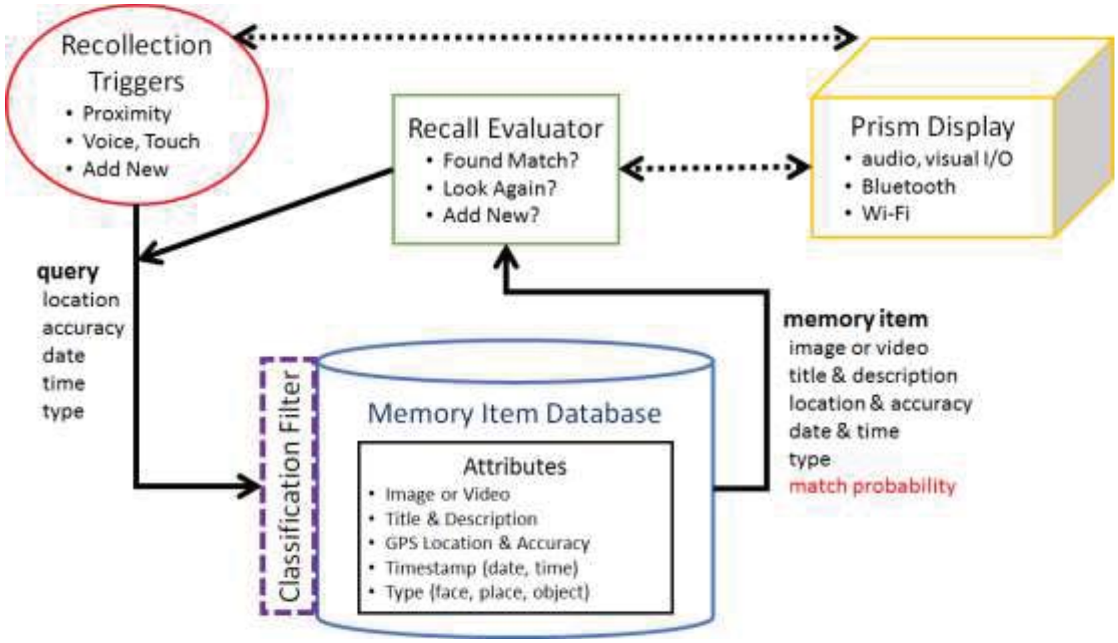


Fig. 3. Diagram of memory assistive, machine learning system for Google Glass.

4.1 Modules

The **Recollection Triggers** are responsible for detecting a noticeable change of location or proximity to a familiar or previously visited location, a direct voice, touch or programmatic command to initiate a query to look-up and add a memory item. The frequency of automatic monitor of GPS location must be balanced with meaningful granularity of location and with battery life as GPS support is known to consume battery power rapidly.

The **Classification Filter** implements multiple machine learning classifiers that are used to perform Memory Item Database look-ups. These classifiers each use one or more of the **Memory Item Attributes** to classify a new query item based on previous memory items. Each classifier reports a probability that a given existing memory item matches the current query so that a most likely memory item match can be found for the current query.

The **Memory Item Database** contains all stored memory items, each of which is an image, textual title and description of that image, the GPS location, GPS measurement accuracy, date and time reported at the time the image was recorded, and an image type as determined using simple image type detection (face, place, object, etc.) or input manually by the user along with the title and description.

The **Recall Evaluator** determines whether the match probability result reported by the Classification Filter and returned with retrieved memory item is a suitable match for the current location. Suitability is determined by the user confirming a correct match. If the match is unsuitable as determined by the user, the Memory Item Database is queried again for the next best match. If no suitable match exists, the user is given the option to add a new memory item to the Memory Item Database.

The **Prism Display** represents the hands-free, Google Glass visual display along with the corresponding hardware that supports audio input and output and both Bluetooth and Wi-Fi network connectivity.

4.2 Automated Learning Extensibility

The opportunity exists to enhance the Recollection Triggers and Recall Evaluator to attempt to identify a title and description for a new memory item and add it automatically to the Memory Item Database. This can be accomplished using the Google Glass Mirror API to perform a Google web query to find a best match for a current location.

ELEPHANT

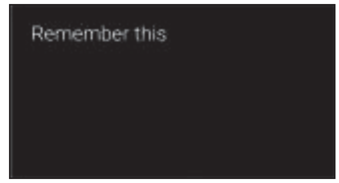
As a proof of concept, a simplified, prototype version of the ELEPHANT system for Google Glass was developed. This prototype application is comprised of two main parts: memory creation and memory display. The memory creation portion guides a user through a series of tasks to capture relevant environmental data about a subject. The memory display portion uses this captured information to create digital, rich media flashcards of the subjects (memories) that are then presented on the high resolution display. Using simple gestures, a user can navigate through these flashcards to review captured information about the subjects. A user can also select any of the flashcards in order to receive additional, more in-depth information about its subject. The hope is that browsing these memories will aid the user with recognition of the subjects captured. Note that the source code is available for this prototype system by contacting the authors, and will be referenced in some detail below for clarity.

4.3 Motivation for Prototype Design

The purpose of developing ELEPHANT was to provide a versatile application that could assist people with a variety of memory deficiencies. While there are many other external memory aids available, Google Glass presents a unique way to easily capture and deliver a wide range of relevant information (visual, audible, and textual) about important subjects in a relatively discreet manner. As a wearable technology, it supplies the means to perform these functions without encumbering users unnecessarily.

4.4 Description

To start ELEPHANT, users can say “ok glass, remember this” from the Glass clock screen, or they can use the touch menu to select the card containing the plus icon and the word “Remember”. Once selected, ElephantService starts. This service owns and displays a LiveCard containing the title “Elephant” and the words “Never Forget,” managed by the RePopulateUIActivity.

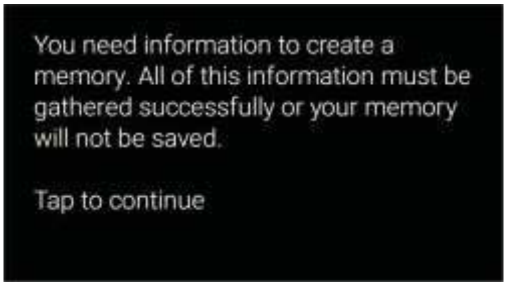


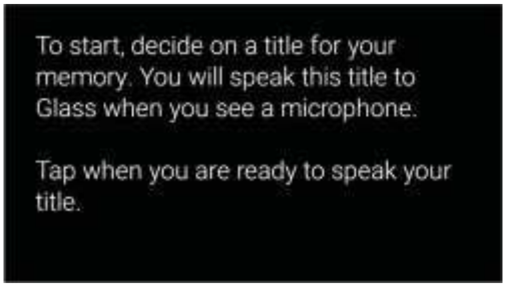
The action provided by this LiveCard is MenuActivity. When tapped, MenuActivity starts and displays a menu with options to “Create Memory”, “Browse Memories”, “Delete Memories”, or “Quit.” Swiping navigates through these options. Tapping chooses an option. Once an option is chosen, the corresponding Runnable is created and run to execute the appropriate set of functionality. This functionality varies with each option.

When “Create Memory” is chosen, date and time information is gathered and stored using a method from the FileOperations class. In addition, a number of Activities are set up for sequential execution. These include Activities for providing instructions on how to gather environmental data about a subject as well as Activities for gathering the data. The sequence of execution is:

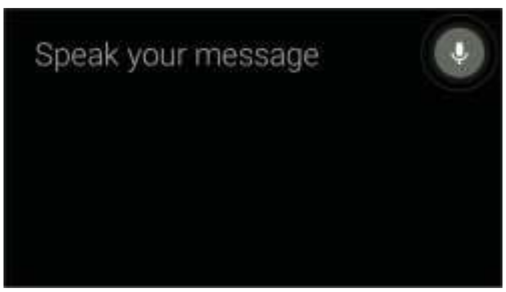
- IntroInstructionsActivity,
- SpeechRecognitionActivity,
- PictureInstructionsActivity,
- PictureActivity,
- VideoInstructionsActivity,
- VideoActivity,
- RePopulateUIActivity, and finally
- FileOperationsWriteActivity.

These instructions are written in a direct, colloquial style. They are designed in such a way that users proceed only when they are ready to do so. They focus on providing only the information necessary to perform the next task in the memory creation, review, or deletion processes. By incorporating these features, users with limited technical experience are enabled to successfully use this application.



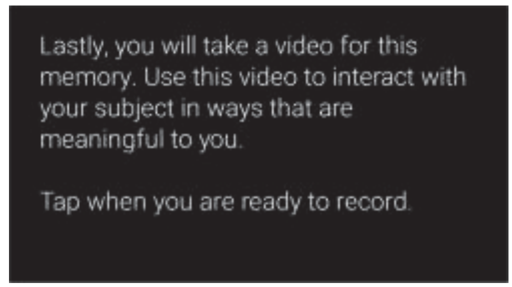


The SpeechRecognitionActivity uses Google's speech recognition technology to create a title for the memory using voice to text.

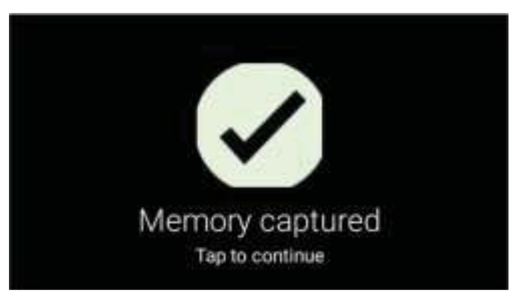


PictureActivity uses Glass's camera to take a photo of the subject.

VideoActivity also uses the camera to capture a "meaningful" interaction with the subject in the form of a video.



The textual title for the memory, and the file paths to the picture and video on the local disk are stored using methods in the FileOperations class. The FileOperationsWriteActivity stores all of this information in the local database so that it can be easily retrieved and accessed as a single "memory" of the subject.



In similar fashion to the "Create Memory" option, three activities are set up for sequential execution when "Browse Memories" is chosen.

In order, these are: MemoryScroll-ActivityInstructions, MemoryScrollActivity, and Re-PopulateUIActivity. The MemoryScrollActivity-Instructions educates users on how to review the stored memories. The RePopulateUIActivity is the same activity that was previously described.

The core functionality within the "Browse Memories" option lies in MemoryScrollActivity. When this activity is started, all information about each of the memories is obtained from the database. If no memories are found, a card is displayed alerting the user that memories should be created.

If memories are found, a Runnable is created on the main UI thread to periodically check on the status of the JpgToBmpTask. The JpgToBmpTask, (an AsyncTask), is then executed in parallel on a background thread to create bitmaps from the jpg images of the subjects that are stored on the disk. (This is necessary so that the UI thread does not become unresponsive, and so that the images can be displayed on cards.) Throughout this process, cards are shown to users to update them on the task's progress.

Once the bitmaps are ready, the Runnable from the main UI thread is stopped. Next, cards are created for each memory and added to a CardScrollView so that users can navigate through them by swiping. Each card contains the title, date, time, and picture for the memory's subject. A video is also associated with each memory. If users tap on a memory's card, Google Glass's video player presents additional information about that memory by playing the appropriate video from the disk. Once the video is finished, users are returned to the CardScrollView to browse additional memories, if desired. Per the MemoryScroll-ActivityInstructions, users swipe down from the CardScrollView to return to the card containing "Elephant" and "Never Forget."

When "Delete Memories" is chosen, the Delete-MemoriesActivity is setup for execution, followed by the RePopulateUIActivity mentioned previously. The DeleteMemoriesActivity presents users with a card warning them that they are about to delete all memories. Per that card's instructions, swiping down will cancel this activity, while tapping will cause deletion to proceed. Swiping down returns users to the card containing "Elephant" and "Never Forget." Tapping causes all information about each of the memories to be deleted from the database by calling a method from FileOperations. It also causes a card to be displayed notifying users that deletion was successful.

When "Quit" is chosen, a request is made to stop ElephantService. This causes ELEPHANT to terminate.

5 Conclusions

There is a definite utility to providing tremor-reduction in software form, as it can apply even as hardware technology and pointing device technology continues rapidly to evolve. A smoother and less jumpy pointer can help eliminate a distraction for an audience during a presentation and provide a presenter who has a tremor of some form with a more comfortable and effective presentation experience. Pointing devices using tremor-reduction software can help those with natural tremors such as Essential Tremor or those with medical conditions that cause other mild to severe tremors. In fact, an approach to tremor filtering may be more widely applicable and useful to a general user, as the tendency to become nervous, and therefore have some mild hand tremor, when making a presentation before an audience is a natural tendency for many speakers.

6 Future Work

Using wearable technologies to aid those with memory deficiencies holds significant promise for improving patient outcomes and reducing the cost of care in the future. Since there is already evidence that some with cognitive impairments prefer electronic memory aids to mnemonic strategies, it seems natural to pursue improving the capabilities of electronic devices [7]. In particular, a sophisticated wearable such as Google Glass may have the potential to provide a wide range of assistive functions that are not possible with other devices currently available.

The ELEPHANT prototype system creates digital, rich media flashcards of subjects (memories) that are then shown to a user on Glass's high resolution display. This is helpful as an external memory aid. However, this application has the potential to be used as a memory training platform as well. One way that this could be accomplished is by incorporating logic into the application that implements the “vanishing cues” learning technique to elicit responses from users about each memory. For some with cognitive impairment, this “vanishing cues” learning technique proved to be an effective way to acquire new information [15].

Delivering more targeted memory reminders has proven to be beneficial to memory enhancement [20]. This targeted approach will be implemented as part of our overall system design that incorporates richer cues that include GPS location, date, time and memory item type, filtering using machine learning classification techniques. Some experimentation will be needed to identify the most appropriate and effective machine learning approaches, although the authors suspect that Logistical Regression, SVMs or Naive Bayes hold the greatest promise.

7 Acknowledgements

This work was sponsored in part by a Google Glass Accessibility Research Program award provided by Google, Inc. We gratefully acknowledge the technical and research guidance provided by Casey Burkhardt and others at Google Mountain View.

8 References

[1] Adams, Mary L., Angela J. Deokar, Lynda A. Anderson, and Valerie J. Edwards. Morbidity and Mortality Weekly Report. Rep. 18th ed. Vol. 62. Atlanta: Centers for Disease Control and Prevention, 2013. Self-Reported Increased Confusion or Memory Loss and Associated Functional Difficulties Among Adults Aged ≥ 60 Years -- 21 States, 2011. Centers for Disease Control and Prevention, 10 May 2013.

[2] David T. Derrer, MD, reviewer. Memory Loss. WebMD article, reviewed Aug. 27, 2013. URL: http://www.webmd.com/brain/memory-loss. Accessed: Jan. 30, 2014.

[3] Giles, G. M. and Shore, M. The effectiveness of an electronic memory aid for a memory impaired adult of normal intelligence. American Journal of Occupational Therapy, 43: 409–411, 1988.

[4] "Principles." Google Developers. Google, 9 July 2014. URL: https://developers.google.com/glass/design/principles.

[5] Google. "Tech Specs." Google Glass Help. Google, 2015. URL: https://support.google.com/glass/answer/3064128

[6] Ha, Kiryong, Zhuo Chen, Wenlu Hu, Wolfgang Richter, Padmanabhan Pillai, and Mahadev Satyanarayanan. "Towards Wearable Cognitive Assistance." Proceedings of the 12th International Conference on Mobile Systems, Applications, and Services (MobiSys 14), Bretton Woods, NH. 2014. pages 68-81.

[7] Haberlandt, Karl. Human Memory: Exploration and Application. Boston: Allyn and Bacon, 1999.

[8] Harris, J. E., and Peter E. Morris. Everyday Memory: Actions and Absent-mindedness. London: Academic, 1984.

[9] Herz, J. C. "Wearables Are Totally Failing the People Who Need Them Most." Editorial. WIRED. Condé Nast, 6 Nov. 2014.

[10] Intons-Peterson, M. J. External memory aids and their relation to memory. In Cognitive psychology applied, Edited by: Izawa, C. 142–166. Mahwah, NJ: Erlbaum. 1993.

[11] Lee, Robin, and Gillian Davies. "Technology: The Cure for Rising Healthcare Costs?" Editorial. MIT Technology Review. Sagentia, 3 Sept. 2013.

[12] Rik Lemoncello, McKay Moore Sohlberg, Steve Fickas, Jason Prideaux. A randomised controlled crossover trial evaluating Television Assisted Prompting (TAP) for adults with acquired brain injury. Neuropsychological Rehabilitation, 21(6), 2011.

[13] A. McDonald, C. Haslam, P. Yates, B. Gurr, G. Leeder, A. Sayers. Google Calendar: A new memory aid to compensate for prospective memory deficits following acquired brain injury. Neuropsychological Rehabilitation, 21(6), 2011.

[14] M. Oriani, E. Moniz-Cook, G. Binetti, G. Zanieri, G. B. Frisoni, C. Geroldi, L. P. De Vreese, O. Zanetti. An electronic memory aid to support prospective memory in patients in the early stages of Alzheimer's disease: A pilot study. Aging & Mental Health, 7(1), 2003.

[15] Parkin, Alan J. Memory and Amnesia: An Introduction. Oxford, UK: Blackwell, 1997.

[16] Parkin, Alan J. Memory, a Guide for Professionals. Chichester, England: J. Wiley, 1999.

[17] PwC’s Health Research Institute. Health Wearables: Early Days. Rep. N.p.: PricewaterhouseCoopers LLP, 2014.

[18] PwC. The Wearable Future. Rep. N.p.: PricewaterhouseCoopers LLP, 2014.

[19] Tehrani, Kiana, and Andrew Michael. "Wearable Technology and Wearable Devices: Everything You Need to Know." Wearable Devices. Wearable Devices Magazine, 26 Mar. 2014.

[20] Turkington, Carol, and Joseph R. Harris. "Rehabilitation of Memory." The Encyclopedia of Memory and Memory Disorders. 2nd ed. New York: Facts on File, 2001.

[21] United States. Centers for Disease Control and Prevention. Cognitive Impairment: A Call for Action, Now! U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, Feb. 2011.

A novel biologically plausible supervised learning method for spiking neurons

Lilin Guo, Zhenzhong Wang, and Malek Adjouadi
Center for Advanced Technology and Education
Department of Electrical and Computer Engineering
College of Engineering and Computing
Florida International University, Miami, Florida, USA

Abstract – *A novel learning rule, Cross-Correlated Delay Shift (CCDS) learning algorithm, is proposed for processing spatiotemporal patterns in this study. CCDS is a supervised learning rule that is able to learn association of arbitrary spike trains in a supervised fashion. Single spiking neuron trained according to CCDS algorithm is capable of learning and precisely reproducing arbitrary target sequences of spikes. Unlike the ReSuMe learning rule, synapse delays and axonal delays in CCDS are variants which are modulated together with weights during learning. Besides biological plausibility, CCDS is also computationally efficient. In the presented experimental analysis, the proposed learning algorithm is evaluated by it properties including its robustness in dealing with noisy environment, and its adaptive learning performance to different spatio-temporal patterns. Simulation results have shown that the proposed CCDS learning method achieves learning accuracy and learning speed improvements comparable to ReSuMe.*

Keywords: Supervised learning, Spiking neuron, Delay Learning, Spike pattern association

1 Introduction

In recent years, supervised learning in a network of spiking neurons has gained increased attention in diverse machine learning applications. One reason for this interest is that learning from instructions or demonstrations is a fundamental property of our brain to acquire new knowledge and develop new skills. Several supervised learning algorithms have been successfully developed for nonlinear benchmark problems. Some of the existing supervised learning rules, such as SpikeProp [1], QProp [2], RProp [2] etc. are using error back propagation similar to the traditional Neural Network (NN). The two major limitations of these methods and their extensions [2]–[4] are that (1) they do not allow multiple spikes in the output spike train, and (2) are sensitive to spike loss, in that no error gradient is defined when the neuron does not fire for any pattern, and hence will never recover. The tempotron learning rule [5], another gradient descent based approach which is efficient for binary temporal classification task, has these two problems as well. As demonstrated in study [6], non-gradient-based methods like evolutionary strategies do not suffer from these tuning issues. An evolutionary strategy is, however, time consuming for large-scale networks. Other temporal learning rules, such as SPAN [7], PSD [8], Chronotron [9], have been developed to train neurons to generate multiple output spikes in response to a spatio-temporal stimulus. In the Chronotron, both analytically-derived (E-learning) and heuristically-defined (I-learning) rules are introduced. Both the E-learning rule and the SPAN rule are based on error function of the difference between the actual output spike train and the desired spike train. Their application is therefore limited to tractable error evaluation, which is unavailable in biological neural networks and is computationally inefficient. The I-learning rule of Chronotron is based on particular case of Spike Response Model, which might have limitations for other spiking neuron models. In addition, it depends on weight initialization. Those synapses with zero initial value will not be updated according to the I-learning rule, which will lead to information loss from afferent neurons.

Well known biologically inspired spike-timing dependent plasticity (STDP) was observed through experiments on hippocampal neurons [10] which directly related the synaptic weight value to the time differential between the pre and post-synaptic firing times. ReSuMe [11] is one of few supervised learning algorithms that based on a learning window concept similar to STDP. Similar to SPAN and PSD, ReSuMe is derived from the Widrow-Hoff rule [12]. It combines STDP and anti-STDP learning window under remote supervision of instruction neuron to produce a desired output spike train in response to a spatiotemporal input spike pattern. With this method, it also can reconstruct the target input/output transformation.

The importance of delays in computing with spiking neurons defining a supervised learning rule acting on the delays of connections (instead of weights) between the reservoir and the readout neurons was well demonstrated in [13]. Axonal conduction delays refer to the time required for an action potential to travel from its initial site near the neuronal soma to the axon terminals, where synapse connect the soma with other neurons. Evidence shows that conduction delay in the mammalian brain can reach from a few ms up to over 50 ms [14]. The effect of delay on the processing ability of the nervous system has been studied in great detail [15], [16]. There is biological evidence that the synaptic delay can be modulated instead of always being invariant [17]. Such evidence supports the introduction of a novel learning algorithm for spiking neurons. Two known approaches for delay learning in SNNs are delay selection [3], [18] and delay shift [19]. In the delay selection method, two neurons are assumed to be connected by multiple synapses with different fixed delays. The weights of connections when related to suitable delays are enhanced while the weights related to unsuitable ones are decreased. Delay shift method adapts the actual delay values of the connections between neurons during training. Input spike patterns close to the synaptic delay vector will make the neuron emit an output spike. Such adaptation may be achieved by changing the length or thickness of dendrites and axons, the extent of myelination of axons, or the density and type of ion channels [20]. However, the weights in study [19] are considered constant during the learning procedure.

In this paper, a novel supervised learning method, called cross-correlated delay shift (CCDS), is proposed to improve ReSuMe by integrating synaptic delay, axonal delay learning with the synaptic weights learning process.

2 Methods

In this new learning method, the synapse delays and axonal delays associated with weights are obtained in the training phase. The neuron model used in this study is described in section 2.1, ReSuMe method is outlined in section 2.2, and details on the CCDS learning rule are given in section 2.3.

2.1 Spiking neuron model

Simple phenomenon models with low computational cost are more popular for studying the dynamics of spiking neural networks as compared to the more detailed conductance-based neuron model. The 1-D leaky integrate-and-fire model is considered in this study, and the dynamic of the i-th neuron is as defined in the following equation:

$$\tau_i \frac{dV_i}{dt} = E - V_i + (I_{syn} + I_{ns}) \cdot R_i \qquad (1)$$

where V_i is the membrane potential, $\tau_i = R_i C_i$ is the time constant of membrane, E defines the resting potential, I_{syn} and I_{ns} are the synaptic current and background noise current, respectively. Note that when membrane voltage V_i reaches the threshold level V_{th}, the neuron emits a spike and V_i is reset to V_{rest} for a refractory period t_{ref}. The synaptic current is thus modeled as

$$I_{syn}(t) = \sum_j w_j I_{PSC}^j(t) \qquad (2)$$

where w_j defines the synaptic efficacy of the j-th afferent neuron, I_{PSC}^j represents the postsynaptic current from afferent spikes. The postsynaptic current with synaptic delay can thus be written as:

$$I_{PSC}^j(t - dt_j) = \sum_{t^f} K(t - t^m - dt_j) H(t - t^m - dt_j) \qquad (3)$$

where t^m and dt_j are the m-th spike and the synaptic delay from the j-th afferent neuron, respectively; $H(t)$ is the Heaviside function; K refers to a normalized exponential kernel function as:

$$K(t) = V_0(\exp(-t/\tau_s) - \exp(-t/\tau_f)) \qquad (4)$$

where V_0 is the normalized factor, τ_s and τ_f are the slow and fast decay time constant, respectively, with $\tau_s / \tau_f = 4$.

2.2 ReSuMe

Supervised learning in temporal encoded SNNs attempts to link the input spike train with output spike sequence. ReSuMe is such a learning method which adjusts the synaptic weights of a neuron to generate a desired spike train $S^d(t)$ in response to a spatio-temporal input spike pattern $S^{in}(t) = [s_1(t), s_2(t), \cdots, s_n(t)]$.

In ReSuMe, synaptic weights are modified according to

$$\frac{d}{dt} w(t) = [S^d(t) - S^o(t)] \left[a + \int_0^\infty W(s) S^{in}(t-s) ds \right] \qquad (5)$$

where $S^d(t)$, $S^{in}(t)$ and $S^o(t)$ are the desired, pre-and post-synaptic spike trains, respectively. The parameter a is a non-Hebbian term. In the case of excitatory synapses, the term a is positive and the learning window $W(s)$ has a shape similar as in STDP. In the case of inhibitory synapses, a is negative and $W(s)$ is defined similarly as for the anti-STDP rule. When the number of spikes in the actual output spike train $S^o(t)$ is more or less than the number of spikes in the desired spike train $S^d(t)$, a decrease/increase is assumed in the weights. This will speed up the convergence of the training process. In ReSuMe, no delay was considered.

2.3 CCDS

Taking into consideration both the synaptic and axonal delays, Fig. 1 illustrates a neuron structure with multi-path connectivity. Each spike from the afferent neuron will result in a post-synaptic current (PSC). The membrane potential of the post-synaptic neuron is a weighted sum of all incoming PSCs from afferent neurons. Fig. 1 shows a multi-connected neuron structure with axonal delays d_i, $i = 1,\ldots,n$ and synapse delays dt_i $i = 1,\dots,n$. Their corresponding weight values are w_i $i = 1,\dots,n$, respectively.

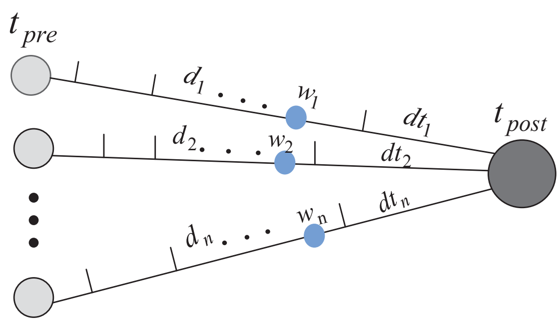


Fig. 1: Neuron structure with multi-path connectivity: axonal delays and synapse delays are from d_1 to d_n and dt_1 to dt_n with corresponding weight values w_1 through w_n , respectively.

The time differential between input and output spike times can be formulated as follows:

$$\delta_{t_i} = t_{post} - (t_{pre} + d_i + dt_i) \, , i = 1, \cdots, n \tag{6}$$

Then, the positive half of the learning window of spike-timing-dependent plasticity (STDP) results in long-term potentiation (LTP) of the synaptic weights as expressed below:

$$\delta w_i = A_1 \exp(-\frac{\delta t_i}{\tau_1}) \tag{7}$$

where A_1 is the maximum value of the weight potentiation, τ_1 is the width of the window for LTP and δ_{t_i} is the time differential as defined by (6).

Similarly, the negative part of the learning window where long-term depression (LTD) occurs is defined as

$$\delta w_i = A_2 \exp(-\frac{\delta t_i}{\tau_2}) \tag{8}$$

where A_2 is the maximum value of weight depression and τ_2 defines the width of the window for LTD.

The weight modulation can be written as

$$w_{i(new)} = w_{i(old)} + \delta w_i \tag{9}$$

Let us first consider a simple example in order to formulate the relative occurrence rule. Assume both data groups d_1 and d_2 has a total of k spikes occurring at various times within a temporal window T . Consider a particular spike time t_s occurring at n different channels (neurons). These are divided into M groups, g_i , $i = 1,\dots,M$, with m channels in each group. t_s occurs p times within group g_1 and q times within group g_2 ,Therefore, occurrence of t_s in g_1 relative to g_2 is

$$O(g_1) = \frac{p}{p+q} \tag{10}$$

Similarly, the relative occurrence of t_s in g_2 can be expressed as

$$O(g_2) = \frac{q}{p+q} \tag{11}$$

Then the weight that reflects the association of t_s with g_1 is modified to

$$w_{ij(new)} = w_{ij(old)} + \frac{p}{p+q} \delta w_{ij}(t_s) \tag{12}$$

where $w_{ij(old)}$ is the pre-trained value associate with connection w_{ij} .

A similar rule that reflects the association of t_s with g_2 is given by

$$w_{ij(new)} = w_{ij(old)} + \frac{q}{p+q} \delta w_{ij}(t_s) \tag{13}$$

Dividing all input spike trains into M groups, each group having m spike trains, the updated weight can be written as

$$w_{ij(new)} = w_{ij(old)} + C_c \cdot \delta w_{ij}(t_s) \tag{14}$$

where the cross correlated term is given by the relation

$$C_c = \frac{\sum_{N_i^f=0}^{N_i^f=k} O(g_i)}{\sum_{N_i^f=0}^{N_i^f=k} \sum_{j=1}^{m} O(g_j)} \tag{15}$$

The proposed CCDS algorithm is a heuristic method which helps the neuron generate a desired output with the ability of removing undesired output instances. In CCDS, the delay is applied to the connection that has the nearest spike before the desired time, which leads to an increase in post-synaptic potential (PSP) at the desired time. In addition, the reduction of the PSP for undesired output spike is achieved by delayed PSP. The reduction may eventually cancel undesired spikes.

The nearest previous input spike is calculated via local variable, $x_i(t)$, described in (16) below:

$$x_i(t) = \begin{cases} A_o \exp(-(t - t_i^f)/\tau), & for\, t_i^f < t < t_i^{f+1} \\ A_o & for\, t = \cdots t_i^f \, , \; t_i^{f+1}, \cdots \end{cases} \tag{16}$$

where amplitude A_o and time decay τ are constants. $x_i(t)$ in this case jumps to a saturated value A_o whenever a presynaptic spike arrives.

If the previous spike is far from the current time t , then $x_i(t)$ is low, otherwise if it is close to t , then $x_i(t)$ is high. The delays d_i and dt_i shift the effect of its spike to time t by using the inverse operation of (16) as expressed below:

$$d_i + dt_i = t - t_o^f = -\tau \ln\left(\frac{x_o(t)}{A_o}\right) \quad (17)$$

At desired spiking time without any actual output spikes, $x_o(t)$ is chosen from excitatory synapses that are not delayed previously. The chosen connection is delayed by $d_i + dt_i$. Then the spike is shifted toward the desired time, which will lead into an increment in the PSP. In contract, at the undesired spiking time with output spikes, $x_o(t)$ is chosen from inhibitory synapses that are not delayed previously.

Considering both the cross-correlation and delay shift effect, the weights as governed by the CCDS learning rule are updated on the basis of (18).

$$\frac{dw_i(t)}{dt} = C_c[s_d(t) - s_o(t)] \cdot [a + \int_0^{+\infty} W(s - d_i - dt_i) S_i(t - d_i - dt_i - s) ds] \quad (18)$$

where C_c is the same as in (12), with the learning window being

$$W(s) = \begin{cases} Ae^{-\frac{s}{\tau}}, s \geq 0 \\ 0, s < 0 \end{cases} \quad (19)$$

3 Results

3.1 Experimental setup

The trained neuron is connected with n afferent neurons, and each fires a spike train in the time interval $(0, T)$. Input spike trains are desired spike train and are randomly generated with a homogeneous Poisson distribution with mean frequency F_{in} and F_d, respectively. The ratio of inhibitory and excitatory synapses is set to the standard ratio of 1/4 as cortical neuron [21]. The initial synaptic weights are drawn randomly from uniform distribution with mean value of -0.5 and a standard deviation of 0.2 for inhibitory synapse, and with mean value of 0.75 and a standard deviation of 0.2 for excitatory one. For the learning parameters, we set the membrane time constant $\tau_i = 10ms$; the refractory period $t_{ref} = 5ms$; the initial voltage, the threshold voltage and the reset voltage are selected as $V_{init} = -60mV$, $V_{th} = -55mV$ and $V_{reset} = -65mV$, group number M=20, number each channels m=30, respectively. The weights are capped in the range [-15, 15] to ensure convergence. At the beginning of the training phase, 20% of the weights are considered inhibitory while 80% of the weights are considered excitatory. In each epoch, synaptic delay and axonal delay are assumed to be adjusted only once. In contrast, the connection weight can be changed many times during the learning phase.

As axonal delays and synapses delays are limited in the biological neurons, all axonal delays and synaptic delays in this method evolve within the interval [0, 40]ms and [0, 2]ms, respectively.

3.2 Learning process

The correlated-based metric(C) [22] is used to evaluate the similarity of the desired spike pattern with the actual output spike train. It takes values between zero and one. The metric C equals one for identical spikes and drops to zero for loosely correlated trains.

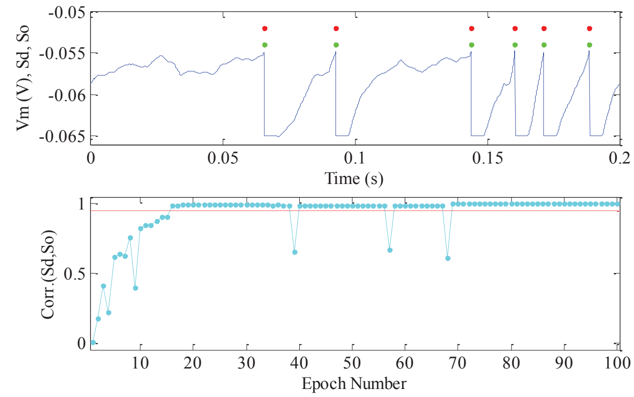
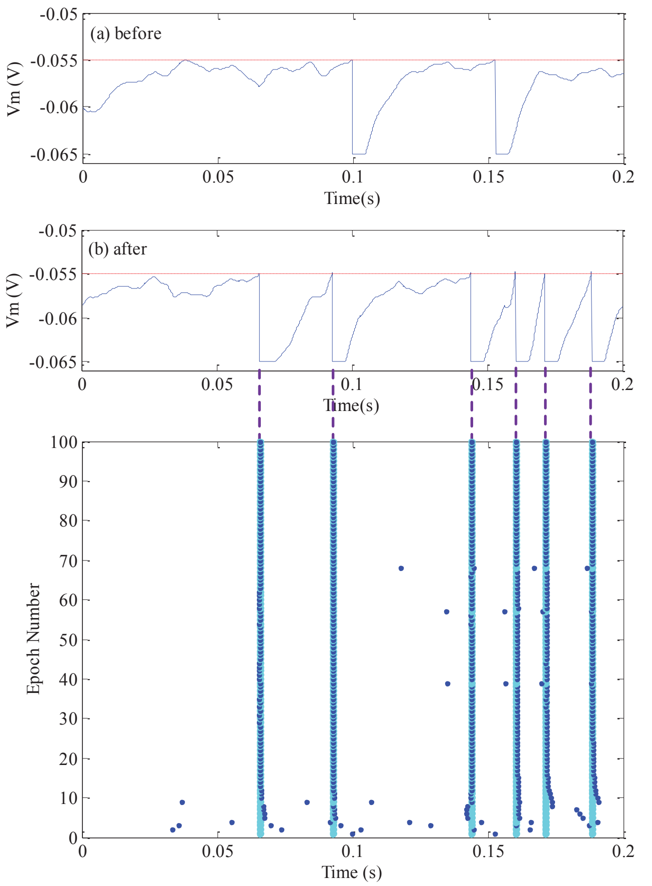


Fig. 2: Training results without noise. (a) V_m: membrane potential after learning; red dots: target spike train; green dots: actual output spike train; (b) correlated-based metric C of target and output spike trains.

Input spike trains are generated by a homogenous Poisson spike train with frequency $F_i = 10Hz$ with n afferent neurons (n=600). Frequency $F_d = 40Hz$ is chosen to produce the output spike train. Delayed version LIF is utilized for the training. In Fig. 2(a), the red dots are the target spikes while the green dots are the actual spikes. In Fig. 2(b), at around 16 epochs, the correlation C of desired and observed output spike trains reach a satisfactory level C>0.95. After a small period oscillating, the correlation C converges towards 1. The evolution of firing patterns generated by the neuron in consecutive learning epochs can be seen in Fig. 3(c), where the cyan line is the desired spike and the blue dots are the actual output spike patterns according to the learning epochs. Fig. 3(a) and Fig. 3(b) present the membrane voltage of learned neuron before learning and after learning, respectively.

The results show that the neuron can successfully learn to emit the desired spike train from the initial random output spike train after just 69 learning epochs. The six randomly generated spike patterns converge perfectly after training.

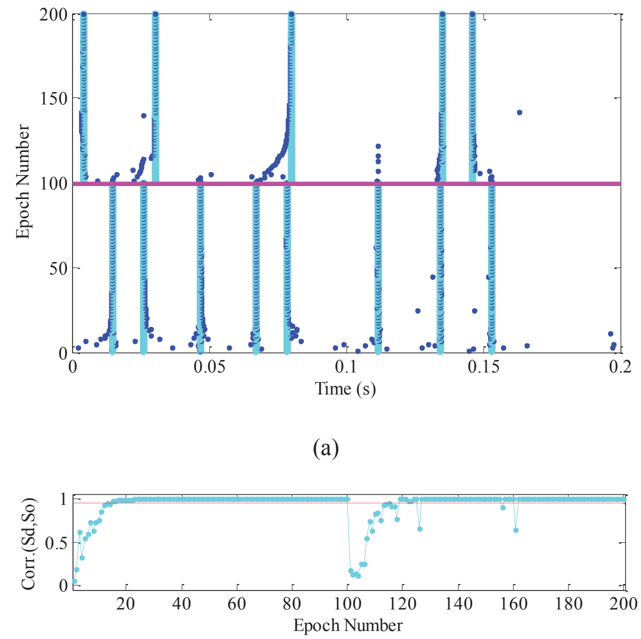


(c)

Fig. 3: Temporal sequence learning of a typical run without noise (a) membrane potential before learning; (b) membrane potential after learning; (c) learning process.

3.3 Adaptive learning performance

At the beginning, the neuron is trained to learn a target train as in the previous experiments. After successfully learning the process, the target spike train is changed to an arbitrarily generated train, where the precise spike time and firing rate may be different from the previous target train. We found that, we could successfully train the neuron to learn the new target within several epochs with the CCDS learning rule. As shown in Fig. 4(a), each dot denotes a spike. At the beginning, the neuron is trained to learn one target (denoted by cyan bar in the bottom part). After 100 epochs of learning (the dashed blue line), the target is changed to another randomly generated train (denoted by the cyan bar in the top part). Again, the neuron successfully learned the new target spike train within 60 epochs. Fig. 4(b) shows the correlated measure C of different desired spike train and output spike train along the learning process.



(b)

Fig. 4: Adaptive learning of different target trains (a) sequence learning with the changed target train; (b) correlated-based metric C of target and output spike trains.

3.4 Robustness to noise

In the previous experiments, the simple case where the neuron is trained to learn a single pattern without noise is assumed. However, in practical settings, the reliability of the results could be significantly affected by the presence of noise. ReSuMe is shown to be robust to noise during the learning process [11]. Here, we re-evaluate the robustness of the proposed CCDS learning rule.

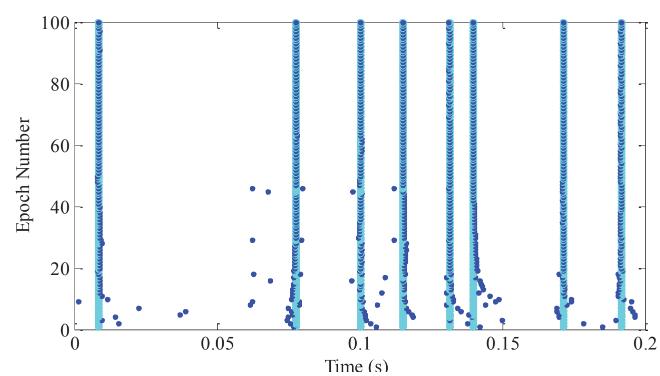


Fig. 5: Temporal sequence learning of a typical run with noise

In this experiment, a LIF neuron with n=600 afferent neurons under background current noise is tested. Gaussian noise is added to the LIF neuron where $I_{ns} = 0.2$ nA. Randomly generated Poisson spike trains are used for both the input and desired spike trains. As shown in Fig. 5, eight spike patterns still converge within 50 epochs. Even when more patterns are considered in the presence of noise, the results still converged within 50 epochs as illustrated in Fig. 5 and Fig. 6.

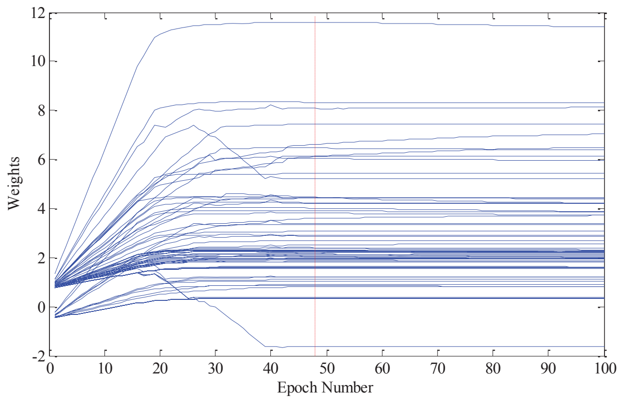


Fig. 6: Synaptic weights during CCDS supervised learning with noise I_{ns}=0.2nA

3.5 Comparison with ReSuMe

In the following experiments, each spike train has a total time duration of $T = 400ms$. At the beginning of CCDS simulation, none of the input spike trains have delays. The same input spike trains with $F_{in} = 5Hz$ and desired spike train $F_d = 100Hz$ are selected for both CCDS and ReSuMe. The performance of the proposed method is compared with that of ReSuMe in Fig. 7. One can note that CCDS learning rule achieves high learning accuracy much faster than ReSuMe. The evolution of weights for each method is given in Fig. 8(a) and Fig. 8(b), respectively. The CCDS method managed to reach the satisfied level C>0.95 much earlier at the 8th epoch and settles on a stable set of weights thereafter. In contract, the ReSuMe training shows the weights continue to adapt even after the 100th epoch.

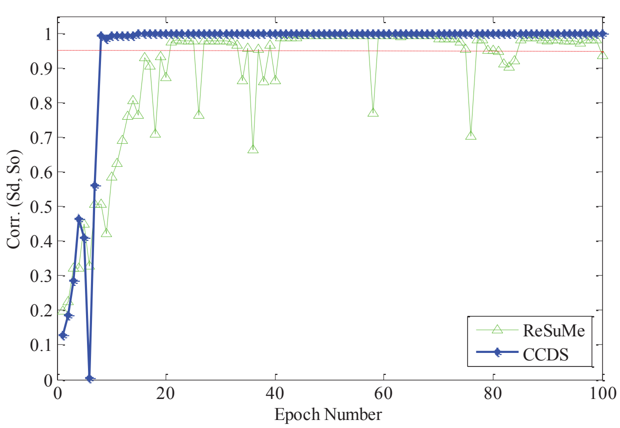
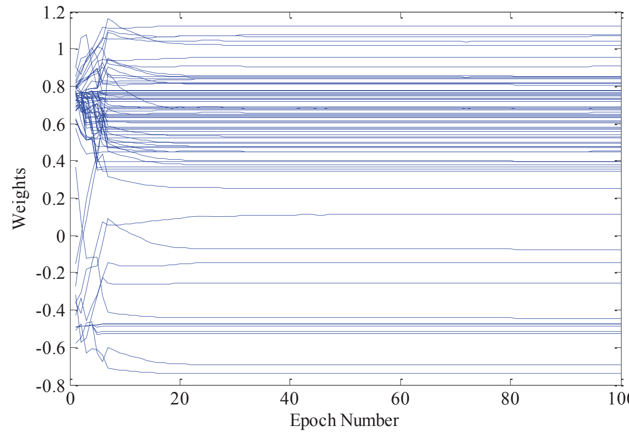
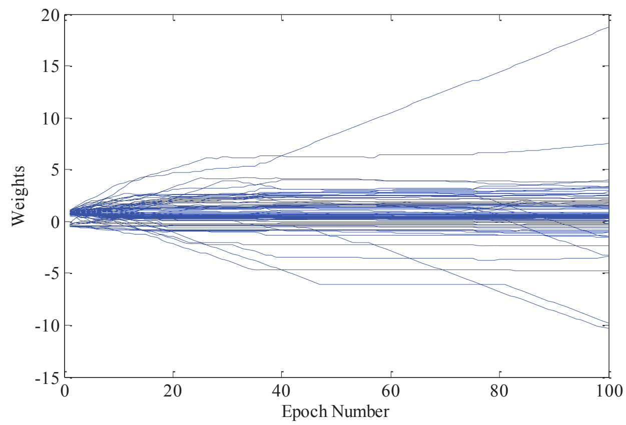


Fig. 7: Evolution of correlated-based metric C for ReSuMe and CCDS



(a)



(b)

Fig 8: Evolution of the weights during learning using (a) CCDS rule; (b) ReSuMe rule

4 Conclusions

In this study, the spatio-temporal associations of key events or patterns were investigated using the proposed CCDS training algorithm. By making use of the biological concepts of spike-timing dependent plasticity (STDP), axonal delays, and synapse delays, the CCDS is able to learn the association between precise test patterns. The results obtained confirm that the proposed method is highly effective and computationally efficient in the spatio-temporal association of arbitrary spike trains in a supervised fashion. Future work will focus on application of proposed learning method to real-world problems such as detecting interictal spikes in electroencephalography (EEG) data and extend the single neuron learning algorithm to network level to do the spatio-temporal pattern classification.

5 Acknowledgements

This research is supported through NSF grants CNS-0959985, CNS-1042341, HRD-0833093, IIP 1338922 and

IIP-1230661. The support of the Ware Foundation is greatly appreciated.

6 References

[1] Sander M. Bohte, Joost N. Kok, and H. La Poutre, "Error-backpropagation in Temporally Encoded Networks of Spiking Neurons," *Neurocomputing*, vol. 48, pp. 17–37, 2002.

[2] S. McKennoch, Dingding Liu, and L. G. Bushnell, "Fast Modifications of the SpikeProp Algorithm," *IJCNN '06. Int. Jt. Conf. Neural Networks, 2006.*, pp. 3970–3977, 2006.

[3] S. Ghosh-Dastidar and H. Adeli, "Improved Spiking Neural Networks for EEG Classification and Epilepsy and Seizure Detection," *Integr. Comput. Aided. Eng.*, vol. 14, no. 3, pp. 187–212, 2007.

[4] S. B. Shrestha and Q. Song, "Adaptive learning rate of SpikeProp based on weight convergence analysis," *Neural Networks*, vol. 63, pp. 185–198, Dec. 2015.

[5] R. Gutig and H. Sompolinsky, "The Tempotron: a Neuron that Learns Spike Timing-Based Decisions," *Nat. Neurosicence*, vol. 9, pp. 420–428, 2006.

[6] A. Beltreche, L. P. Maguire, M. Mcginnity, and Q. Wu, "Evolutionary Design of Spiking Neural Networks," *New Math. Nat. Comput.*, vol. 02, no. 03, pp. 237–253, Nov. 2006.

[7] A. Mohemmed, S. Schliebs, S. Matsuda, and N. Kasabov, "SPAN: Spike Pattern Association Neuron for Learning Spatio-Temporal Spike Patterns," *Int. J. Neural Syst.*, vol. 22, no. 4, p. -1, 2012.

[8] Q. Yu, H. Tang, K. C. Tan, and H. Li, "Precise-Spike-Driven Synaptic Plasticity: Learning Hetero-Association of Spatiotemporal Spike Patterns," *PLoS One*, vol. 8, no. 11, p. e78318, 2013.

[9] R. V Florian, "The Chronotron: A Neuron That Learns to Fire Temporally Precise Spike Patterns," *PLoS One*, vol. 7, no. 8, pp. 1–27, 2012.

[10] R. P. N. Rao and T. J. Sejnowski, "Spike-Timing-Dependent Hebbian Plasticity as Temporal Difference Learning," *Neural Comput.*, vol. 13, no. 10, pp. 2221–2237, 2001.

[11] F. Ponulak and A. Kasinski, "Supervised Learning in Spiking Neural Networks with ReSuMe: Sequence Learning, Classification, and Spike Shifting," *Neural Comput.*, vol. 22, pp. 467–510, 2010.

[12] B. Widrow and M. Hoff, "Adaptive switching circuits.," *1960 IRE WESCON Conv. Rec.*, no. 4, pp. 96 – 104, 1960.

[13] A. G. Ioana Sporea, "Supervised Learning in Multilayer Spiking Neural Networks," *Neural Evol. Comput.*, vol. 25, no. 2, pp. 473–509, 2013.

[14] H. A. Swadlow, "Physiological properties of individual cerebral axons studied in vivo for as long as one year," *J. Neurophysiol.*, pp. 1346–1362, 1985.

[15] B. Glackin, J. A. Wall, T. M. McGinnity, L. P. Maguire, and L. J. McDaid, "A spiking neural network model of the medial superior olive using spike timing dependent plasticity for sound localization.," *Front. Comput. Neurosci.*, vol. 4, 2010.

[16] M. Gilson, M. Bürck, A. N. Burkitt, and J. L. van Hemmen, "Frequency Selectivity Emerging from Spike-Timing-Dependent Plasticity," *Neural Comput.*, vol. 24, no. 9, pp. 2251–2279, 2012.

[17] J. W. Lin and D. S. Faber, "Modulation of synaptic delay during synaptic plasticity," *Trends Neurosci.*, vol. 25, no. 9, pp. 449–455, 2002.

[18] S. Ghosh-Dastidar and H. Adeli, "A New Supervised Learning Algorithm for Multiple Spiking Neural Networks with Application in Epilepsy and Seizure Detection," *Neural Networks*, vol. 22, no. 10, pp. 1419–1431, 2009.

[19] P. Adibi, M. R. Meybodi, and R. Safabakhsh, "Unsupervised learning of synaptic delays based on learning automata in an RBF-like network of spiking neurons for data clustering," *Neurocomputing*, vol. 64, no. 1–4 SPEC. ISS., pp. 335–357, 2005.

[20] R. Wang, G. Cohen, K. M. Stiefel, T. J. Hamilton, J. Tapson, and A. van Schaik, "An FPGA Implementation of a Polychronous Spiking Neural Network with Delay Adaptation," *Front. Neurosci.*, vol. 7, no. February, pp. 1–14, 2013.

[21] J. Mishra, J. M. Fellous, and T. J. Sejnowski, "Selective attention through phase relationship of excitatory and inhibitory input synchrony in a model cortical neuron," *Neural Networks*, vol. 19, no. 9, pp. 1329–1346, 2006.

[22] S. Schreiber, J. M. Fellous, D. Whitmer, P. Tiesinga, and T. J. Sejnowski, "A new correlation-based measure of spike timing reliability," *Neurocomputing*, vol. 52–54, pp. 925–931, 2003.

Definition and mining of quasi-cyclical patterns in agroclimatic data

Geise K. S. Santos, Tércio A. Santos Filho, Marcos A. Batista and Sérgio F. da Silva
geise.kss@gmail.com, tercioas@gmail.com, marcos.batista@pq.cnpq.br, sergio@ufg.br
IBiotec, Federal University of Goiás, Catalão, GO, Brasil
Av. Dr. Lamartine Pinto de Avelar, 1120, CEP 75704-020

Abstract— *Mining patterns in the association rules form are important to extract knowledge in multidimensional datasets. Several patterns occur in an approximately cyclical (quasi-cyclical) form in the nature and the variations of these cycles hinders the application of the traditional pattern mining techniques. The literature methods identify only common cycles. In this work, we presents the formal definition of quasi-cyclical pattern concept and we develop a method for cyclical and quasi-cyclical patterns mining. We conducted experiments using quantitative temporal databases to demonstrate that our temporal rules mining technique achieve satisfactory results.*

Keywords: Cyclical patterns, quasi-cyclical patterns, association rules

1. Introduction

Currently, there are several techniques that discover association rules among items set according to the past transactions. Association rules describe important relationships among items or variables in databases. For example, the purchase history mining from a supermarket can extract a rule that all people who buy meat also buy beer, and then we have the rule "meat $\rightarrow$ beer".

The extracted rules occur with a certain frequency, this is a cyclic variation over time. About the example cited before, the rule "meat $\rightarrow$ beer" can be observed, for example, on Saturdays during the interval 2pm-4pm. Then the people who buy meat on Saturday in the period of 2pm to 4pm hours also buy beer. There are also rules of associations that occur cyclically in nature, as the summer and winter seasons in a tropical region. These rules having cyclical occurrence are denominated cyclical patterns.

Nowadays, several methods of mining temporal data have been proposed [1], [2], [3], [4], [5], [6], [7], [8], [11], [12], [13], [14]. However, these methods are not applicable to mine generalized cyclical rules from quantitative temporal data. There are cyclic pattern definitions in the literature, as related in [9], but the proposed methods identify only exact cycles and they do not detect events that happen approximately cyclically (quasi-cyclical), like: the dry season and the rainy season, planting and harvest cycles of agricultural crops, and others. In these cases, the application of the traditional pattern mining techniques is more difficult because the variations (start, end, duration, and intensity) of the quasi-cycles.

In this work, we present the formal definition of the quasi-cyclical pattern concept and we develop an approach to detect these patterns from quantitative temporal databases. This mining process of quasi-cyclical patterns in the form of rules can be summarized as following: we use a genetic algorithm (GA) to extract rules from quantitative temporal databases; The rules encoded by GA chromosome are checked in the database to build a binary sequence according to their occurrence, and non-occurrence in each period; We calculate the ability of a chromosome based on how well these binary sequences fit in a cycle.

This work is divided into the following sections: in the Section 2, we present the definitions of cyclical and quasi-cyclical patterns. In the Section 3, we describe the used method and the designed algorithm to detect the quasi-cyclical and cyclical patterns. Then we present results obtained from synthetic binary sequences and real data in the Section 4. Finally, in the Section 5, we discuss about the work, its contributions and future work.

2. Definitions

Let $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_m\}$ the set of observed variables. Episode is the value of a given variable $\mathbf{v}_i$, observed at a given instant of time. Let $E = \{e_1, e_2, \ldots, e_m\}$ a set of episodes recorded every time instant t, where e_i is an episode associated with the variable $\mathbf{v}_i$. The set E, at a given instant of time t, is called *super-episodesets* because they are recorded values of all m variables, at a given instant t.

A given interval condition is used to analyze if a variable value occurs at a given interval. An episodic condition is a condition for an interval variable at a given time interval. An association rule is an implication of the form $\mathbf{X} \Rightarrow \mathbf{Y}$ (if $\mathbf{X}$ then $\mathbf{Y}$), where $\mathbf{X}$ and $\mathbf{Y}$ are conjunctions of episodic conditions associated to the observation variables.

A primary cycle or period l_{cb} is a contiguous sequence (adjacent) of observed *super-episodesets*. The base cycle length (or duration) l_{cb} is given by the number of *super-episodesets* of the base cycle, which is predetermined. Mathematically, each base cycle or period corresponds to the time interval specified by $[t_{i \cdot l_{cb}}, t_{(i+1) \cdot l_{cb}})$. The base cycles number n_{cb} is calculated as $\lfloor \frac{|\mathcal{D}|}{l_{cb}} \rfloor$, where $|\mathcal{D}|$ is the *super-episodesets* number in the database.

According Ozden et al. [9], cyclical patterns are rules that occur in regular intervals from a given offset, for example, weekly, monthly, yearly, and others. Let c an occurrence cycle of the rule R, then:

$$c = (l, o),$$

where l is the cycle length, obtained by the number of base cycles or the period between an occurrence and another; and o is the offset, which corresponds to the first instant that the rule R occurs, such that $0 \leq o \leq l$.

The pattern of an association rule occurrence in the base cycles or periods can be represented by a binary sequence, which each value caractherize a period. Thus a value 1 at a given period means that the rule occurred at this period and a value 0 represents no occurrence of the rule. For example, given the rule cited in the section 1 "meat $\rightarrow$ beer" and the binary sequence 010110011010 to represent this rule occurrence, then the rule has the cycle $c = (3, 1)$. It means that this rule occurrence starts in the second period (p_1) and it occurs every 3 times.

A quasi-cyclic pattern is the occurrence of certain rule R in periods not totally accurate. A quasi-cyclic has a frequency component or occurrence probability, which determines rate repetition of the the rule R in the periods. Given a quasi-cycle qc, then:

$$qc = \{(l, o), freq\}$$

where l is the cycle length (calculated by the number of base cycles or periods), which the rule occurrence is checked and the o is the offset, i.e., the first period that R occurs, given $o \geq 0$. We can have $o > l$, when the quasi-cycle does not occur in the first period. If $o < l$, then the rule occurs in the first period. The probability of occurrence of the rule R, $0 \leq freq \leq 1$, is given by:

$$freq = \frac{nPeriods}{n_p},$$

where $nPeriods$ is the periods number that R occurs from o, and n_p is the cycles number from o.

Using again the binary sequence 010110011010 as example, it has, for instance, the quasi-cycle $qc = \{(4, 0), 0.667\}$ with a frequency of about 66.7%. That is, there are 3 periods of length 4 and the quasi-cycle appears in 66.7% of these periods. Remember that each bit of the binary sequence corresponds to the occurrence or not of the rule in a base cycle.

3. Methodology

To detect quasi-cyclical patterns of rules occurrence, first, we developed a mining technique of quantitative temporal association rules based on the proposition of a specific genetic algorithm to this task. The technique is described in terms of the steps and operators of the genetic algorithm and it is given by Algorithm 1.

Algorithm 1: Genetic Algorithm (GA)

Input: Coding of chromosome, fitness function, constraints (attributes of antecedent of rules and attributes of consequent of rules), maximum length of interval ($piv.(\max \mathbf{v}_i - \min \mathbf{v}_i)$), value minimum of time window ($janTime$).

Output: Quantitative temporal association rules.

1 Generate, randomly, a population of chromosomes ($\mathcal{C}$);
2 Evaluate each chromosome $\mathcal{C}_i$ of the population based in the fitness function;
3 Apply the method of *niching*;
4 Select the chromosomes by the roulette method until complete the set of parents (*matting pool*);
5 Apply the uniform *crossover* taking pairs of individuals in the set of parents;
6 Apply the uniform mutation in the newly generated chromosomes;
7 Select the best chromosomes among parents and children to the next generation;
8 While the maximum number of generations is not reached, return to step 2;
9 Return the set of association rules encoded by the population of chromosomes;

w	AC	v_0	v_1	t_0	t_1

Fig. 1: Gene representation.

3.1 Coding of chromosome

In the chromosome coding, each database variable is associated with a gene. If we consider m variables of observation, then we will have m genes: G_1, G_2, G_3, ..., G_m. Each chromosome gene G_i represents a episode associated to a variable $\mathbf{v}_i, i = 1 \ldots m$, and it is encoded as shown in the Figure 1.

In the Figure 1, w is a weight that is compared to a threshold to indicate if the episodic condition represented by the gene will be part or not of the rule. *AC* is a *flag* to indicate whether the episodic condition represented by gene will be part of the antecedent (*AC*=0) or consequent (*AC*=1) of the rule. v_0 and v_1 are the lower and upper limits of the variable range, respectively. t_0 and t_1 are the lower and upper limits of the time interval, respectively.

3.2 Fitness measure

The chromosome fitness measure is obtained by the rule occurrence frequency of this chromosome. If the rules occurrence corresponds to an exact cycle, then the frequency is 1 and consequently the chromosome fitness is also 1. But, if the rule occurrence pattern is only quasi-cyclic, so the fitness is the quasi-cycle frequency, calculated as:

$$freq = \frac{nPeriods}{n_p},$$

where $nPeriods$ is the number of periods that R occurs from o, and n_p is the cycles number from o, given that $freq \in \mathbb{R}$ and $0 \leq freq \leq 1$. If we identified more than one quasi-cycle, the fitness value is given as the frequency of the most frequent quasi-cycle.

3.3 Genetic operators

We did the selection for reproduction using the roulette method. The pairs of individuals selected for reproduction are crossed by uniform crossover: select by random, a mask with the chromosome length, which indicates what parent chromosome will provide each gene to the first child; the second child is generated by the mask complement.

Each chromosome selected for mutation will have one mutated gene. The mutation can occurs in the weight w, in *AC* (when restrictions were not applied on the variables that composes the antecedent and consequent), or in the lower limits (v_0 and t_0) and higher (v_1 and T_1) of the variables intervals and of time.

3.4 Niching

We use a niching method , called clearing, described in Petrowski [10]. We propose a distance measure for quantitative temporal association rules in this method. The following, we describe the clearing niching method and the proposal distance measure.

The clearing method corresponds to the niching concept enunciated by JH Holland in 1975: the resources sharing by individuals population characterized by some similarity. However, instead of sharing the available resources, the clearing method provides a niche features only for the best individual of each subpopulation. This allows that the genetic algorithm performs a multimodal optimization (find the optimal and optimum at the same time). Furthermore, the clearing method of the enables that GA reduces the genetic drift problem, when it is used together with an appropriate selection operator.

The clearing is applied between the chromosomes fitness evaluation and the selection operator application for the crossing. The method uses a distance measure (dissimilarity) among the chromosomes (the phenotype, in our case, means association rules) to determine if they belong to the same subpopulation or not. Each subpopulation will have a dominant chromosome: witch has the highest fitness value in the subpopulation. If a chromosome belongs to a subset, then their dissimilarity related to the dominant chromosome is smaller than a given threshold σ, called clearing radius. The clearing method preserves the the dominant chromosome ability while decreases to zero the other chromosomes ability. Thus, the clearing assigns all the resources of a niche for a single chromosome: the winner. This method corresponds to remove of the population, in an imaginary way, all dominated individuals.

The clearing method also is generalizable to accept multiple winners chosen from the best individuals of the niche [10]. The niche ability is defined as the maximum number of chromosomes that a niche can have. If the capacity is greater than 1, then the population has more than one winner. If the niche capacity is equal to the population size, the clearing effect disappears and the search method becomes a standard GA. Thus, the choice of the niche capacity between 1 and the population size provides intermediate situations between the maximum clearing effect and a standard GA search.

The algorithm 2 presents a clearing method as in Petrowski [10]. Consider $\mathcal{C}$ (the chromosomes population) and $n_{\mathcal{C}}$ (chromosomes number in the population) as the global variables. Adding, σ (the clearing radius) and κ (capacity of each niche) are the input parameters. The variable *nbWinner* counts the number of population winners associated to the current niche. The chromosomes population $\mathcal{C}$ is represented by a vector of $n_{\mathcal{C}}$ chromosomes.

Algorithm 2: Clearing niching

Input: σ (clearing radius), κ (capacity of each niche).
Output: Clearing – allocation of the niche resources for the the fittest individual.

```
SortFitness(C);
for i ← 0 to n_C − 1 do
    if Fitness(C[i]) > 0 then
        nbWinners ← 1;
        for j ← i + 1 to n_C − 1 do
            if Fitness(C[j]) > 0 and
            Distance(f(C[i]), f(C[j])) < σ then
                if nbWinners < κ then
                    nbWinners ← nbWinners + 1;
                else
                    Fitness(C[j]) ← 0;
                end
            end
        end
    end
end
```

The clearing algorithm (Algorithm 2) uses three functions:

- *SortFitness*($\mathcal{C}$): ordering the chromosomes population in the fitness order.
- *Fitness*($\mathcal{C}[i]$): returns the i-th chromosome ability of the population $\mathcal{C}$.
- *Distance*($f(\mathcal{C}[i])$, $f(\mathcal{C}[j])$): returns the distance between the phenotypes of population chromosomes;

- $f(\mathcal{C}[i])$: returns the chromosome $\mathcal{C}[i]$ phenotype. In our case the phenotype is a quantitative temporal association rule.

3.5 Calculating the distance between association rules

To take two association rules:

$$\begin{aligned} R &= (CA_R(v_{j_1}) \text{ AND } \dots \text{ AND } CA_R(v_{j_n})) \\ &\Rightarrow (CC_R(v_{j_1}) \text{ AND } \dots \text{ AND } CC_R(v_{j_m})) \end{aligned}$$

and

$$\begin{aligned} S &= (CA_S(v_{j_1}) \text{ AND } \dots \text{ AND } CA_S(v_{j_o})) \\ &\Rightarrow (CC_S(v_{j_1}) \text{ AND } \dots \text{ AND } CS_S(v_{j_p})) \end{aligned}$$

where $CA_R(v_{j_i})$, $CC_R(v_{j_i})$, $CA_S(v_{j_i})$, $CC_S(v_{j_i})$ are episodic conditions related to some variable v_{j_i} of the database. Each episodic condition has the form:

$$v_i \in [v0_i, v1_i] \text{ in the time interval } [t0_i, t1_i]$$

where v_i is any database variable. For a given rule R, the intersection of variables associated to the precedent (VA_R) episodic conditions with the variables associated to consequent (VC_R) episodic conditions must be empty, ie, $\text{VA}_R \cap \text{VC}_R = \emptyset$.

The distance between R and S, denoted by *Distance*(R, S) is given by the algorithm 3. For each episodic condition in the antecedent rule R, CA_R, we check if there is any episodic condition in the antecedent rule S, CA_S, which is comparable to CA_R. Two episodic conditions are comparable if they refer to the same variable. If we have two episodic conditions that are comparable, we calculate the distance between them. Otherwise, we increment the distance counter by one. Then the distance counter is divided by the number of episodic conditions n_{CA_R}. The same calculation is done to the subsequent episodic conditions. Finally, the distance among the rules is given by $(dist_A + dist_C)/2$, and the distance among episodic conditions, $DistanceEp(C_1, C_2)$, is given by algorithm 4.

The cyclical patterns mining and quasi-cyclical patterns mining is given by the Algorithm 5. The quasi-cycles are identified by the occurrence mapping or not of each detected rule for each cycle and period. Since each chromosome encode a rule, if the rule occurrence has a exactly cyclic pattern, then the chromosome fitness is one, corresponding to a rule frequency of 100% in the period. If the rule occurrence corresponds to a quasi-cyclic pattern, the chromosome fitness is the occurrence frequency of the rule.

4. Results

In order to evaluate the developed algorithm to detect cycles and quasi-cycles, we did an experiment using a synthetic binary sequence. We used in this experiment, the

Algorithm 3: *Distance*(R, S) – Distance between two quantitative temporal association rules.

Input: R and S, two quantitative temporal association rules.
Output: $dist$, distance between R and S.

$n_{CA_R} \leftarrow$ *numCond*(CA_R);
$dist_A \leftarrow 0$;
for $i \leftarrow 0$ **to** $n_{CA_R} - 1$ **do**
  if $\exists CA_S(v_{j_k}) | v_{j_k} = v_{j_i}, v_{j_i} \in VA_R$ **then**
    $dist_A \leftarrow dist_A + DistanceEp(CA_R(v_{j_i}), CA_S(v_{j_k}))$;
  else
    $dist_A \leftarrow dist_A + 1$;
  end
end
$dist_A \leftarrow dist_A / n_{CA_R}$;
$n_{CC_R} \leftarrow$ *numCond*(CC_R);
$dist_C \leftarrow 0$;
for $i \leftarrow 0$ **to** $n_{CC_R} - 1$ **do**
  if $\exists CC_S(v_{j_k}) | v_{j_k} = v_{j_i}, v_{j_i} \in VC_R$ **then**
    $dist_C \leftarrow dist_C + DistanceEp(CC_R(v_{j_i}), CC_S(v_{j_k}))$;
  else
    $dist_C \leftarrow dist_C + 1$;
  end
end
$dist_C \leftarrow dist_C / n_{CC_R}$;
$dist \leftarrow (dist_A + dist_C)/2$;

Algorithm 4: *DistanceEp*(C_1, C_2) – Distance between two temporal episodes associated to the same variable.

Input: C_1, C_2, v (two episodic conditions associated with the same variable v).
Output: $distEp$, distance between C_1 and C_2.

calculate the minimum value that v assume in the database and store into variable $\min v$;
calculate the maximun value that v assume in the database and store into variable $\max v$;
calculate the minimum value that t assume in the database and store into variable $\min t$;
calculate the maximun value that t assume in the database and store into variable $\max t$;
$dv \leftarrow \min\{(v_1^{(C_1)} - v_0^{(C_1)}), v_1^{(C_2)} - v_0^{(C_2)})\} - (\min\{v_1^{(C_1)}, v_1^{(C_2)}\} - \max\{v_0^{(C_1)}, v_0^{(C_2)}\})$;
$distV \leftarrow dv / (\max v - \min v)$;
$dt \leftarrow \min\{(t_1^{(C_1)} - t_0^{(C_1)}), t_1^{(C_2)} - t_0^{(C_2)})\} - (\min\{t_1^{(C_1)}, t_1^{(C_2)}\} - \max\{t_0^{(C_1)}, t_0^{(C_1)}\})$;
$distT \leftarrow dt / (\max t - \min t)$;
$distEp \leftarrow (distV + distT)/2$;

Algorithm 5: *Cycle_QuasiCycle_Detection*($BoolVet$, $nVars$) – Detecting cyclical patterns and quasi-cyclical patterns that express the rules detected occurrence.

Input: $boolVector$ (vector of boolean values that express the occurrence of each rule), $nVars$ (number of observed variables).
Output: list of detected cycles ($cyclesList$) and list of quasi-cycles detected ($quasiCyclesList$).

```
for j ← 0 to length(boolVector) do
    if boolVector[j] = 1 then
        positions.add(j) ;
    end
end
for i ← 1 to (length(boolVector)/2) do
    for k ← 0 to (length(positions) − 1) do
        quasiCycle[0] ← cycle[0] ← currentQuasi ←
        currentCycle ← positions[k];
        for j ← (k + 1) to (length(positions) − 1) do
            if positions[j] = (currentCycle + i) then
                currentCycle ← cycle[length(cycle) −
                1] ← positions[k];
            end
            if (positions[j] − currentQuasi)%i = 0
            then
                quasiCycle.add(positions[k]);
            end
        end
        nPeriods ←
        length(boolVector) − positions[k];
        if length(cycle) = (length(boolVector)/i)
        then
            cyclesList.add((i, positions[k]));
        end
        if length(quasiCycle) > 1 then
            if length(quasiCycle) = nPeriods then
                cyclesList.add((i, positions[k]));
            else
                if length(quasiCiclo) <
                lenght(boolVet)/i and quasiCiclo is
                not subset of any cycle ∈ listaCiclos
                then
                    frequency ←
                    length(quasiCycle)/nPeriods;
                    quasiCyclesList.add(
                        ((i, positions[k]), frequency));
                end
            end
        end
    end
end
```

```
Cycles:
      (1, 9)
      (1, 10)
      (3, 0)
      (4, 2)
      (4, 3)
      (5, 2)
      (5, 6)

Quasi−cycles:
      ((1, 0), 0.67)
      ((1, 2), 0.7)
      ((1, 3), 0.67)
      ((1, 6), 0.83)
      ((1, 7), 0.8)
      ((2, 0), 0.33)
      ((2, 2), 0.3)
      ((2, 3), 0.44)
      ((2, 6), 0.33)
      ((2, 7), 0.6)
      ((3, 2), 0.2)
      ((3, 7), 0.4)
      ((5, 0), 0.17)
```

Fig. 2: Cycles and quasi-cycles mined.

binary sequence "101100110111". The quasi-cycles and the cycles extracted from the sequence are shown in Figure 2.

We also performed two case studies in real quantitative temporal databases related to agrometeorological data, which are reported in the following. We used fixed values to the chromosomes population size (n_C=50), the generations number (*nGen*=250), the clearing radius (σ=0.5) and the maximum allowable variable range. The maximum allowable variable range is defined by $piv \cdot (\max v - \min v)$, where piv in these experiments has a value of 10%. We also kept the cross rate of 80% and the mutation rate of 3% for chromosome. In order to present the detected rules, we used a threshold of 0.4 for the fitness measure. Rules with lower fitness measure were not considered important because they have low occurrence frequency. The presentation of quasi-cycles were done in the same way.

4.1 Case study 1: Araraquara

In this case study, we used the database named Araraquara, collected by the Brazilian Agro-meteorological Monitoring System – Agritempo (http://www.agritempo.gov.br/). This database contains monthly agro-meteorological data of Araraquara, corresponding to the average minimum temperature values (*Tmin*), average maximun temperature values (*Tmax*), accumulated rainfall (*Prec*), average of Normalized Difference Vegetation Index (*NDVI*) and average of Water Requirement Satisfaction Index (*WRSI*). The data

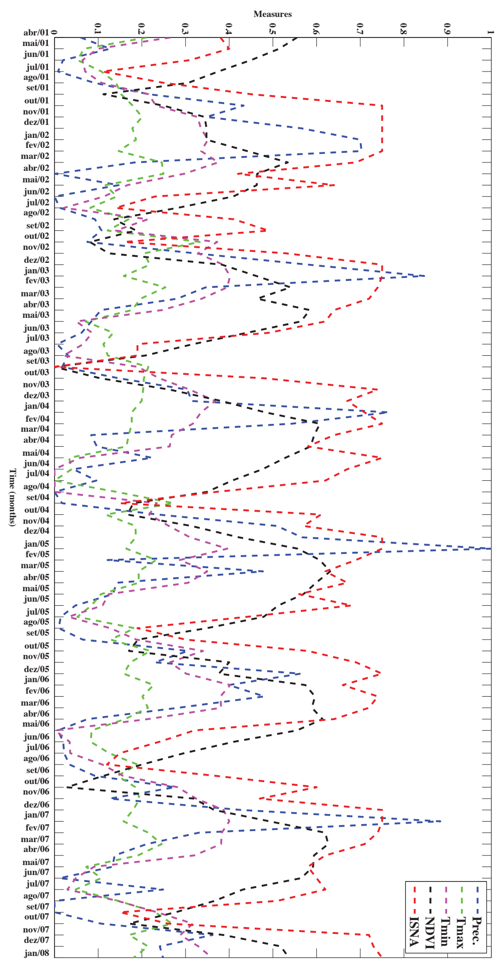


Fig. 3: Agrometeorological monthly measures of Araraquara's brazilian city.

```
Rule:
    Tmax (celsius) in [27.23, 28.33] on dez-jan
    ==>
    NDVI (taxa) in [0.58, 0.62] on dez-jan

    Fitness = 1.000

    Cycles:
    (1, 0)
```

```
Rule:
    WRSI
 (taxa) in [0.98, 1.00] on jan-fev
    ==>
    NDVI (taxa) in [0.58, 0.62] on jan-fev

    Fitness = 1.000

    Cycles:
    (1, 0)
```

```
Rule:
    Prec (mm) in [10.16, 61.17] on abr-mai
    ==>
    NDVI (taxa) in [0.26, 0.30] on abr-mai

    Fitness = 0.667

    Quasi-cycles:
    ((1, 0), 0.666667)
    ((1, 2), 0.666667)
    ((1, 3), 0.500000)
```

Fig. 4: The rules mined in the Araraquara's database.

corresponds to the period from April 2001 to January 2008. We plotted the variable values collected during these months in the Figure 3. Due to the large scale difference among the variables values, they were scaled in the range $[0, 1]$, so that we observe better the trends.

In this study, we had imposed the restriction that the it NDVI will be part of the rule consequent. The Figure 4 presents the mined rules together with the quasi-cycles and the cycles detected in the technical implementation using the parameters described in the section 4.

4.2 Case study 2: Piracicaba

In this case study, we use the cane sugar productivity database in Piracicaba, provided by Cepagri (Centre for Research in Agrometeorology and Climate Applied to Agriculture) maintained by State University of Campinas (UNICAMP). The database consists of four variables taken monthly in Piracicaba: the average minimum temperature (*Tmin*), the average maximum temperature (*Tmax*), average rainfall (*Prec*) and productivity of cane sugar (*Prod*) in tonnes per hectare (ton/hec). The database corresponds to the period from January 2003 to December 2009. We plotted the variable values collected during these months in the Figure 5.

In this study, we had imposed the restriction that productivity ($Prod$) will be part of the consequent of the rule. We fixed the used parameters, as described before in the Section 4. The mined rules, cycles and quasi-cycles detected for an algorithm execution are shown in Figure 6.

4.3 Considerations

Overall, we can see that the technique can mine several association rules and can detect cyclical patterns and *quasi-cyclical* occurrences of these rules. The chromosome fitness value is calculated using the highest detected pattern frequency, which have a maximum value of one, when a cycle is found. We fixed the parameters for the two case studies, but they can be different and change some results. For example, using a larger population size (n_C) and a larger generations number (*nGen*), we expect to mine a larger rules number. Likewise, how smaller the clearing radius, theoretically, the environment capacity to accommodate niches and, therefore, the amount of mined rules increase.

5. Conclusion

Given the quasi-cyclical patterns definition and the rules mining with quasi-cyclical temporal occurrence, we showed that is possible to detect relevant rules for the analyzed databases. These rules, in this format used, not can mined by any literature algorithm for cycles mining. Additionally, the quasi-cycle definition allows the relevant patterns identification, which could not be found using the exact cycle detection.

Therefore, in this work, we presented a methodology for quasi-cyclic patterns detection, which can be used to detect patterns that occur approximately cyclically from quantitative temporal data, as in the case studies 4.1 and 4.2.

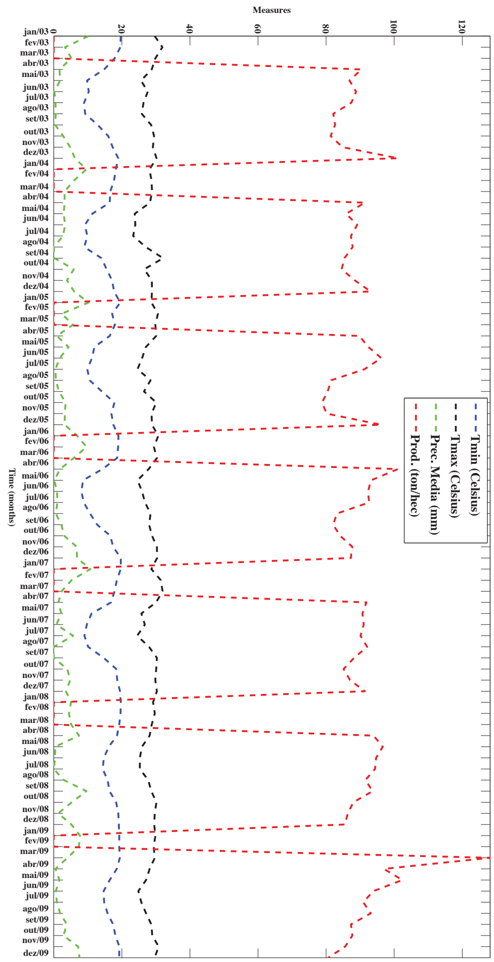


Fig. 5: Agrometeorological measures related to productivity of cane sugar in Piracicaba's brazilian city.

```
Rule:
    AvgPrec (mm/dia) in [2.50, 3.58] on out-nov
    ==>
    Prod
 (ton/hec) in [119.97, 128.20] on out-nov

    Fitness = 0.833

    Quasi-cycles:
    ((1, 0), 0.833333)
    ((1, 2), 0.666667)
    ((1, 3), 0.500000)
```

```
Rule:
    Tmin (celsius) in [17.93, 19.10] on abr-mai
    ==>
    Prod
 (ton/hec) in [119.97, 128.20] on abr-mai

    Fitness = 0.429

    Quasi-cycles:
    ((1, 5), 0.428571)
```

```
Rule:
    Tmax (celsius) in [28.35, 29.23] on nov-dez
    ==>
    Prod
 (ton/hec) in [119.97, 128.20] on nov-dez

    Fitness = 0.429

    Quasi-cycles:
    ((1, 0), 0.428571)
    ((1, 1), 0.428571)
```

Fig. 6: The rules mined in the Piracicaba's database.

This pattern type is extremely common in nature, according to we discussed in the introduction of this article.

As future work, we intend to set the genetic algorithm parameters, which were kept fixed in all experiments described in this work. We expect that this adjustment will allow to find a larger number of relevant rules. Other future work is to develop and to test other fitness measures, or improve the adopted fitness measure.

Acknowledgments

The authors would like to thank the Brazilian National Council for Scientific and Technological Development (CNPq) and Research Support Foundation of Goiás State (FAPEG).

References

[1] S. Amo, N. A. Silva, R. P. Silva, and F. S. Pereira. Tree pattern mining with tree automata constraints. *Information Systems*, 35:570–591, 2010.

[2] B. Catania and A. Maddalena. A unified framework for heterogeneous patterns. *Information Systems*, 37:460–483, 2012.

[3] D.-A. Chiang, C.-T. Wang, S.-P. Chen, and C.-C. Chen. The Cyclic Model Analysis on Sequential Patterns. *IEEE Transactions on Knowledge and Data Engineering*, 21(11):1617–1628, 2009.

[4] J. K. Febrer-Hernández and J. Hernández-Palancar. Sequential pattern mining algorithms review. *Intelligent Data Analysis*, 16(3):451–466, 2012.

[5] T. Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011.

[6] Y. Hai and X. Li. A general temporal association rule frequent itemsets mining algorithm. *International Journal of Advancements in Computing Technology*, 3(11):63–71, 2011.

[7] Yong Joon Lee, Jun Wook Lee, Duck Jin Chai, Bu Hyun Hwang, and Keun Ho Ryu. Mining temporal interval relational rules from temporal data. *Journal of Systems and Software*, 82(1):155–167, 2009.

[8] N. R. Mabroukeh and C. I. Ezeife. A taxonomy of sequential pattern mining algorithms. *ACM Computing Surveys*, 43(1), 2010.

[9] B. Ozden, S. Ramaswamy, and A. Silberschatz. Cyclic Association Rules. In *Proceedings of the Fourteenth International Conference on Data Engineering*, pages 412–421, Washington, DC, USA, 1998.

[10] A. Pétrowski. A clearing procedure as a niching method for genetic algorithms. In *Proceedings of 3rd IEEE International Conference on Evolutionary Computation*, pages 798–803, New York, USA, 1996.

[11] M. Plantevit, A. Laurent, D. Laurent, M. Teisseire, and Y. W. Choong. Mining multidimensional and multilevel sequential patterns. *ACM Transactions on Knowledge Discovery from Data*, 4(1):1–37, 2010.

[12] N. Tatti and B. Cule. Mining closed strict episodes. *Data Mining and Knowledge Discovery*, 25(1):34–66, 2011.

[13] X. Yan, C. Zhang, and S Zhang. Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. *Expert Systems with Applications*, 36:3066–3076, 2009.

[14] M. Zhang and C. He. Survey on association rules mining algorithms. *Advancing Computing, Communication, Control and Management*, 56:111–118, 2010.

Interpreting the Geochemistry of Southern California Granitic Rocks using Machine Learning

Germán H. Alférez[1], Jocksan Rodríguez[1], Benjamin Clausen[2], and Lance Pompe[2]
[1]Global Software Lab, Facultad de Ingeniería, Universidad de Montemorelos, Montemorelos, N.L., Mexico
[2]Geoscience Research Institute, Department of Earth and Biological Sciences, Loma Linda University, Loma Linda, CA, USA

Abstract – *Extensive geochemical analyses have been done on granitic rocks in southern California. Almost forty elements were measured for each of several hundred samples. In our previous work, we analyzed the geochemical components of these rocks using two methods, namely Principal Component Analysis (PCA) and Geographic Information Systems (GIS). In this paper, machine learning is used to validate the results previously obtained. We describe an evaluation in which it was found that the results obtained with machine learning are similar to the results obtained by means of PCA and GIS.*

Keywords: Machine Learning, Principle Component Analysis, Geographic Information Systems, Geochemistry.

1 Introduction

A combination of disciplines such as geochemistry and computer science can provide a powerful tool for conducting a thorough study of rocks of interest. Geochemistry helps one to determine the physical conditions under which the rocks formed and the chemical distribution or redistribution of elements over geologic time [1]. Here we are studying the Cretaceous batholithic rocks in southern California [2], which were emplaced in a plate tectonic subduction zone. A batholith (or large granitic body) covers more than one hundred square kilometers in the crust [3, 4].

In previous work [5], we used two approaches to understand the statistical and spatial geochemistry variation of part of the aforementioned area: Principal Component Analysis (PCA) and Geographic Information Systems (GIS). In that data analysis, we used 287 samples from a large systematically collected granitoid geochemical data set [6].

In this work, our contribution is to compare our previous geochemical interpretation of the Californian northern Peninsular Ranges Batholith based on PCA and GIS, and the results from machine learning based on a larger data set with almost 800 samples that comes from a larger area in southern California. This data set includes the 287 samples used for PCA and GIS [6]. We decided to use a larger data set for machine learning analysis to get results as accurate as possible according to our most exhaustive and updated data space.

We believe that our results are of interest to geologists because they demonstrate that analysis of geochemical data with PCA and GIS, as well as machine learning, can elucidate plate tectonic environments. Specifically, in this study we used the Simple K-Means method of machine learning.

This paper is organized as follows. Section II presents the basis of our approach. Section III presents the geochemical analysis by means of machine learning. Section IV presents related work. Finally, Section V presents conclusions and future work.

2 Basis of our approach

In order to understand our approach, it is important to describe the underlying concepts. First, on the one hand PCA is a statistical method based on the variance between variables where high-dimensional data is transformed into low dimensional data. This method can be used to detect coherent patterns [7]. On the other hand, GIS is a way to approximate the values of the discrete sample points over the whole study region, attempting to recreate the continuous geochemical variation that was discretely sampled in the field [8].

In our previous work [5], multivariate outliers were identified using Mahalanobis distance [9], and excluded. Then four components identified by PCA were mapped with GIS to observe their spatial distribution. Bivariate plots relating the component variable to the distance from the transition zone between oceanic and continental crust were used to better understand the trends.

Data were analyzed using PCA with IBM SPSS. Using this method, we were able to reduce 40 geochemical variables to 4 components, which are approximately related to the compatible, High Field Strength (HFS), Heavy Rare Earth (HRE), and Large Ion Lithophile (LIL) elements. The 4 components were interpreted as follows: 1) compatible (and negatively correlated incompatible) elements indicate extent of differentiation as typified by SiO_2; 2) HFS elements indicate crustal contamination as typified by Sr_i; 3) HRE elements indicate source depth as typified by the Gd/Yb ratio; and 4) LIL elements indicate alkalinity as typified by the K_2O/SiO_2 ratio. Note that concentrations for major elements are usually expressed as percent major oxide. Also, Sr_i is a calculated $^{87}Sr/^{86}Sr$ ratio.

Our goal in this paper is to analyze the geochemical data of the southern California granitic rocks using machine learning. Machine learning is a branch of Artificial Intelligence, which studies agents or programs that learn or evolve based on experience to perform a particular task better [10].

There are many machine learning methods for data analysis. One of the most popular is Simple K-Means [11]. Simple K-Means is a clustering technique with a relatively simple implementation. The goal of clustering is to partition a set of objects, which have associated multidimensional vectors of attributes in homogeneous groups (i.e., the "K"); such that patterns in each group are similar.

There are four steps to describe the functionality of Simple K-Means [12, 13]: 1) a set of objects to be partitioned, the number of groups, and each group's centroid are defined; 2) for each object in the data set, the nearest centroid is determined, and the object is added to the group related to that centroid; 3) for each generated group, the centroid is recalculated; and 4) multiple convergence conditions are used. The most common ones are the following: converge when a number of iterations has been reached, converge when there is no exchange of objects among the groups, or converge when the difference among centroids in two iterations is smaller than a given threshold. If the convergence condition is not satisfied, steps two, three, and four are repeated.

3 Geochemical analysis by means of machine learning

In this study, WEKA was used to carry out the geochemical analysis of the southern California granitic rocks [14]. WEKA is a free tool written in Java that has a large number of data analysis techniques, such as preprocessing and clustering. It also facilitates data visualization.

In this section we present the comparison between our previous results with PCA and GIS, and our present results with Simple K-Means for the following geochemical factors: SiO_2, Sr_i, Gd/Yb, and K_2O/SiO_2.

3.1 SiO_2 analysis

Through PCA and GIS, we found that the extent of differentiation is more uniformly high or low in the East and more intermediate in the West.

A trend surface analysis interpolation map of SiO_2 shows the spatial distribution to be high in the far West, low in the West Central, and moderately high in the East. Figure 1 shows the distribution of this oxide. Red areas represent a high concentration of SiO_2 and blue areas show a low concentration. The other colors indicate intermediate concentrations.

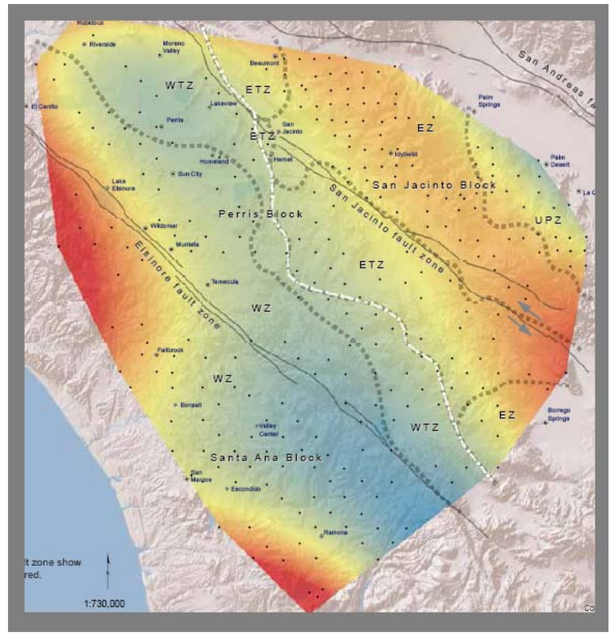


Figure 1. Spatial distribution of SiO_2. The zones in red have a concentration above 70%. The zones in blue have a concentration below 60%

For the same SiO_2 oxide from the larger data set, the results with Simple K-Means can be seen in Table 1. In our experiments, on the one hand we found that within a cluster the sum of squared errors decreases as the number of clusters increases. On the other hand, we found that if a very large number of clusters is generated, then some of them will have a very small number of samples. This fact can produce inconsistent results. We argue that it is important to have a balance between the error and the average number of clusters. In our case, we realized that four clusters gave us the best balance when analyzing SiO_2 and the other geochemical variables.

Table 1. WEKA results for percent SiO_2

Cluster #	Number of samples	Oxide concentration
0	104	54.4%
1	294	63.4%
2	181	73.4%
3	192	68.0%

With the data in Table 1, it was possible to generate Figure 2. The horizontal axis indicates longitude and the vertical axis latitude. Cluster 0 is in blue, Cluster 1 is in yellow, Cluster 2 is in red, and Cluster 3 is in green.

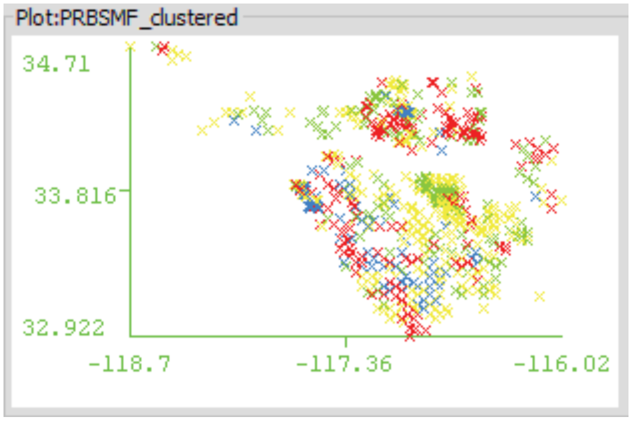


Figure 2. Cluster assignment visualization for SiO_2. Cluster 0 is in blue, Cluster 1 is in yellow, Cluster 2 is in red, and Cluster 3 is in green

A similarity can be observed between the concentration of elements in Figure 1 and the lower half of Figure 2. For instance, Cluster 2 (which is in red) has a high concentration of SiO_2 (73.4%); whereas, cluster 0 (which is in blue) has a low concentration of this oxide (54.4%). These results reflect a similarity with the results in the map of Figure 1.

3.2 Sr_i analysis

The analysis using PCA and GIS on the one hand shows a low Sr_i in the West and an increasing Sr_i to the East. Higher values indicate greater crustal contamination. Figure 3 was generated using kriging interpolation. The blue color represents a low value of Sr_i, whereas the red color shows a high value. Table 2 shows the results with Simple K-Means for this element.

Table 2. WEKA results for Sr_i

Cluster #	Number of samples	Isotope ratio
0	135	0.7091
1	358	0.7068
2	31	0.7126
3	243	0.7042

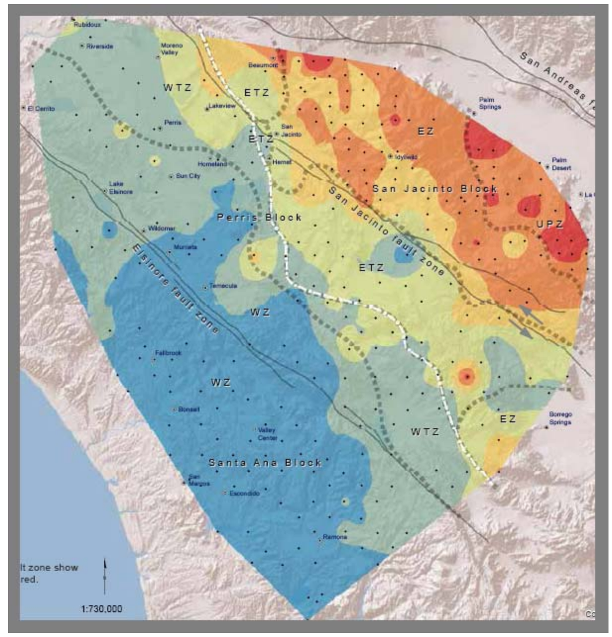


Figure 3. Spatial distribution of Sr_i. The zones in red have a value greater than 0.707 for this variable. The zones in blue have a value less than 0.705

The visual description of the concentration and distribution of Sr_i is presented in Figure 4. Cluster 0 is in yellow, Cluster 1 is in green, Cluster 2 is in red, and Cluster 3 is in blue.

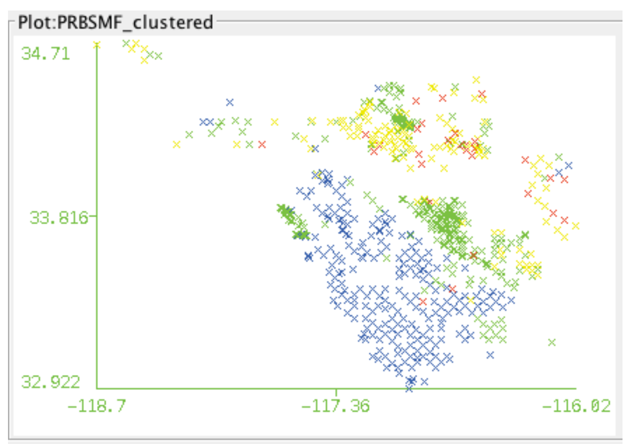


Figure 4. Cluster assignment visualization for Sr_i. Cluster 0 is in yellow, Cluster 1 is in green, Cluster 2 is in red, and Cluster 3 is in blue

Cluster 3 has very low values similar to what is found in Figure 3. Likewise, Cluster 1 has higher values, also similar to what is found in Figure 3. The results reflect a similarity with the results in the map of Figure 3.

3.3 Gd/Yb analysis

According to the experiments with PCA and GIS, Gd/Yb ratios are related to magma source depth (see Figure 5). In this map, the West is uniformly low indicating a shallow magma source depth.

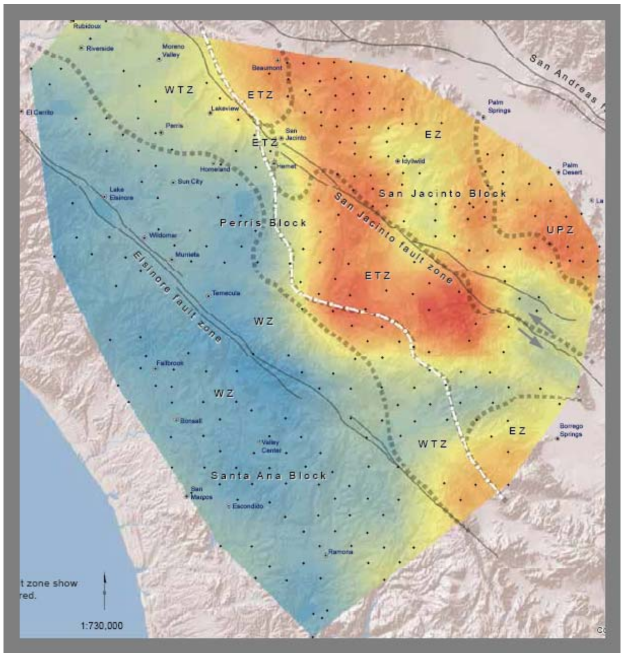


Figure 5. Spatial distribution of Gd/Yb. The zones in red have a high concentration above 2 for this ratio. The zones in blue have a low concentration below 2 for this ratio

The results with Simple K-Means for Gd/Yb are shown in Table 3.

Table 3. WEKA results for Gd/Yb

Cluster #	Number of samples	Element ratios
0	461	2.4
1	96	3.6
2	119	1.8
3	95	1.3

Figure 6 was generated according to the data in Table 3. Cluster 0 is in yellow, Cluster 1 is in red, Cluster 2 is in blue, and Cluster 3 is in green. The bottom half of this map is similar to the one shown in Figure 5. Specifically, Cluster 1 (which is in red) has the highest ratio. In contrast, Cluster 3 (which is in blue) has the lowest ratio.

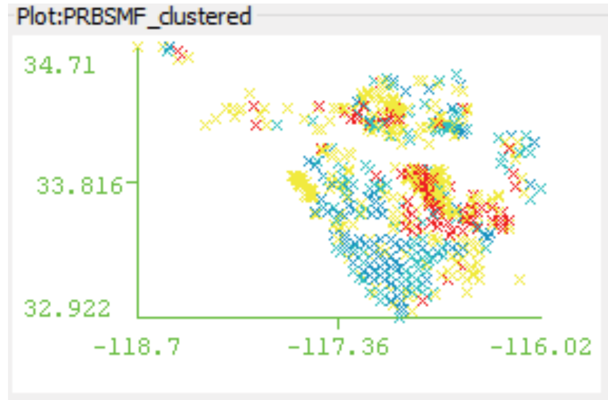


Figure 6. Cluster assignment visualization for Gd/Yb. Cluster 0 is in yellow, Cluster 1 is in red, Cluster 2 is in blue, and Cluster 3 is in green

3.4 K_2O/SiO_2 analysis

According to our study based on PCA and GIS, the map in Figure 7 shows the distribution of K_2O/SiO_2, which indicates alkalinity. The red color represents a larger ratio and the blue color represents a lower ratio. The yellow and orange colors represent intermediate ratios.

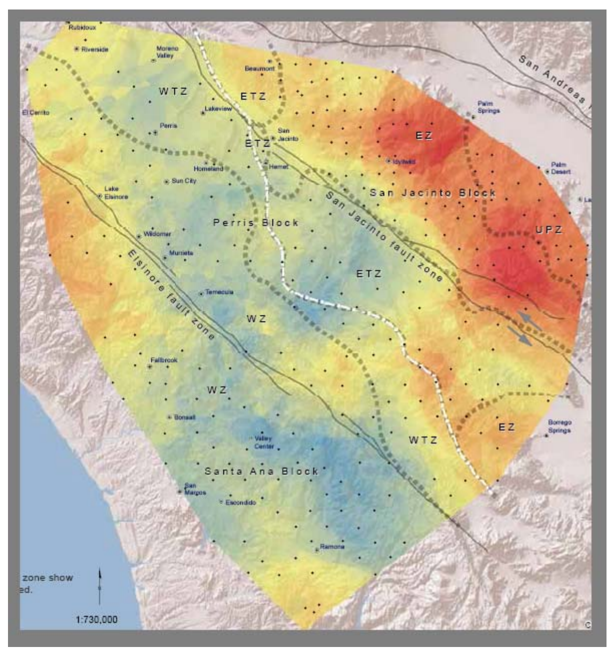


Figure 7. Spatial distribution of K_2O/SiO_2. The zones in red have a high ratio above 0.3. The zones in blue have a low ratio below 0.3

The results with Simple K-Means for K_2O/SiO_2 are shown in Table 4.

Table 4. WEKA results for K_2O/SiO_2

Cluster #	Number of samples	Ratio values
0	277	0.045
1	81	0.007
2	164	0.066
3	249	0.029

The spatial distribution of K_2O/SiO_2 is presented in Figure 8. Cluster 0 is in yellow, Cluster 1 is in blue, Cluster 2 is in red, and Cluster 3 is in orange. High ratios, which are represented in red, are in Cluster 2. In contrast, low ratios in Cluster 1 are in blue. Figure 7 and 8 show similar results.

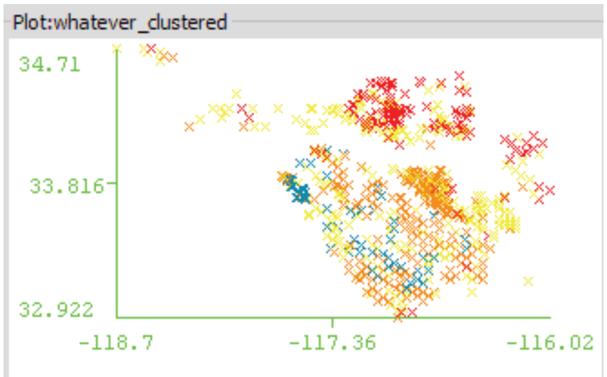


Figure 8. Cluster assignment visualization for K_2O/SiO_2. Cluster 0 is in yellow, Cluster 1 is in blue, Cluster 2 is in red, and Cluster 3 is in orange

4 Related work

Increasingly larger geochemical data sets are becoming available from the geology literature. The purpose for the current research project is to determine what new information can be gleaned from these data sets using statistical analysis, geospatial analysis, and machine learning techniques.

Early geochemical clustering done by Pearce et al. [15] was able to discriminate between granitic-type rocks from different plate tectonic environments just by using pairs of trace elements displayed on bivariate plots. Sr_i values have been used to discriminate between granitic magma sources from the Earth's mantle and the Earth's crust [16, 17]. The ratio between light and heavy rare earth elements has been used to discriminate between granitic magma from shallow and deep sources [18]. Instead of using only two or three elements to group the data into clusters, this research is asking whether it is possible to use PCA, GIS, and machine learning to group large geochemical data sets more effectively and to find new patterns.

Grunsky et al. [19] has been able to classify volcanic rocks into three types using their major element geochemistry. Grunsky and Smee [20] have used PCA and digital topography to visualize, classify, and interpret the geochemistry of 1665 soil samples based on 27 elements. Grunsky [21] used thousands of observations with as many as fifty elements for process identification and pattern discovery using multivariate data analysis and geospatial analysis. Templ et al. [22] used cluster analysis on geochemical data to group samples from northern Europe. Classic books on geostatistical analysis of compositional data include Aitchison [23] and Pawlowsky-Glahn and Olea [24].

Machine learning approaches have shown promising results when applied to complex geological problems involving big data sets. For example, Lüdtke et al. [25] used a supervised machine learning technique to automatically analyze large quantities of spatially referenced seafloor video mosaics of mud volcanoes. Classification accuracy and speed varied between four commonly applied machine leaning classifiers, namely support vector machines, K-nearest-neighbors classifier, C4.5 decision trees, and the naïve Bayes classifier. Classification rates of up to 98.86% were achieved on the full data set with support vector machines when cross-validated with the training data. An average error rate of 1.52% was found when testing the system over a reference data set covering 60% of the investigation area.

Some of the most recent machine learning techniques have been used in discriminating tsunami deposits in Japan [26], predicting acid mine drainage [27], and prospecting for minerals [28, 29].

5 Conclusions and future work

In this paper, we presented an approach to carry out geochemical analysis by means of machine leaning. Specifically, we have focused our analysis on Simple K-Means. We demonstrated that the results with PCA and GIS are similar to the results found with Simple K-Means. This is an important finding because geologists will be able to: 1) use machine learning to validate what they find with statistical tools; or 2) use machine learning to obtain fast results with easily available tools, such as WEKA.

In the future we would like to explore other ways to use machine learning to analyze geochemical data and geological events. For instance, Could we predict possible earthquakes by means of generating forecasts based on historical data?

6 References

[1] F. Albarède, 2003. Geochemistry: An Introduction. Cambridge University Press.

[2] A. K. Baird and A. T. Miesch, 1984. Batholithic rocks of southern California—a model for the

petrochemical nature of their source materials. U. S. Geological Survey Professional Paper 1284.
[3] W. Pitcher, 1993. The Nature and Origin of Granite. Blackie Academic & Professional, pp. 193-217.
[4] A. Hall, 1987. Igneous Petrology. Longman Scientific & Technical, pp. 91-93.
[5] L. Pompe, B. L. Clausen, and D. M. Morton, December 17, 2014. Interpreting the Geochemistry of the northern Peninsular Ranges Batholith Using Principle Component Analysis and Spatial Interpolation. Abstract V33B-4852 presented at 2014 Fall Meeting, AGU, San Francisco, CA, 15-19 Dec. https://agu.confex.com/agu/fm14/meetingapp.cgi#Paper/13737
[6] D. M., Morton, et al., 2014. Framework and petrogenesis of the northern Peninsular Ranges batholith, southern California. In: Peninsular Ranges batholith, Baja California and southern California. D. M. Morton and F. K. Miller, eds. GSA Memoir 211, pp. 61-143.
[7] I. Jolliffe, 2002. Principal Component Analysis, Series: Springer Series in Statistics, 2nd ed. Springer, NY, XXIX, pp. 487.
[8] E. C. Grunsky, 2010. The interpretation of geochemical survey data. Geochemistry: Exploration, Environment Analysis 10, pp. 27-74.
[9] P.C. Mahalanobis, 1936. On the generalised distance in statistics. Proceedings of the National Institute of Sciences of India 2 (1), pp 49–55.
[10] T. Mitchell, 1997. Machine Learning, McGraw Hill.
[11] J. MacQueen, 1967. Some Methods for Classification and Analysis of Multivariate Observations. Proc. 5th Berkeley Symp. Math. Statistics and Probability, 1, pp. 28 -297.
[12] T. Kanungo, N. S., Netanyahu, A.Y., and Wu, 2002. An Efficient K-Means Clustering Algorithm: Analysis and Implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 7.
[13] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y., Wu, (2002). A Local Search Approximation Algorithm for K-Means Clustering. Proc. 18th Annual ACM Symposium on Computational Geometry (SoCG'02), pp. 10-18.
[14] I. H. Witten and E. Frank, 2000. Data Mining. Morgan Kaufmann Publishers.
[15] J. A. Pearce, et al., 1984. Trace element discrimination diagrams for the tectonic interpretation of granitic rocks. Journal of Petrology 25, pp. 956-983.
[16] R. W. Kistler and Z. E. Peterman, 1973. Variations in Sr, Rb, K, Na, and initial Sr87/Sr86 in Mesozoic granitic rocks and intruded wall rocks in central California. Geological Society of America Bulletin 84, pp. 3489-3511.
[17] V. E. Langenheim, et al., 2004. Geophysical and isotopic mapping of preexisting crustal structures that influenced the location and development of the San Jacinto fault zone, southern California. Geological Society of America Bulletin 116(9/10), pp. 1143-1157.
[18] L. P. Gromet and L. T. Silver, 1987. REE variations across the peninsular ranges batholith: implications for batholithic petrogenesis and crustal growth in magmatic arcs. Journal of Petrology 28, pp. 75-125.
[19] E. C. Grunsky, et al., 1992. Characterization and statistical classification of Archean volcanic rocks of the Superior Province using major element geochemistry. In: Geology of Ontario. P. C. Thurston, H. R. Williams, R. H. Sutcliffe, and G. M. Stott, eds. Ontario Geological Survey, Special Volume 4, Part 2, pp. 1397-1438.
[20] E. C. Grunsky and B. W. Smee, 1999. The differentiation of soil types and mineralization from multi-element geochemistry using multivariate methods and digital topography. Journal of Geochemical Exploration 67, pp. 287-299.
[21] E. C. Grunsky, 2010. The interpretation of geochemical survey data. Geochemistry: Exploration, Environment Analysis 10, pp. 27-74.
[22] M. Templ, et al., 2008. Cluster analysis applied to regional geochemical data: Problems and possibilities. Applied Geochemistry 23, pp. 2198-2213.
[23] J. Aitchison, 1986. The statistical analysis of compositional data. New York, Chapman and Hall.
[24] V. Pawlowsky-Glahn and R. A. Olea, 2004. Geostatistical analysis of compositional data, Oxford University Press.
[25] A. Lüdtke, K. Jerosch, O. Herzog, and M. Schlüter, 2012. Development of a machine learning technique for automatic analysis of seafloor image data: Case example, Pogonophora coverage at mud volcanoes. Computers and Geosciences 39, pp. 120-128.
[26] T. Kuwatani, et al., 2014. Machine-learning techniques for geochemical discrimination of 2011 Tohoku tsunami deposits. Scientific Reports 4; doi: 10.1038/srep07077

[27] G. D. Betrie, et al., 2013. Predicting copper concentrations in acid mine drainage: a comparative analysis of five machine learning techniques. Environmental Monitoring and Assessment 185, pp. 4171-4182.

[28] R. Zuo, and E. J. M. Carranza (2011). Support vector machine: A tool for mapping mineral prospectivity. Computers and Geosciences 37, pp. 1967-1975.

[29] M. Abedi, et al., 2012. Support vector machine for multi-classification of mineral prospectivity areas. Computers and Geosciences 46, pp. 272-283.

An Interval Expectation Maximization Algorithm for Outlier Detection in Linear Regression

Daniel B. Barreiros[1], **Marco A.O. Domingues**[2], **Renata M.C.R. Souza**[1], **and Francisco J. A. Cysneiros**[3]
[1]Centro de Informática, Universidade Federal de Pernambuco, Recife, PE, Brazil. dbb2@cin.ufpe.br, rmcrs@cin.ufpe.br
[2]CTADS, Instituto Federal de Pernambuco, Recife, PE, Brazil. marcodomingues@recife.ifpe.edu.br
[3]Departamento de Estatística, Universidade Federal de Pernambuco, Recife, PE, Brazil. cysneiros@de.ufpe.br

Abstract— *Outlier detection has an important role in diverse fields of research and application domains including pattern recognition, exploratory data analysis and data mining. In classical regression analysis, these outliers are often removed from the data set, being usually regarded as errors of the process. However, in SDA domain, this procedure is unsuitable because a single symbolic data observation may represent the generalization of a subset of other classical observations. This paper introduces an expectation-maximization algorithm for interval data in order to detect atypical intervals concerned with regression analysis problems. The algorithm is evaluated regarding different simulated and real interval data sets.*

Keywords: Outlier detection, Expectation-Maximization, Interval Data, Symbolic Data Analysis

1. Introduction

Interval data have been considered in real world applications: analysis of census data [1], electricity load profiling [2], scientific production of researches [3]. This kind of data has been studied mainly in *Symbolic Data Analysis* (SDA) which is a research field related to multivariate analysis, pattern recognition and artificial intelligence [4]. SDA aims to provide a comprehensive way to summarize data sets by means of symbolic data resulting in a smaller and more manageable data set which preserves the essential information. In the literature of SDA, several approaches for interval data have been introduced: recommendation systems [5], classification [6], principal component analysis [7], regression [8]. Symbolic data allow multiple values for each variable. These new variables (set-valued, interval-valued, and histogram-valued) make it possible to hold data intrinsic variability and/or uncertainty from the original data set as shown in [4].

Interval-valued data arise in practical situations such as recording monthly interval temperatures in meteorological stations, daily interval stock prices, among others. Another common source of interval data is the aggregation of data into a reduced number of groups. In this case, SDA starts extracting knowledge from a data set in order to provide symbolic descriptions that are mathematically modeled by a generalization process applied to a set of individuals. An example is an amanita mushroom specie data set formed by 23 mushroom species. The intervals of this data set were obtained by aggregating individual mushrooms according to the kind of species. Each individual mushroom is described by three interval variables that are: stipe length, stipe thickness and pileus cap.

Figure 1 shows the amanita data set. In this figure, we can observe that there are two intervals which are substantially different from all other ones. They were obtained from the generalization process applied to the amanita data set.

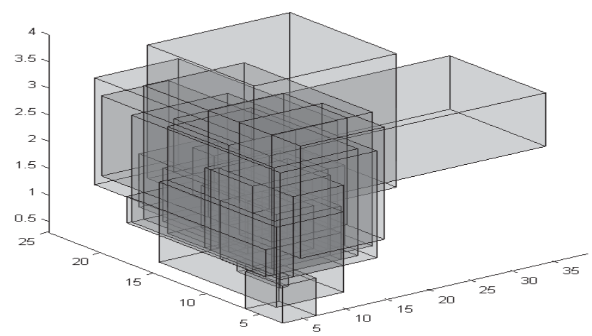


Fig. 1: Interval *Amanita* mushroom data

According to [4], overgeneralization problems can arise when these extreme values are actually outliers or when the set of individuals to generalize is in fact composed of subsets of different distributions. Indeed, in these situations interval outliers can be found, as it is highlighted in the amanita interval data set, and methods that identify them are essential. Investigation methods of outliers as primary analysis is an opened research topic.

In classical data analysis, point outliers are observations in a data set which do not follow the pattern of the other observations. Such data play important role in regression since they can lead to inaccurate regression estimates. It is a common practice to distinguish between two types of outliers: on the response variable, called outlier, represents a model failure and may indicate a sample peculiarity, a data entry error or another problem; and with respect to the predictors variables, called leverage points. This paper addresses outliers on response variables.

In SDA, interval outliers are also unusual observations and interval regression is an extension of the classical regression for symbolic interval data [8]. In the amanita data set of the Figure 1, the regression problem concerns to estimate pileus cap (response interval variable) from stipe length, stipe thickness (predictor interval variable).

The main contribution of this work is to propose an EM-type algorithm regarding a multivariate gaussian mixture model for interval data to identify atypical intervals in regression analysis. The proposed algorithm is evaluated with different real and simulated interval data sets. For simulated interval data, the performance of the proposed algorithm is measured by the false negative and false positive rates in the framework of a Monte Carlo experiment.

The rest of the paper is organized in the following form: Section 2 presents the simulated and real data sets used in this work. Section 3 describes the EM-type algorithm for detecting atypical intervals. Section 4 presents a performance analysis. Finally, Section 5 gives the concluding remarks.

2. Interval data sets

Different simulated interval data sets which comprises two arrangements for interval outliers and the Amanita interval data set are presented in this section.

2.1 Simulated interval data sets containing outliers

Initially, each seed s_i^x $(i = 1, \ldots, n)$ on coordinate X arises from an uniform distribution $[a, b]$. A seed s_i^y on coordinate Y is related to the seed s_i^x as $s_i^y = \beta_0 +\ \beta_1 s_i^x + \varepsilon_i$ $(i = 1, \ldots, n)$ where β_0 and β_1 are simulated from an uniform distribution $[c, d]$ and ε_i is simulated from a standard normal distribution.

Thus, seed data sets are now formed by bivariate points (s_i^x, s_i^y) $(i = 1, \ldots, n)$. For each point i, a random sample of size 30 is drawn from a bivariate gaussian distribution with mean vector and the diagonal covariance matrix $\boldsymbol{\mu}= (s_i^x, s_i^y)$ and $\Sigma = \sigma\mathbf{I}$ where σ is a parameter of scale. From each sample, the rectangle i is defined by a vector of two intervals

$$\mathbf{v} = (x_i = [a_i, b_i], y_i = [\lambda, \gamma])'$$

where $a_i = Q_1^x$, $b_i = Q_3^x$, $\lambda_i = Q_1^y$ and $\gamma_i = Q_3^y$ are first and third quartiles of the samples on coordinates X and Y, respectively.

Interval outliers are created in the following way. First of all, the sets are sorted ascending by the dependent variable Y^c and a small cluster containing the m first points of the sorted set (y_i^c, x_i^c) is selected. The observations of this cluster are changed into outlier points by

$$x_i^c = x_i^c + f_x.S(X^c)$$

$$y_i^c = y_i^c + f_y.S(Y^c)$$

where $S(Y^c)$ and $S(X^c)$ are, respectively, the standard deviation of $(y_1^c, \ldots, y_n^c)$ and the standard deviation of $(x_1^c, \ldots, x_n^c)$, and, f_x and f_y are fixed values.

Two different configurations for rectangles containing remote intervals in terms of position (center of the intervals) are considered in this paper. Figures 2 and 3 display the interval data sets 1 and 2, respectively, with $s^x \sim U[a, b] = [10, 40]$, $\beta_0, \beta_1 \sim U[c, d] = [1, 10]$, $n = 50$ and $\sigma = 3$. Figure 2 ($f_x = 0$ and $f_y = 10$) shows a scenario in which there are intervals that are strongly outliers. Figure 3 ($f_x = 5$ and $f_y = 10$) considers a data set with a group of intervals that are slightly outliers.

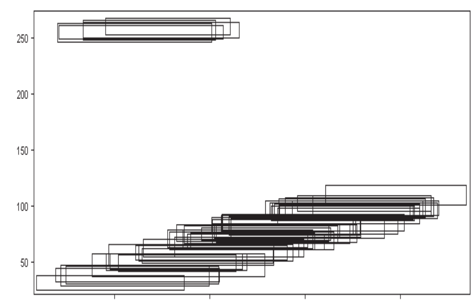


Fig. 2: Interval data set 1 containing intervals that are strongly outliers.

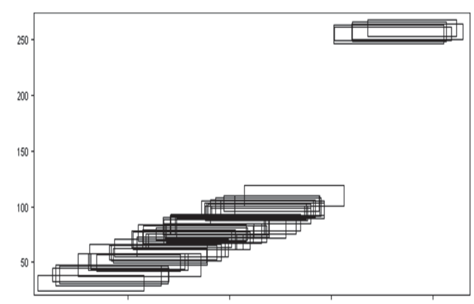


Fig. 3: Interval data set 2 containing intervals that are slightly outliers.

2.2 Amanita interval data set

Table 1 shows a mushroom specie data set. These mushroom species are members of the genus *Amanita* in which the values were collected from the *Fungi of California Species Index* ($http : //www.mykoweb.com /CAF/species\ \_index.\ html$).

From the values in the table above, three species are candidate outliers on the response variable. They are: Lanei, Muscaria and Pachycolea. Regarding the pileus cap response variable, the Lanei and Muscaria species have atypical intervals.

Table 1: Ranges of pileus cap, stipe length and stipe thickness of the *Amanita* mushroom family.

Amanita species	Interval Variables		
	Pileus Cap	Stipe Length	Stipe Thickness
Lanei	[8.00 : 25.00]	[10.00 : 20.00]	[1.50 : 4.00]
Constricta	[6.00 : 12.00]	[9.00 : 17.00]	[1.00 : 2.00]
Franchetii	[4.00 : 12.00]	[5.00 : 15.00]	[1.00 : 2.00]
Novinupta	[5.00 : 14.00]	[6.00 : 12.00]	[1.50 : 3.50]
Muscaria	[6.00 : 39.00]	[7.00 : 16.00]	[2.00 : 3.00]
Ocreata	[5.00 : 13.00]	[10.00 : 22.00]	[1.50 : 3.00]
Pachycolea	[8.00 : 18.00]	[10.00 : 25.00]	[1.00 : 3.00]
Pantherina	[4.00 : 15.00]	[7.00 : 11.00]	[1.00 : 2.50]
Phalloides	[3.50 : 15.00]	[4.00 : 18.00]	[1.00 : 3.00]
Protecta	[4.00 : 14.00]	[5.00 : 15.00]	[1.00 : 3.00]
Vaginata	[5.50 : 10.00]	[6.00 : 13.00]	[1.20 : 2.00]
Velosa	[5.00 : 11.00]	[4.00 : 11.00]	[1.00 : 2.50]
Aprica	[5.00 : 15.00]	[3.30 : 9.10]	[1.40 : 3.50]
Bivolvata	[7.00 : 10.00]	[13.00 : 15.00]	[1.60 : 2.50]
Gemmata	[3.00 : 11.00]	[4.00 : 15.00]	[0.50 : 2.00]
Magniverrucata	[4.00 : 13.00]	[7.00 : 11.50]	[1.00 : 2.50]
Smithiana	[5.00 : 17.00]	[6.00 : 18.00]	[1.00 : 3.50]
Cokeri	[7.00 : 15.00]	[10.00 : 20.00]	[1.00 : 2.00]
Porphyria	[3.00 : 12.00]	[5.00 : 18.00]	[1.00 : 1.50]
Silvicola	[5.00 : 12.00]	[6.00 : 10.00]	[1.00 : 2.50]
Californica	[6.00 : 7.00]	[6.00 : 10.00]	[0.60 : 0.80]
Farinosa	[2.50 : 6.50]	[3.00 : 6.50]	[0.30 : 1.00]
Breckonii	[4.00 : 9.00]	[7.00 : 10.00]	[0.90 : 2.00]

3. EM-type algorithm for interval data

The Expectation Maximization(EM) algorithm [9] has been widely applied to estimation problems involving incomplete data, or in problems which can be modeled as mixture of distributions. In brief, the EM algorithm aims at finding maximum likelihood estimates of parameters in probabilistic models in the presence of missing or hidden data. Due to its simplicity, the EM for multivariate gaussian mixture model is by far the most employed mixture model with many applications in cluster analysis and statistical pattern recognition (see, for instance, [10]).

In the outlier framework, the EM algorithm can be employed as a tool for detecting atypical observations from the data sets . For this reason, a EM-type algorithm for interval data (EM-IVD) is introduced in this paper. EM-IVD extends the standard EM algorithm for multivariate gaussian mixture model to treat interval-valued data.

Consider $\mathbf{X}^*$ as a $n \times r$ input data matrix and whose each row is represented as an interval feature vector $\mathbf{x}_i^* = (x_{i1}^*, \ldots, x_{ir}^*)^{'}$ where $x_{ij}^* = [a_{ij}, b_{ij}]$, $(j = 1, \ldots, r) \in \Im = \{[a, b] : a, b \in \Re, a \leq b\}$. The interval Expectation-Maximization (iE-M) algorithm sets an initial partition and alternates two steps such an expected log likelihood-type function reaches a stationary value representing a local maximum.

Let $\{C_1, C_2\}$ be a partition of $\mathbf{X}^*$ in 2 clusters (outliers and inliers) and $\boldsymbol{\theta}_k = (\tau_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{'}$ $(k \in \{1, 2\})$ be a parameter vector of C_k where $\boldsymbol{\mu}_k = ([\mu_{kl}^1, \mu_{ku}^1], \ldots, [\mu_{kl}^r, \mu_{ku}^r])^{'}$ is an average interval vector, $\boldsymbol{\Sigma}_k$ be a covariance matrix and τ_k be a mixture coefficient of C_k. In the iE-M method, there is an average interval vector represented as $\boldsymbol{\mu}_k$ that correspond average values of boundaries of intervals and a single covariance matrix $\boldsymbol{\Sigma}_k$ whose the values measure the variability of the intervals related to this average interval vector.

3.1 Initialization step

Randomly choose 2 different objects $\mathbf{g}_1$ and $\mathbf{g}_2$ belonging to $\mathbf{X}^*$ and assign each objects i to a class C_m such that $m =$ arg $\min_{k=1,2}$ $d(\mathbf{x}_i^*, \mathbf{g}_k)$ where d is the normalized Hausdorff distance [11] between two interval vectors.

Let $\mathbf{x}_i^*$ e $\mathbf{x}_h^*$ two interval vectors in $\Re^r$, the normalized Hausdorff distance between these vectors is given by:

$$d(\mathbf{x}_i^*, \mathbf{x}_h^*) = \left\{ \sum_{j=1}^{r} \left[\frac{Max[\ |a_i^j - a_h^j|, |b_i^j - b_h^j|\ ]}{H_j} \right]^2 \right\}^{1/2}, \tag{1}$$

with

$$H_j^2 = \frac{1}{2n^2} \sum_{i=1}^{n} \sum_{h=1}^{n} \left[Max[\ |a_i^j - a_h^j|, |b_i^j - b_h^j|\ ] \right]^2 .$$

Given a partition $\{C_1, C_2\}$, the initial values for the parameters of the class C_k $(k = 1, 2)$ are computed as:

- average interval vector

$$\hat{\boldsymbol{\mu}}_k = ([\hat{\mu}_{kl}^1, \hat{\mu}_{ku}^1], \ldots, [\hat{\mu}_{kl}^r, \hat{\mu}_{ku}^r])^{'} \tag{2}$$

with

$$\hat{\mu}_{kl}^j = \frac{1}{|C_k|} \sum_{i \in C_k} a_i^j$$

and

$$\hat{\mu}_{ku}^j = \frac{1}{|C_k|} \sum_{i \in C_k} b_i^j.$$

- covariance matrix $\hat{\boldsymbol{\Sigma}}_k = (\hat{\sigma}_k^{vj})$ with

$$\hat{\sigma}_k^{vj} = \frac{\sum_{i,v \in C_k} \left[(a_i^v - \hat{\boldsymbol{\mu}}_{kl}^v)(a_i^j - \hat{\boldsymbol{\mu}}_{kl}^j) + (b_i^v - \hat{\boldsymbol{\mu}}_{ku}^v)(b_i^j - \hat{\boldsymbol{\mu}}_{ku}^j)) \right]}{2|C_k|}. \tag{3}$$

- mixture coefficient

$$\hat{\tau}_c = \frac{|C_k|}{n}. \tag{4}$$

3.2 E step

Let $\hat{\boldsymbol{\mu}}_{kl} = (\hat{\mu}^1_{kl}, \dots, \hat{\mu}^r_{kl})'$ and $\hat{\boldsymbol{\mu}}_{ku} = (\hat{\mu}^1_{ku}, \dots, \hat{\mu}^r_{ku})'$ be vectors associated to lower and upper bounds of the intervals of $\hat{\boldsymbol{\mu}}_k$. Consider also $\mathbf{x}^*_{il} = (a^1_i, \dots, a^p_i)'$ and $\mathbf{x}^*_{iu} = (b^1_i, \dots, b^p_i)'$ as vectors associated to lower and upper bounds of the intervals of the pattern $\mathbf{x}^*_i$ $(i = 1, \dots, n)$.

Given the parameter vector $\hat{\boldsymbol{\theta}}_k = (\hat{\tau}_k, \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k)'$ $(k \in \{1, 2\})$, the probability of the object i belong to C_k is defined as:

$$\hat{Pr}(C_k|\mathbf{x}^*_i) = \frac{\hat{\tau}_k \hat{Pr}(\mathbf{x}^*_i|C_k)}{\sum_{k=1}^2 \hat{\tau}_k \hat{Pr}(\mathbf{x}^*_i|C_k)}, \tag{5}$$

where

$$\hat{Pr}(\mathbf{x}^*_i|C_k) = \frac{\exp^{-\frac{1}{2}[A+B]}}{\sqrt{(2\pi)^p \times |\hat{\boldsymbol{\Sigma}}_k|}}, \tag{6}$$

$$A = (\mathbf{x}^*_{il} - \hat{\boldsymbol{\mu}}_{kl})^r \hat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}^*_{il} - \hat{\boldsymbol{\mu}}_{kl})$$

$$B = (\mathbf{x}^*_{iu} - \hat{\boldsymbol{\mu}}_{ku})^r \hat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}^*_{iu} - \hat{\boldsymbol{\mu}}_{ku})$$

3.3 M step

The parameter vector $\hat{\boldsymbol{\theta}}_k = (\hat{\tau}_k, \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k)'$ $(k \in \{1, 2\})$ is updated by:

$$\hat{\tau}_k = \frac{1}{n} \sum_{i=1}^n \hat{Pr}(C_k|\mathbf{x}^*_i), \tag{7}$$

$$\hat{\boldsymbol{\mu}}_k = ([\hat{\mu}^1_{kl}, \hat{\mu}^1_{ku}], \dots, [\hat{\mu}^r_{kl}, \hat{\mu}^r_{ku}])'$$

with

$$\hat{\mu}^j_{kl} = \frac{\sum_{i \in \Omega} a_{ij} \cdot \hat{Pr}(C_k|\mathbf{x}^*)}{\sum_{i \in \Omega} \hat{Pr}(C_k|\mathbf{x}^*)}, \tag{8}$$

$$\hat{\mu}^j_{ku} = \frac{\sum_{i \in \Omega} b_{ij} \cdot \hat{Pr}(C_k|\mathbf{x}^*)}{\sum_{i \in \Omega} \hat{Pr}(C_k|\mathbf{x}^*)} \tag{9}$$

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\sum_{i \in \Omega} \hat{Pr}(C_k|\mathbf{x}^*) \times (W + V)}{2 \cdot \sum_{i \in \Omega} \hat{Pr}(C_k|\mathbf{x}^*)}, \tag{10}$$

with

$$W = (\mathbf{x}^*_{il} - \hat{\boldsymbol{\mu}}_{ki})(\mathbf{x}^*_{il} - \hat{\boldsymbol{\mu}}_{kl})'$$

$$V = (\mathbf{x}^*_{iu} - \hat{\boldsymbol{\mu}}_{ku})(\mathbf{x}^*_{iu} - \hat{\boldsymbol{\mu}}_{ku})'.$$

3.3.1 Algorithm schema

The iE-M algorithm has the following steps:

Algorithm 1 A EM-type algorithm for interval data.

1: ***Initialization step***: Randomly choose a partition $(C_1^{(0)}, C_2^{(0)})$ of $\mathbf{X}^*$ or randomly choose 2 distinct objects $\mathbf{g}_1^{(0)}, \mathbf{g}_2^{(0)}$ belonging to $\mathbf{X}^*$ and assign each objects i to the closest prototype $\mathbf{g}_m$, where $m = \arg\min_{k=1,2} d(\mathbf{x}^*_i, \mathbf{g}_k)$ and d is the Hausdorff distance defined in Eq. (1). Obtain initial estimate for parameters $\hat{\tau}^0_k, \hat{\boldsymbol{\mu}}^0_k$ and $\hat{\boldsymbol{\Sigma}}^0_k$ $(k = 1, 2)$ according to the Eqs. (2), (3) and (4), respectively. Do $t = 1$.
2: **E-*step***: For $i = 1, \dots, n$, compute the probability $\hat{Pr}(\mathbf{x}^*_i|C_k)^t$ $(k = 1, 2)$ using the Eq. (5).
3: **M-*step***: For $k = 1, 2$, compute the vector $\hat{\boldsymbol{\theta}}^t_k = (\hat{\tau}^t_k, \hat{\boldsymbol{\mu}}^t_k, \hat{\boldsymbol{\Sigma}}^t_k)$ according to the Eqs. (7), (8), (9) and (10).
4: ***Stopping criterion*** If $||\frac{\hat{\boldsymbol{\theta}}^t_k - \hat{\boldsymbol{\theta}}^{t-1}_k}{\hat{\boldsymbol{\theta}}^t_k}|| < \varepsilon$ for $k = 1, 2$ then go to step 5 else do $t = t + 1$ and go to 2.
5: ***Classification step***: For $i = 1, \dots, n$ find the cluster C_{k^*} such that

$$k* = arg \max_{1 \le k \le 2} \hat{Pr}(\mathbf{x}_i|C_k).$$

Let K be the number of classes (here, $K = 2$). The time complexity of the **E-*step*** is $O(nKr^2t)$ and the time complexity of the **M-*step*** is $O(nKt + nKrt + nKr^2t)$. Therefore, the time complexity of the iE-M algorithm is $O(nKr^2t)$.

4. Performance Analysis

For simulated interval data sets 1 and 2, the performance is measured by the false positive and false negative rates (FNR and FPR) in the framework of a Monte Carlo experience with 100 replications for each interval data set. Here, FNR is the number of elements of the inlier class labeled as belonging to outlier class divided by the size of the inlier class and FPR is the number of elements of the outlier class labeled as belonging to inlier class divided by the size of the outlier class.

For each data set, four situations are considered taking into account the quantity (percentage of the data set) of outlying observations presents in each interval data set, that is, 2%, 6%, 10% and 20% of the interval data are indeed interval outliers. Moreover, values for the seed s^c are generated from an uniform $U[1, 10]$ and the values for the parameters β_0, β_1 are selected randomly from an uniform distribution $U[1, 10]$. Each interval data set has two clusters, one with regular intervals and the other with outlying intervals.

Tables 2 shows the the average of the false negative and false positive rates (FNR and FPR). The iE-M method performs well in terms of false positive rate for all cases. Moreover, this method based on the full covariance surpasses that based on the diagonal matrix for both scenarios. This is expected because the linear relation assumed for the interval variables. Regarding the false negative rate, the iE-M method improves when the number of outliers increases and it is important to observe that this method has the worst

Table 2: FPR(%) and FNR (%) for scenarios 1 and 2.

Outliers	Scenario 1				Scenario 2			
	FPR		FNR		FPR		FNR	
	Diag	Full	Diag	Full	Diag	Full	Diag	Full
2%	0.00	0.00	10.90	10.07	0.00	0.00	11.09	13.45
6%	10.34	0.00	1.18	0.83	2.67	0.40	1.62	1.03
10%	18.80	0.00	1.00	0.76	7.00	0.40	1.32	0.92
20%	32.50	0.00	0.65	0.55	20.50	0.10	0.78	0.70

Table 3: Average number of iterations for the iE-M algorithm.

Outliers	Scenario 1		Scenario 2	
	Diagonal Matrix	Full Matrix	Diagonal Matrix	Full Matrix
2%	3.20	3.22	3.53	3.18
6%	4.66	3.13	4.16	3.11
10%	4.48	3.05	4.03	3.04
20%	3.83	3.15	3.46	3.22

performance for the data sets containing a small group of outliers (2% of the data set).

Table 3 shows the average number of iterations for the iE-M algorithm and scenarios 1, 2 and 3. In general, the convergence of this method was achieved with less than five iterations. The algorithm based on full covariance matrix achieves the convergence faster than the algorithm based on diagonal covariance.

With respect the application of the iE-M algorithm to the amanita data set, two groups are obtained. The first group contains 20 species: Lanei, Constricta, Franchetii, Novinupta, Pantherina, Phalloides, Protecta, Vaginata, Velosa, Aprica, Bivolvata, Gemmata, Magniverrucata, Smithiana, Cokeri, Porphyria, Silvicola, Californica, Farinosa and Breckonii. The second group contains 3 species: Muscaria, Ocreata and Pachycolea. From these results and Figure 1 that points out two outliers belonging to the amanita data set, we can say that the Muscaria, Ocreata and Pachycolea species are candidate outliers.

5. Conclude remarks

In this paper, an interval Expectation-maximization for detecting outlier in the framework of regression analysis which is related to symbolic data analysis is presented. The method has as input data a set of predictor interval symbolic variables and a response interval symbolic variable. The experiments regarding different scenarios of simulated interval data sets containing interval outliers and an application with a mushroom interval data base showed the usefulness of this algorithm.

6. Acknowledgements

The authors would like to thank CNPq, CAPES and FACEPE (Brazilian Agencies) for their financial support.

References

[1] Cariou, V. Extension of multivariate regression trees to interval data. Application to electricity load profiling. Computational Statistics 21 325-341, (2006).

[2] Giusti, A. and Grassini L. Cluster analysis of census data using the symbolic data approach. Advances in Data Analysis and Classification, 2 (2) 163-176 DOI: 10.1007/s11634-008-0024-5, (2008).

[3] Pimentel, B.A. & Souza, R.M.C.R. Using Weighted Clustering and Symbolic Data to Evaluate Institutesś Scientific Production. IN Artificial Neural Networks and Machine Learning ICANN 2012. Lecture Notes in Computer Science Volume 7553, 435-442, 2012.

[4] Noirhomme-Fraiture, M. & Diday, E. Symbolic Data Analysis and the SODAS Software, Wiley, (2008).

[5] Bezerra, B.L.D and De Carvalho, F.A.T. Symbolic data analysis tools for recommendation systems Knowledge Information System 26, 3, 385-418, (2011)

[6] Malia, K. & Mitra, S. Symbolic classification, clustering and fuzzy radial basis function network. Fuzzy Sets and Systems 152 553-564, (2005).

[7] Gioia, F. & Lauro, C.N. Principal Component Analysis on Interval Data. Computational Statistics 21, 343-363, (2006).

[8] Lima Neto, E.A. & De Carvalho, F.A.T.. Centre and Range method for fitting a linear regression model to symbolic interval data. Computational Statistics & Data Analysis 52, 1500-1515, (2008).

[9] Dempster, A.P., Laird, N.M. and Rubin, D.B. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, B 39, 1, 1-38, (1977).

[10] He, Yi, Pan, Wei and Lin, Jizhen Lin. Cluster analysis using multivariate normal mixture models to detect differential gene expression with microarray data. Computational Statistics & Data Analysis 51, 2, 641-658, (2006).

[11] Billard, L., Diday E., Symbolic Data Analysis: Conceptual Statistics and Data Mining, Wiley, West Sussex, England, (2006).

Proactive Control of Traffic in Smart Cities

Benjamín Zavala and Germán H. Alférez
Global Software Lab
Facultad de Ingeniería y Tecnología, Universidad de Montemorelos
Montemorelos, N.L., Mexico

Abstract - *The excessive growth of modern cities generates major problems in public administration. One problem is the control of traffic flow during peak hours. In this paper we propose a solution to the problem of vehicular control through a proactive approach based on Machine Learning. Through our solution, a traffic control system learns about the traffic flow in order to prevent future problems of long queues waiting at traffic lights. The traffic system architecture is based on the principles of Autonomic Computing to change the timers of the lights automatically. A simulation of the streets on a smart city and a tool based on Weka were created in order to validate our approach.*

Keywords: Machine Learning, Proactive Control, Traffic, Smart Cities, Autonomic Computing.

1 Introduction

In recent years, the world has experienced mass migration from the countryside to the city. For example, in Mexico today over 70% of the population lives in the city [1]. There is an exponential growth in the population of cities turning them into mega cities. This phenomenon leads to many problems. One of these problems is traffic congestions [2]. Traffic congestions negatively affect the quality of life of citizens by increasing travel time, generating stress and economic losses, and increasing environmental pollution.

The current solutions to control traffic in smart cities wait for an event to happen (i.e., a long queue at a traffic light) to generate an action to try to solve this event. These solutions may be referred to as "reactive". However, we believe that for cities to become truly intelligent, they need "proactive" solutions that anticipate traffic problems and prevent these problems from becoming evident.

In this paper, we present a proactive solution for controlling vehicular traffic in smart cities. Our solution is based on Machine Learning and Autonomic Computing. First, our approach analyzes data traffic levels with Machine Learning. Using this technique of artificial intelligence, the system takes proactive decisions based on historical traffic data. In order to perform autonomous adjustments at traffic lights, the system self-adjusts according to the principles of Autonomic Computing of IBM [3]. A simulation of traffic in a smart city demonstrates the efficiency of our solution. In order to predict the problems that can occur in the future, our solution is based on Weka API [4].

The remainder of this paper is organized as follows. Section II presents a simulation that demonstrates the problem of traffic congestions in big cities. Section III presents related work. Section IV presents the underpinnings of our approach. Section V describes our solution to carry out proactive adaptations of traffic lights. Section VI presents our running prototype and evaluation results. Section VII presents conclusions and future work.

2 Traffic congestions in big cities

In this section we present a simulation of the streets of a city that helps to understand the arising traffic problem. This simulation consists of a vehicular crossing of two unidirectional streets. In order to simplify the simulation, only the change between green and red light signals is simulated. Congestion becomes evident when the amount of vehicles that can cross a traffic light in a particular moment is lower than the amount of arriving cars to the queue.

In the simulation, every traffic light has an assigned time value "x" for red or green. Also, we specified the average time that a car takes to cross an intersection ("y"). Based on these data we calculated how many vehicles could cross the traffic light when it is green. Also, a random value was generated for the vehicles that come to the queue in the traffic light "i".

As seen in Figure 1, the number of vehicles tends to increase linearly as time passes. In our simulation, this trend was generated when "x" takes a value equal to 15 seconds, "i" has values between 0 and 9, and "y" equals to 3 seconds.

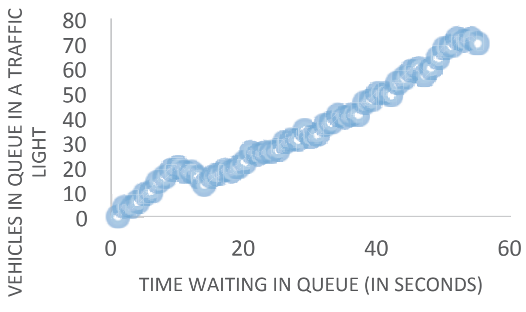


Figure 1. Results of the traffic simulation

3 Related work

Currently fixed-time and sensitive traffic strategies have been implemented for managing traffic flow. These strategies are presented below.

3.1 Fixed time strategies

Fixed time strategies are adjusted for long periods of time where parameters are assumed constant. This approach can be problematic in settings with high variability demands or with usual presence of non-standard conditions (e.g. accidents, riots, or unexpected events).

For example, SIGSET is based on traffic flow patterns at an intersection traffic light. This is a well-known system for traffic engineers [5]. SIGSET works in isolation at each intersection and assigns fixed times at traffic lights.

3.2 Sensitive traffic strategies

Sensitive traffic control strategies execute their logic based on traffic measurements performed in real time at the entrances of the intersections. In order to perform measurements, it is necessary to have some type of traffic detectors.

Within traffic sensitive methods, there are two reactive methods that solve problems when they are already evident. On one hand, there are approaches that detect the presence of heavy traffic at an intersection and change traffic light timers to give preference to the direction of greatest activity. On the other hand, there are adaptive solutions with networks of traffic lights. Plans are implemented together in these networks in order to optimize traffic flow [6].

Another sensitive strategy is a proactive one. For instance, RHODES [7] takes as input data of real-time measurement of traffic flow. Then, it controls the flow through a network. The system uses a control architecture that decomposes the traffic control problem into several sub-problems that are interconnected in a hierarchical fashion and predicts traffic flows at appropriate resolution levels (individual vehicles and platoons). This approach has several optimization modules for solving hierarchical sub-problems. Also, it uses a data structure and communication approaches to reach fast solutions of sub-problems. RHODES depends on a central control module. We argue that a more decentralize approach could be used to distribute calculations on site. In this way, the costs and complexity related to infrastructure communication could be reduced.

4 Underpinnings of our approach

The solution proposed in this paper is intelligent, autonomous, and proactive. These underlying concepts are described below.

4.1 Machine Learning

Machine Learning is a term used to encompass a wide variety of techniques for discovering patterns and relationships in sets of data. The primary goal of any Machine Learning algorithm is to discover meaningful relationships in a set of training data and produce a generalization of these relationships that can be used to interpret new, unseen data [8]. Among Machine Learning methods, we can find forecasting. Forecasting is the process of making statements about events whose actual outcomes have not yet been observed [9].

4.2 Autonomic Computing

Inspired by biology, Autonomic Computing has evolved as a discipline to create self-managing software to overcome the complexities to maintain systems effectively. Autonomic Computing covers the broad spectrum of computing in domains as diverse as mobile devices and home-automation, thereby demonstrating its feasibility and value by automating tasks such as installation, healing, and updating. Since doing manual adaptations is a difficult (and sometimes impossible) task, our approach is based on IBM's reference model for autonomic control loops (which is sometimes called the MAPE-K loop) [10].

4.3 Proactive Adaptations

On one hand, reactive adaptations are performed in response to an event. On the other hand, proactive adaptations take an action in advance (i.e., before an incident negatively impacts the system) [11]. Reactive adaptation mechanisms may cause increases in the execution time and financial loss, which can lead to user and business dissatisfaction [12]. Proactive approaches try to solve these problems by detecting the need for an adaptation before the problem is evident.

5 Our approach

In this section we present our solution to carry out proactive adaptation of traffic lights. The structural blocks of our approach are based on the components of the MAPE-K loop, i.e., Monitor, Analyze, Plan, Execute, and Knowledge (see Figure 2).

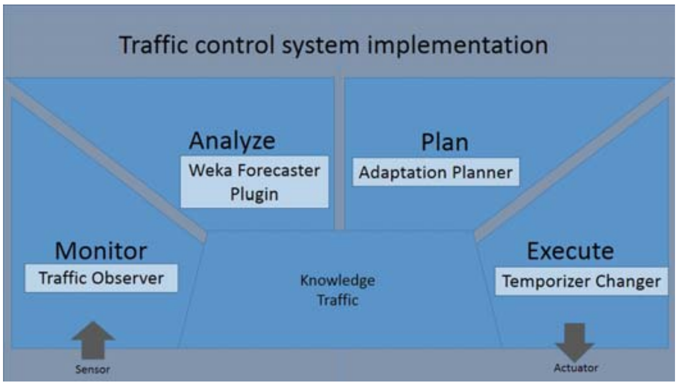


Figure 2. MAPE-K cycle of our approach

In our approach, sensors collect traffic data in the Analysis component. This data is used for training our Machine Learning approach. The task of observing the traffic is in charge of the Traffic Observer, which detects violations of any Service Level Agreement (SLA) at specific times. A violation of the SLA becomes evident when the number of cars in queue is greater than the desired SLA. After completing training, the Weka Forecasting plugin analyzes the data collected and predicts potential traffic problems. Then, the Adaptation Planner plans the necessary changes to the traffic light timers in order to prevent traffic problems proactively.

Finally, the Temporizer Changer makes the necessary changes in the traffic light timers trough actuators.

Our solution is described in the following subsections based on the MAPE-K components. Our approach is exemplified based on the traffic simulation described in Section II.

5.1 Monitor

Monitoring involves capturing properties of the context that are meaningful to the self-properties of the system. In our case, we are interested in observing the traffic. To this end, we propose the Traffic Observer, which is a tool that observes the traffic through sensors. In our case, the Traffic Observer periodically checks the activity in the traffic simulation described in Section II with simulated sensors.

At the start of the simulation, we create two traffic lights. A status, either green or red, was assigned to each one of them. Also, a random number of vehicles that come into the queue was generated. In order to simplify the simulation, the preventive stage of a traffic light, which is usually yellow, was included within the green phase.

The Traffic Observer detects if the number of vehicles queued at a certain moment at the traffic light is greater than the SLA (i.e., a violation of the SLA). If so, this event is saved in a log with .arff extension. Files with this extension have the required format to carry out forecasting in Weka. Specifically, the .arff file format requires that the data has a special header, such as the one in the following example:

```
@relation A @attribute seconds numeric
@attribute cars numeric @data 60, 7
```

First of all, there is a relationship of data (@relation A), and this relationship has two attributes, both of type numeric: seconds (@attribute seconds) and cars (@attribute cars). After this, the problematic event (i.e., a violation of the SLA) data is recorded (@data 60, 7). In the example above there is a violation of the SLA after 60 second of execution. At this time, there are 7 vehicles in queue, when the SLA indicates that the maximum number of cars in queue should be 3 cars.

5.2 Analysis

The objective of this phase is to detect, in a proactive manner, the traffic problems that may occur according to the results captured in the log generated by the Traffic Observer. In order to accomplish the prediction of traffic problems, we use the Weka Forecasting plugin. This plugin can load or import a time series forecasting model and use it to generate a forecast for future time steps beyond the end of incoming historical data [13]. Among the forecasting methods available in Weka, we chose the Multilayer Perceptron due to its lowest percentage of error after completing tests with different methods (Gaussian Process, Kernel Regression, Linear Regression, Multilayer Perceptron, and SMOreg).

In order to run the forecaster, the following parameters are required: type of data to predict, measure of time for training data, and the number of times to predict (e.g. the epochs). In our case, the forecaster predicts the number of vehicles that will arrive at the traffic light after training. The data generated by the forecaster is stored in a text file.

For example, a possible training (i.e., the observation of the traffic for a period of time) concluded in the second 1,800 of the simulation execution. During this time, 25 violations of the SLA were found. With this data, predictions are performed. Specifically, the forecaster predicts that if the traffic lights keep operating with the current value of their timers, then there will be a greater number of vehicles than those specified in the SLA in the second 1,860. Through forecasting, it is possible to predict problems ahead based on historical data.

5.3 Planning

The objective of this phase is to plan how to automatically solve the traffic problems predicted in the analysis phase. Specifically, in this stage the traffic light timers are recalculated in order to avoid traffic problems.

During planning, the file created at the analysis phase is read. Each entry in this file is a problem to solve (i.e., a violation of the SLA). In this phase, we propose the Adaptation Planner. This tool performs the following steps to plan a change in traffic light timers:

1. The Adaptation Planner keeps in a variable the text recovered from the file generated in the previous phase. For example, the Adaptation Planner takes the data of 35 vehicles in queue that will violate the SLA at some future time (after training) according to forecasting.

2. The Adaptation Planner performs the following operation:

   *newTimer = (int) (d * tCross);*

   The "d" variable is the number of cars that could violate the SLA according to forecasting. The "tCross" variable indicates the average time that a vehicle takes to pass the intersection. The "newTimer" variable is the new traffic light timer. The result of the operation is transformed to integer.

3. The value of the "newTimer" variable, which corresponds to the solution to a problem (i.e., a possible violation of the SLA), is saved. For example, a vehicle takes around 3 seconds to cross the intersection (tCross) and there are 35 expected vehicles that will arrive at the traffic light (d). Therefore, the result of *newTimer* is 105. It means that 35 vehicles take 105 seconds to cross the intersection.

Each of the values calculated in this phase corresponds to the solution of the problems referred to in the analysis phase.

5.4 Execution

The objective of this phase consists of making changes in the timers of the traffic lights according to the results of the planning phase. The idea in this phase is to prevent the materialization of predicted problems. Actuators are in charge of making changes in traffic light temporizers. In our case, actuators are simulated. These actuators take the new time for a traffic light ("newTimer" in the previous phase), and modify the traffic light timer with that time.

For example, the Adaptation Planner in the planning phase discovered that it takes 105 seconds for 35 vehicles to pass through the intersection. Therefore, at this stage our Temporizer Changer assigns 105 seconds to the timer of the traffic light. As a result, the 35 cars will have enough time to cross.

6 Running prototype

This section describes the prototype that has been created to demonstrate our approach. The prototype was created in Java and Weka, which can be accessed through Java [14].

The prototype GUI is divided into three areas (see Figure 3). The parameters under which the traffic control is performed are displayed in Area #1. Specifically, the following parameters are determined in this area: traffic light timer, execution time of the traffic simulation, crossing time, and expected SLA.

The text fields in Area #2 show the SLA violations in each traffic light (Traffic Light A and Traffic Light B). The first column indicates the time in seconds in which the violation of the SLA happened. The second column indicates the amount of cars with which the SLA was violated. Area #3 shows how many times the SLA at every traffic light is violated before and after the implementation of Machine Learning.

The prototype is trained according to the parameters in Area #1 of Figure 3. During training, the number of times that the SLA was violated is pretty high at each traffic light (e.g. 59 violations at the traffic light A and 57 violations at the traffic light B). Then, by using forecasting the number of SLA violations descends dramatically (17 violations at the traffic light A and 26 violations in traffic light B). Incidents of violation of the SLA are not fully eliminated but decrease greatly.

Figure 4 shows how Machine Learning dramatically decreased the SLA violations in our running prototype. The execution of Machine Learning occurs in the second 1,800. From this moment it is possible to see how the incidences of violations of SLAs are kept at a very low level.

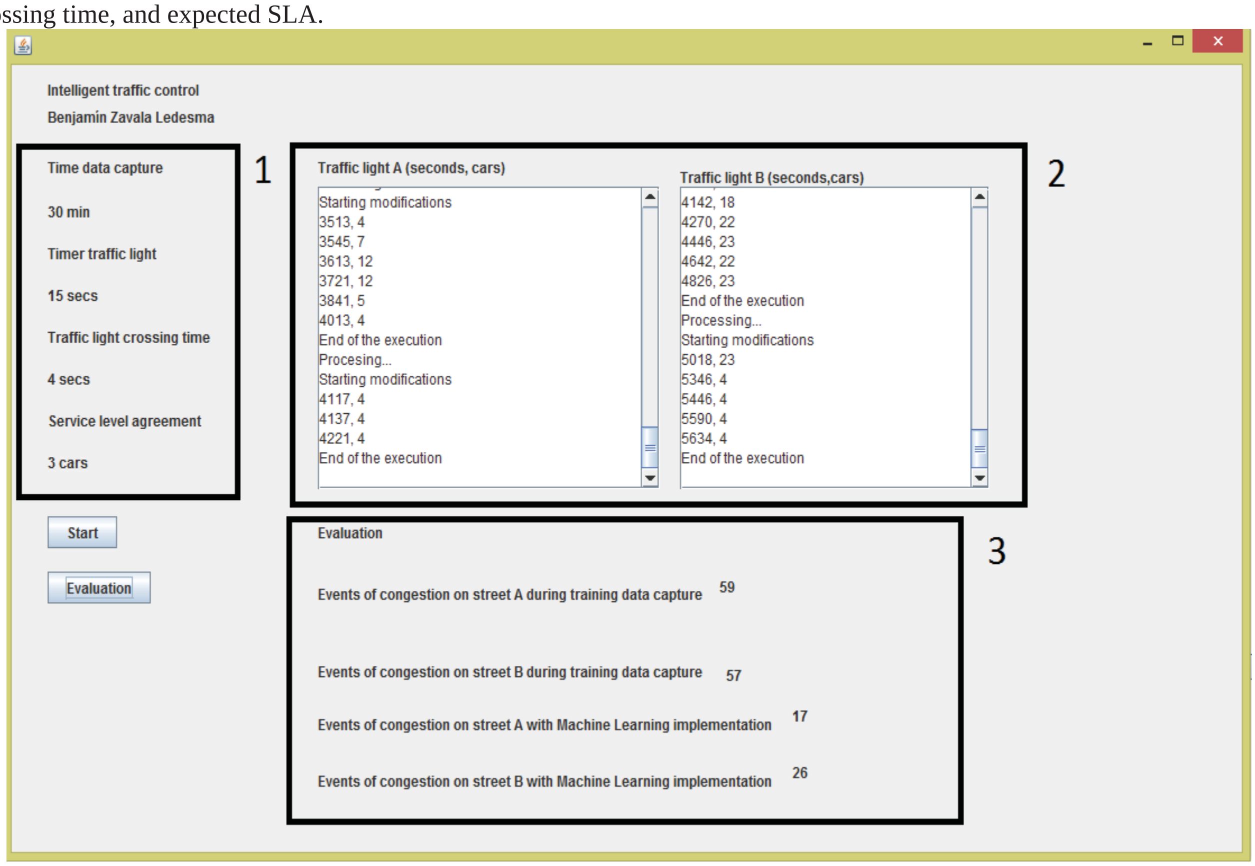


Figure 3. Implementation of the prototype

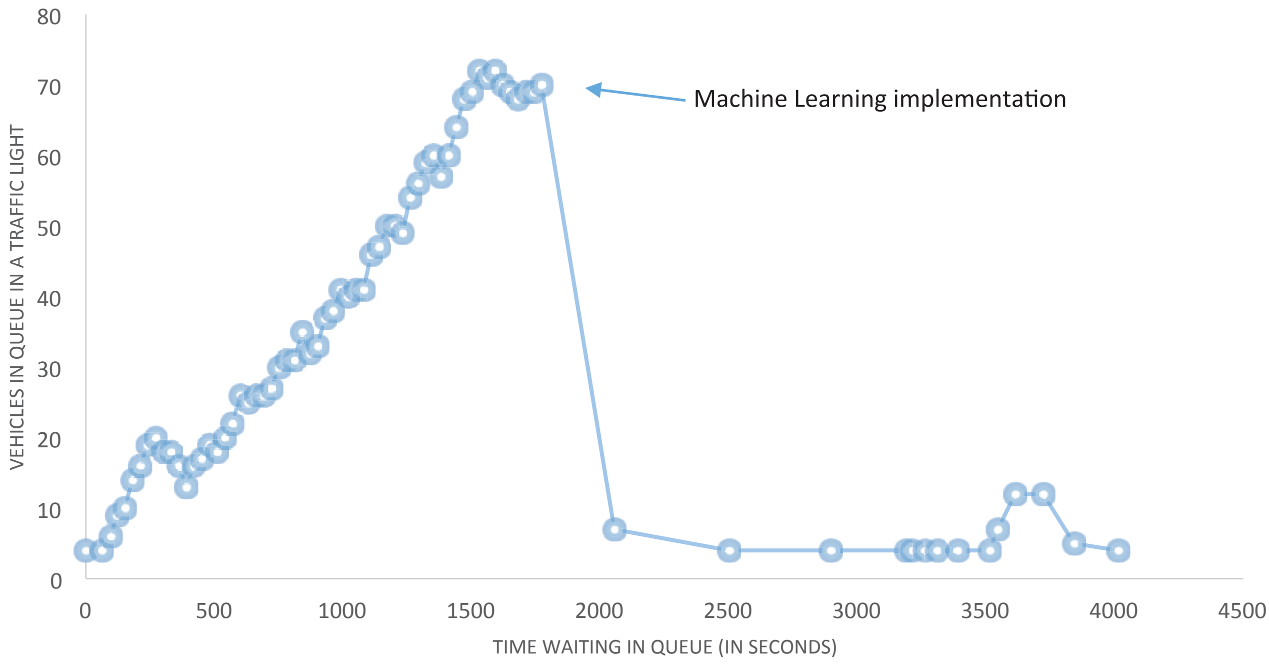


Figure 4. Application of Machine Learning in the second 1,800 of our traffic simulation

7 Conclusions and future work

This paper proposed a proactive solution to traffic control by means of Machine Learning and the principles of Autonomic Computing. Our solution could offer several benefits for the development of mega cities: 1) economic losses can be avoided by allowing that a cargo or a worker arrive in the shortest possible time to their destination; 2) pollution can be reduced; and 3) the number of vehicle accidents in rush hours can be reduced. These accidents may be caused by the stress produced in traffic jams [15].

As future work, we will implement a computer vision module for live traffic control through cameras. The Traffic Observer will collect this data. We will also develop a mobile application to help users make queries about the status of the traffic and find the most optimal path between the starting point and the destination point by getting data in real time from our autonomic solution.

8 References

[1] United Nations. (2015). UNFPA – Country programmes and related matters [Online]. Available: https://data.unfpa.org/docs/mex

[2] SIEMENS. (2015). *Siemens traffic solutions*. Available: https://www.swe.siemens.com/spain/web/es/industry/mobility/Documents/traffic.pdf

[3] IBM. (2006). "An architectural blueprint for autonomic computing," *IBM White Paper*.

[4] Pentaho. (2015). *Time Series Analysis and Forecasting with Weka*. Available: http://wiki.pentaho.com/display/DATAMINING/Time+Series+Analysis+and+Forecasting+with+Weka

[5] R. E. Allsop. (1981). "Computer program SIGSET for calculating delay-minimissing traffic signal timings description and manual for users". University College London, Transport Studies Group.

[6] D. Robles, P. Ñañez, and N. Quijano. (2009). "Control y simulación de tráfico urbano en Colombia: Estado del arte," *Revista de ingeniería*, pp. 59-69.

[7] P. Mirchandani and L. Head. (2001). "A real-time traffic signal control system: architecture, algorithms, and analysis," *Transportation Research Part C: Emerging Technologies*, vol. 9, pp. 415-432.

[8] S. Mitchell. (1995). "The application of machine learning techniques to time-series data"

[9] J. S. Armstrong. (2001). *Principles of forecasting: a handbook for researchers and practitioners* vol. 30: Springer Science & Business Media.

[10] C. Ayora, V. Torres, V. Pelechano, and G. H. Alférez. (2012). "Applying CVL to business process variability management," in *Proceedings of the VARiability for You Workshop: Variability Modeling Made Useful for Everyone*, pp. 26-31.

[11] A. Metzger. (2011) "Towards accurate failure prediction for the proactive adaptation of service-oriented systems," in *Proceedings of the 8th workshop on Assurances for self-adaptive systems*.

[12] R. R. Aschoff and A. Zisman. (2012). "Proactive adaptation of service composition," presented at the Proceedings of the 7th International Symposium on

Software Engineering for Adaptive and Self-Managing Systems, Zurich, Switzerland.

[13] Pentaho. (2013). *Using the Weka Forecasting Plugin*. Available: http://wiki.pentaho.com/display/DATAMINING/Using+the+Weka+Forecasting+Plugin

[14] Pentaho. (2013). *Weka 3: Data Mining Software in Java*. Available: http://www.cs.waikato.ac.nz/ml/weka/

[15] Urbeconomica. (2015). "Accidentes vehiculares," vol. 2015, ed. Puebla: Redacción Urbeconómica.

SentAMaL- A Sentiment Analysis Machine Learning Stock Predictive Model

SA Bogle[1], and WD Potter[2]
[1]Computer Science Department, University of Georgia, Athens, Georgia, USA
[2] Artificial Intelligence Institute, University of Georgia, Athens, Georgia, USA,

Abstract - *Social media comments have in the past had an instantaneous effect on stock markets. This paper investigates the sentiments expressed on the social media platform Twitter and their predictive impact on the Jamaica Stock Exchange. A hybrid predictive model of sentiment analysis and machine learning algorithms including decision trees, neural networks and support vector machines are used to predict the Jamaica Stock Exchange. The architecture created, SentAMaL, investigated the impact of sentiments on medical marijuana legalization on relevant stock indices. Due to the unstructured nature of tweets, a customized pre-processing routine was developed prior to determining sentiment and to perform the prediction. Experimental results show 87% accuracy in the movement prediction and 0.99 correlation coefficient for price prediction.*

Keywords: *sentiment analysis, stock prediction, pre-processing*

1 Introduction

Social media comments have in the past had a rapid effect on stock markets [1], [2]. This paper describes the use of sentiment analysis and machine learning (ML) techniques applied to social media comments coupled with historical stock data to predict the Jamaica Stock Exchange (JSE) *in the short term.* It builds on an earlier work [3] which gained 90% accuracy in the movement prediction and 0.95 correlation coefficient for price prediction on applying machine learning approaches to the JSE.

1.1 Background

The JSE was incorporated as a private limited company in August 1968 and the stock market began operations six months later in 1969. While the JSE is not considered a major market, it has been defined by Standard & Poor's as a frontier market. Supervised learning algorithms such as Support Vector Machine (SVM) and Artificial Neural Network(ANN) are more accurate than generic statistical models such as regression and presents a more accurate stock prediction model on the JSE dataset [3].

1.2 Contribution

Social media is inherently assistive in predicting stock trading volumes since it captures the views of many within the population and tweets and posts often go viral in very miniature increments of time. While studies such as [4], [5] and [6] show that volume shifts can be correlated with price movements, text mining prediction studies have not generally focused on the regression problem of predicting prices. Also, while social data have been used to predict economical outcomes in studies such as [7], these predictions are not in the context of financial markets.

2 Literature Review

This section outlines the research to date on sentiment analysis from sources such as news and social media data and use of semantic web architectures for stock prediction. It also compares these approaches with traditional non-sentiment based ML methods.

2.1 News Analysis for Stock Prediction

Wuthrich's group [8] analyzed news articles, collected from five popular financial websites, available before the opening of the Hong Kong stock market with text mining techniques including k-nearest-neighbor and a variety of neural networks. In predicting the trend of the Hong Kong market, they achieved an average accuracy of 46%, which proved better than the accuracy of a random predictor, which achieved a maximum 33% accuracy. News Categorization and Trading System (NewsCATS), was designed to predict stock price trends for the time immediately after the publication of press releases and achieved an average weighted recall of 54% [9]. These accuracy values are low in comparison to most ML-based studies which report between 80%-90% accuracy on binary predictions of market trends.

Wang et al [10] developed an ontology for knowledge about news in financial instrument markets and suggested the framework can be used as input to stock price prediction algorithms. In their later work, [11], which utilized the previous ontology, an expert reasoning system was designed to integrate the domain knowledge in the data mining process through building data mining models consisting of multi news variables with certain financial instrument trading activity and suggesting the potential polar ("positive", "neural", "negative") effect of each news variable, on trading activities.

However, no results were presented regarding accuracy of actual price predictions based on the model.

The AzFinText system [12] investigated whether subjectivity and objectivity impacted stock news article prediction. Based on sentiment analysis it was found that subjective news articles were easier to predict in price direction by 9%, while articles with a negative sentiment were easiest to predict by 1%.

Most web mining or ontology based approaches do not report accuracy of predictions, nor error rates, which is typical of statistical and machine learning papers. This makes the comparison between the models challenging as standard benchmarks are necessary for comparative analysis. However, a conceptual or qualitative analysis is given in the proceeding section since quantitative measures are not available from most semantic related financial forecasting studies.

2.2 The Case for Social Media Input

Markets will react quicker based on the efficient market hypothesis since the investor knowledge and intentions are publicized on the web. This is evident by a false tweet, when the Associated Press Twitter account was hacked on April 23rd, 2013, which alluded to two explosions at the White House and President Obama being hurt. This caused the market to plunge within minutes [1], [2]. Fortunately however, it reacted after the Associated Press denounced the false tweet. Although the USA market is strong form efficient based on the efficient market hypothesis, the market did not correct itself momentarily after the information went public. This is critical with electronic trading as it could have resulted in market crashes since trades are executed within nanoseconds and several minutes passed before it was corrected. CNN reported that built in circuits to facilitate 'trading halts,' failed to execute. According to Subrahmanyam [13]: "under reasonable conditions, discretionary [or] randomized, trading halts may be less susceptible than rule-based halts to 'gravitational' or 'magnet' effects which occur when traders concerned about an impending closure accelerate their orders (p.1)". The real time impact of the semantic web model can serve as an informant to markets so they can reflect factors with immediate impact.

One of the advantages of sentiment based models is that they consider qualitative factors which are missing from most ML models in the literature. However, the proposed approach in section 3 would consider qualitative factors similar to the prior knowledge artificial neural networks (ANN) used by [14] and the qualitative ANN approach used by [15]. In [15] a functional link ANN architecture was used for both long and short term stock forecasting, which utilized a standard least mean square algorithm with search-then-converge scheduling to compute a learning rate parameter that changes temporally and required less training experiments. Kohara et al [14] focused on the impact of qualitative factors including social and economic change and was less rigorous on the quantitative factors. The model proposed in section 3 includes a mix of qualitative and quantitative factors.

As evidenced by the instantaneous impact of the false tweet, [16] concurs that semantic stock prediction models provide identification of early warnings of financial systemic risk, based on the activity of users of the WWW and the query volume dynamics that emerges from the collective but seemingly uncoordinated activity of many users.

In [17], variants of dynamic social network analysis were used to predict movie stock values on the Hollywood Stock Exchange (HSX). They predicted the daily changes in prices and explored the effectiveness of sentiment analysis and web matrices in predicting trends. No explicit measure of accuracy or value was given for predicting stock price. After examining the nature of messages, [18] found that Web talk does not predict stock movements, although it is a good predictor of volatility.

2.3 Semantic Web Architectures

A semantic web approach to establish a correlation between daily trading volumes of stocks traded and volumes of queries related to the same stocks in the NASDAQ-100 was used in [16]. An OWL based application, Stockwatcher [19], tracked relevant news items on the NASDAQ-100 and predicted one of three possible effects it will have on the company: positive, negative or neutral. The news items were extracted from RSS feeds. The Stockwatcher architecture utilizes natural language processing (NLP) and text mining techniques such as tagging and morphologic analysis.

A generic news based stock prediction system using tagging and classification is outlined in [20]. Feature selection and features weighting are performed on the categorized news and the weighted vectors inputted to the classifier. A survey of eight developed classifier systems were compared. The classifiers used were SVM (or a SVM-variant) and decision trees. The datasets ranged from one month to eight years. The most accurate classifier among that survey was decision trees with 82% on a three month dataset in the system developed by [21]. SVM had a directional accuracy of 70% on a 15 month dataset used by [22].

2.4 Comparison of Machine Learning and Sentiment based Web Approaches

Sentiment and semantic web based prediction models may be more accurate in predicting trends than actual prices. Compared to non-sentiment based approaches, ML approaches have gained superior accuracy in predicting several stock attributes: trend, price and volumes. For example if a company is at the brink of crisis, such as: the loss of a law suit, explosion on the compounds or is approaching bankruptcy; this is likely to be spread rapidly by social media and traders using the semantic model would be able to react more quickly by selling off the stock than if they relied on a traditional non-sentiment based ML based predictor.

In summary, sentiment and semantic knowledge based approaches may be able to offer faster predictions than traditional ML based approaches as well as give a good indication of volatility in the markets and predict potential trading volumes with reasonable accuracy. Although their learning or processing time is slower than semantic approaches, ML based approaches are better able to handle regression problems and give a good prediction not just on volume traded or market trends, but also on stock prices which is key to determining potential profits.

The semantic web knowledge based approach heavily incorporates human perceptions and inklings and is similar to fuzzy based reasoning on degrees of uncertainty. Unlike other ML approaches that take a non-human like approach to learning focusing mainly on the numbers, the semantic web KB approach favorably considers non-quantitative factors which often have a quick and direct impact on trading.

The traditional ML models are likely to outperform the semantic model if there are no major social or economic changes, since as lengthy supervised learning approaches they place greater emphasis on the relations between quantitative factors over a time series. The semantic web models are likely to be more accurate in short term forecasting whilst ML approaches, especially those that incorporate qualitative factors, may be more accurate on long term forecasts. If ML hybrid approaches were to include the real time inputs from the semantic approach, then the accuracy is expected to improve, as long as it is not given a false positive.

2.5 Summary

Sentiment based models seem better suited for short term forecasts while machine learning approaches will likely outperform them on long term forecasts. As discussed, a variety of architectures emerged within the last eight years on sentiment analysis and semantic web to predict stocks. However, except for the AzFinText system, most have not focused on price prediction. The next section will focus on the experimental and architectural design.

3 Experimental Design

This section details the research design used. A hybrid (qualitative and quantitative) research approach was employed in this study. The architectural design of SentAMaL is outlined in Figure 1. Machine learning algorithms are used both to classify sentiments from social media mining as well as to predict stocks based on qualitative and quantitative inputs, denoted by the shaded objects.

The qualitative data acquisition involves collecting data from the Twitter social network using a customized software developed in the open source R programming language and the Twitter application programming interface (API). Of note, the data on Twitter was fairly readily available through a connection to its API barring the restrictions on its API. A function was created and used to extract tweets from relevant Twitter timelines using hashtags, for example #marijuana, screen names (@tvj and keywords, for example, “weed” and “legalize”.

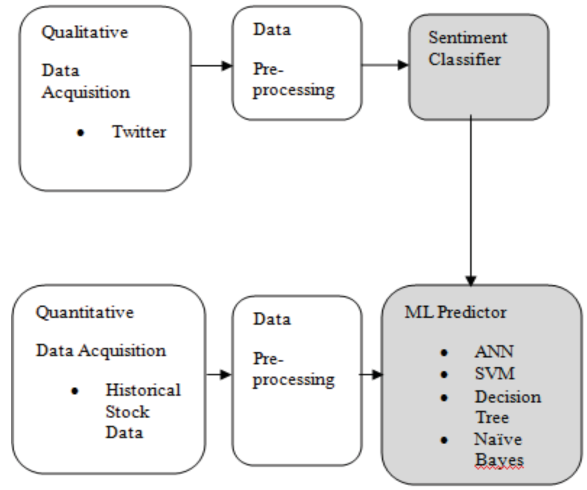


Figure 1: Architectural Diagram of SentAMaL

3.1 Cleaning of Tweets

Figure 2 shows snippets of code used to perform qualitative data pre-processing. The procedures used in this study compared three machine learning classifiers by utilizing a supervised classification approach. The classifier that was best suited for the problem was used to analyze the tweets obtained from Twitter between January and February 2015, in order to determine the sentiments being expressed about marijuana and its legalization.

The qualitative data pre-processing involves removal of duplicate tweets, numbers, punctuation and symbols. A pre-processing function developed in R, allowed for the initial filtering of unwanted or unnecessary verbiage from the tweets, while being extracted from Twitter. It also included replacing certain emoticons with words (e.g. :-) with “happy”). Figure 3 shows samples of tweets cleaned using the R programming application. The tweets downloaded were stored as comma separated values (.csv) files for further processing.

Normalizing Tweets

The normalizing tweets function used in a spreadsheet application, entailed the removal of duplicate tweets. This was done by comparing two rows of tweets at a time to determine whether they were similar. If so, then a user-defined identifier was used to mark one of them and this record was subsequently removed from the dataset.

The quantitative aspect of the analysis was realized using data classification tools that quantified and classified data instances based on the sentiments expressed in each tweet. The analysis was conducted predominantly based on the establishment of sentiment polarity (positive, negative or neutral) of the tweets.

The quantitative data acquisition involves historical data from S&P 500, NASDAQ and JSE over the trading period that the tweets were acquired.

3.2 Population and Sample

The population for the study was comprised of a corpus of approximately 1941 tweets that were extracted from Twitter pages between January and February 2015.

```
clean.text = function(x)
{
  # to lower
  x = tolower(x)
  # remove rt
  x = gsub("rt", "", x)
  # remove at
  x = gsub("@\\w+", "", x)
  # remove punctuation
  x = gsub("[[:punct:]]", "", x)
  #
  x = gsub('[[:cntrl:]]', '', x)
  #
  x = gsub('\\d+', '', x)
  # remove numbers
  x = gsub("[[:digit:]]", "", x)
  # remove links http
  x = gsub("http\\w+", "", x)
  # remove tabs
  x = gsub("[ |\t]{2,}", "", x)
  # remove blank spaces at the beginning
  x = gsub("^ ", "", x)
  # remove blank spaces at the end
  x = gsub(" $", "", x)
```

Figure 2: Tweet Cleaning Function Code Snippet

utech receives machine to advance medical marijuana research
jamaica legalizes medical marijuana amp decriminalizes all weed
legalize jamaica legalizes marijuana jamaica
rt weedfeed jamaica legalized medical marijuana on bob marleys birthday
jamaica legalized medical marijuana on bob marleys birthday
rt whaxyapp jamaica legalized medical marijuana amp decriminalized possession on bob marleys bdayâ€
jamaica legalized medical marijuana amp decriminalized possession on bob marleys bday€
jamaica legalizes medical marijuana amp decriminalizes all weed
medical marijuana could be jamaicaâ€™s economic legacy says businessman joe issa theradioshow itsyourlifestyle
florida and pennsylvania work on new medical marijuana bills and jamaica makes history on bob marleys birthday

Figure 3: Samples of Tweets Cleaned using R Programming Application

This population represented a sample of tweets expressing polarities of possible positive, negative or neutral sentiments about the legalization of medical marijuana. A stratified random sampling method was used to select approximately1000 tweets from the entire corpus for building the classification models. From this, ten fold cross validation was used to classify the dataset.

Of the training dataset, the positive tweets were approximately six times less than those deemed to be negative or neutral. In order to balance the polarity representation, a Synthetic Minority Oversampling Technique (SMOTE) filter was applied to stabilize the unbalanced data instances with synthetic data. Thus, a more equitable classification model was expected to be derived for use in determining the sentiment polarity of new data instances.

3.3 Data Pre-processing

Further pre-processing of the tweets was conducted to remove values not captured by the R tweet cleaner. This involved conversion of the data string into vectors of words by applying a *StringToWordVector* filter. This filter converts string attributes into a set of attributes representing word occurrence (depending on the tokenizer). It also sets parameters that were relevant to the information being retrieved. Such parameters include limits placed on the number of terms repeated (term frequency); the number of words to output (word count); the tokenizer which delimits words within the string; and stemmer that facilitates conversion of terms to their base forms, for example, the base term *love* for the words like lovely, lovable, loving.

3.3.1 Sentiment Classification

In order to determine sentiment of tweets, several machine learning classifiers were evaluated to identify the data mining classification model that is best suited for the problem. Three machine learning classifiers were explored : the Naive Bayes Multinomial Text Classifier, the Support Vector Machine (SVM), and the J48 Decision Tree, which are all said to work well on text categorization.

After training the three classification models to correctly categorize the tweets into positive, negative and neutral classes they were explored to validate their accuracy as well as their efficiency. Naive Bayes Multinomial Text emerged as the best performing model derived from these classifiers and was applied to unknown instances of tweets that were extracted from Twitter.

4 Results

Table I shows the non-sentiment based price prediction of all companies that traded on the JSE within the two month period while Table II shows the price prediction of only the five relevant companies that are perceived would be impacted by sentiments derived from the marijuana tweets. These companies are distributors of medical, pharmaceutical or tobacco products denoted on the JSE by the symbols: MDS, LASM, LASD, JP and CAR.

Table III shows movement prediction of the various indices of the various indices using SentAMaL. The number of instances for the seven indices combined was 2233, while each of the seven indices had 68 instances. While ANN had a slighter higher accuracy for than SVM for all the indices tested, the error was also marginally higher. This denotes the robustness of the SVM classification technique. Decision Trees also proved the superior binary classifier among the three.

Table I
Price Prediction of All Companies on the JSE

	Multilayer Perceptron	SVM
CC	0.9924	0.9954
MAE	3.0385	1.3479
RMSE	5.3307	4.1644
RAE	20.9405 %	9.2894 %
RRSE	12.3317 %	9.6337 %

Total Number of Instances 3547

KEY
CC-Correlation coefficient
MAE- Mean absolute error
RMSE-Root mean squared error
RAE-Relative absolute error
RRSE-Root relative squared error

Table II
Price Prediction of Drug Related Companies on the JSE

	SVM SentAMaL	SVM without SentAMaL
CC	0.9993	.9994
MAE	0.2577	0.2631
RMSE	0.6466	0.6004
RAE	1.7819 %	1.8214
RRSE	3.8413 %	3.563 %

Total Number of Instances 169

5 Conclusion

It is likely that there was not a great difference in values between SentAMaL and its counterpart because the contents of the tweet corpus did not contain shocking content or a newsflash as in the case of the false tweet which temporarily threatened the stability of the US market [2]. Although SentAMaL receives a similar correlation coefficient to its non-sentiment based counterpart, its error is significantly lower than the purely quantitative model. This indicates the accuracy of the SentAMaL model in using sentiment for its qualitative input to complement the traditional quantitative input of most ML stock forecasting models.

6 Acknowledgement

The authors acknowledge the use of the JSE dataset available on their website at http://www.jamstockex.com/ Investor centre => downloads.

Table III
SentAMaL Movement prediction of various JSE indices : Main JSE index(1), JSE Select (2), All Jamaican Composite (3), Cross Listed Index(4), Junior Market Index(5), Combined Index (6) and US Equities Index (7)

Index	**Scheme**	**M**	**Accuracy (%)**
All*	DT	0.09	99.7
All*	ANN	0.076	89
All*	SVM	0.251	87
1	DT	0.089	97
1	ANN	0.167	86.7
1	SVM	0.132	86.7
2	DT	0.014	98.5
2	ANN	0.15	92.6
2	SVM	0.088	91.1
3	DT	0.029	97
3	ANN	0.164	89.7
3	SVM	0.117	88.2
4	DT	0.022	95
4	ANN	0.117	85.2
4	SVM	0.025	84.1
5	DT	0.02	97
5	ANN	0.107	89.7
5	SVM	0.251	88.2
6	DT	0	100
6	ANN	0.192	88.2
6	SVM	0.102	89.7
7	DT	0.01	98.5
7	ANN	0.11	86.7
7	SVM	0.268	82.3

KEY
*All refers to all seven indices
M -Mean absolute error

7 References

[1] Blaine, C.(2013) *Stocks drop on false tweet of White House explosion* Retrieved 4/23/2013 from http://money.msn.com/top-stocks/post.aspx?post=a4b8243d-94ec-432b-ae75-750f850240f

[2] CNN, (2013) *Stocks plunge after twitter account hacked* Retrieved 4/23/2013 from http://www.cnn.com/video/#/video/us/2013/04/23/nr-stocks-plunge-after-ap-twitter-acct-hacked.cnn

[3] Bogle, S.A. and Potter, W.D. (2015) A Machine Learning Predictive Model for the Jamaica Frontier Market *Proceedings of the 2015 Int'l Conference of Data Mining and Knowledge Engineering* by IAENG(ISBN978-888-19253-4-3)

[4] Podobnik B, Horvatic D, Petersen A, Stanley H E (2009) Cross-correlations between volume change and price change. Proceedings National Academy Science USA 106: 22079–22084.

[5] Plerou V, Gopikrishnan P, Rosenow B, Amaral L, Stanley H E (2000) Econophysics: financial time series from a statistical physics point of view. Physica A 279: 443–456.

[6] Yamasaki K, Muchnik L, Havlin S, Bunde A, Stanley H E (2005) Scaling and memory in volatility return intervals in financial markets. Proceedings National Academy Science USA 102: 9424–9428.

[7] Asur, S., & Huberman, B. A. (2010). Predicting the future with social media. *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on 1* 492-499.

[8] Wuthrich, B., Cho, V., Leung, S., Permunetilleke, D., Sankaran, K., & Zhang, J. (1998). Daily stock market forecast from textual web data. *Systems, Man, and Cybernetics, 1998. 1998IEEE International Conference on, , 3* 2720-2725.

[9] Mittermayer, M. (2004). Forecasting intraday stock price trends with text mining techniques. *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on,* 10 pp.

[10] Wang, S., Zhe, Z., Kang, Y., Wang, H., & Chen, X. (2008). An ontology for causal relationships between news and financial instruments. *Expert Systems with Applications, 35*(3), 569-580.

[11] Wang, S., Xu, K., Liu, L., Fang, B., Liao, S., & Wang, H. (2011). An ontology based framework for mining dependence relationships between news and financial instruments. *Expert Systems with Applications, 38*(10), 12044-12050

[12] Schumaker, R. P., Zhang, Y., Huang, C., & Chen, H. (2012). Evaluating sentiment in financial news articles. *Decision Support Systems, 53*(3), 458-464.

[13]Subrahmanyam, A. (1995). On rules versus discretion in procedures to halt trade. *Journal of Economics and Business, 47*(1), 1-16.

[14] Kohara, K., Ishikawa, T., Fukuhara, Y., & Nakamura, Y. (1997). Stock price prediction using prior knowledge and neural networks. *Intelligent Systems in Accounting, Finance and Management, 6*(1), 11-22.

[15] Padhiary, P. K., & Mishra, A. P. (2011). Development of improved artificial neural network model for stock market prediction. *International Journal of Engineering Science, 3*

[16] Bordino, I., Battiston, S., Caldarelli, G., Cristelli, M., Ukkonen, A., & Weber, I. (2012). Web search queries can predict stock market volumes. *PloS One, 7*(7), e40014.

[17] Doshi, L., Krauss, J., Nann, S., & Gloor, P. (2010). Predicting movie prices through dynamic social network analysis. *Procedia-Social and Behavioral Sciences, 2*(4), 6423-6433.

[18] Antweiler, W., M. Frank. 2004. Is all that talk just noise? The information content of Internet stock message boards. J. Finance 59(3) 1259–1295.

[19] Micu, A., Mast, L., Milea, V., Frasincar, F., & Kaymak, U. (2009). Financial news analysis using a semantic web approach. *Semantic Knowledge Management: An Ontology-Based Framework,* , 311-328.

[20] Nikfarjam, A., Emadzadeh, E., & Muthaiyah, S. (2010). Text mining approaches for stock market prediction. *Computer and Automation Engineering (ICCAE), 2010 the 2nd International Conference on, , 4* 256-260.

[21]Rachlin, Last, Alberg, and Kandel, 2007 "ADMIRAL: A Data Mining Based Financial Trading System," *2007 IEEE Symposium on Computational Intelligence and Data Mining*, pp. 720-725.

[22] Zhai, Hsu, and Halgamuge,2007 "Combining News and Technical Indicators in Daily Stock Price Trends Prediction," *Lecture Notes in Computer Science*, pp. 1087-1096.

Supervised Potentiality Actualization Learning for Improving Generalization Performance

Ryotaro Kamimura
IT Education Cente and Graduate School of Science and Technology, Tokai Univerisity
1117 Kitakaname, Hiratsuka, Kanagawa 259-1292, Japan
ryo@keyaki.cc.u-tokai.ac.jp

Abstract— *The present paper proposes an application of potentiality learning to supervised learning. The potentiality has been developed as a measure of the importance of components in the self-organizing maps (SOM) to extract important input neurons. The main characteristics lies in its simplicity and thus it can be easily implemented. If it is possible to use it for conventional supervised learning, better performance can be expected with much simpler computational method. The potentiality is defined by the variance of input neurons and it is incorporated into supervised learning. Using the potentiality inside, two data sets were used to evaluate the performance. The results show that the potentiality method outperformed ones without it and other conventional methods in terms of generalization performance.*

Keywords: Potentiality, selective potentiality, determination, actualization, generalization

1. Introduction

1.1 Potentiality and Its Actualization

Neural networks have been applied to many problems with better performance than that by the conventional statistical methods. Though the performance of neural networks has been improved, it can be said that the potentiality of components of neural networks cannot be fully explored [1] . The potentiality is considered as the implicit capability of neural networks. The potentiality can be actualized or realized in terms of a number of different forms. For example, the potentiality is realized as the properties of components which can be used to interpret network behaviors or to improve generalization performance. One of the main problems is that little attempts have been made to determine the main potentiality of components of neural networks.

In the present paper, the simple potential method is proposed with two main characteristics, namely, variance and separation. First, the potentiality is supposed to be represented in the form of variance of connection weights. The potentiality is considered to be higher when the neurons respond to input patterns as differently as possible. Second, the potentiality determination and use phase are separated. There have been many attempts to interpret network behaviors and to improve generalization [2], [3], [4], [5], [6] [7], [8], [9]. One of the main difficulties inherent to those approaches is that the errors between targets and actual outputs are minimized and simultaneously generalization performance is improved or interpretation is improved. Error minimization and performance improvement are sometimes contradictory to each other. For example, to have more interpretable networks, internal representations should be simplified as much as possible, which may degrade the performance of neural networks. To overcome those problems, a new method is proposed, where potentiality determination and actualization phase are completely separated. For example, the potentiality is determined roughly and then this potentiality is incorporated into the process of error minimization. Then, contradiction between error minimization and potentiality determination is minimized.

1.2 Relations to the Input Neuron Selection

To demonstrate the potentiality method, the method is applied to the detection of important input neurons (variables) [10], [11], [12], [13], [14] . The variable selection has played important roles in improving the performance of neural networks. In particular, in application, the interpretation of input variables is necessary. However, in this interpretation, neural networks are said to be weaker than the conventional methods such as the regression analyses. The regression analysis has been used in many practical problems, because the coefficients obtained by the regression analysis can be interpreted, though the actual generalization performance is much weaker.

To have more interpretable input variables or input neurons, the potentiality is introduced. The potentiality is defined as the capability of neurons responding to input patterns as differently as possible. Thus, the potentiality is defined as a variance of connection weights. Because the potentiality is an abstract concept, it can be actualized. In the potentiality actualization phase, the potentiality is actualized so as to represent the importance of input neurons.

1.3 Outline

Section 2 introduces the potentiality in the supervised learning. The method is composed of two phase. First, the potentiality determination phases is applied to detect the important input neurons with higher potentiality. Then, the potentiality is normalized and the corresponding connection weights are modified. Then, the final fine tuning phase is performed. In Section 3, the method was applied to the two data sets. In both sets, generalization performance was improved by the potentiality.

2. Theory and Computational Methods

2.1 Potentiality Actualization Learning

The potential actualization learning aims to determine the potentiality of neurons and actualize its potentiality as much as possible. As mentioned, in the potentiality method, the determination of the potentiality and its actual use is separated to facilitate learning. The computational procedure is composed of two phases, namely, potentiality determination and actualization phase in Figure 1. In the potentiality determination phase, the potentiality of neurons is determined by using the variance of connection weights. Then, connection weights are given into the potentiality actualization phase as initial connection ones. In addition, connection weights are weighted by the relative potentiality to take into account the importance or potential importance of input neurons as shown in Figure 1(b). Thus, in the potential actualization phase, connection weights are actually updated to take into account the potentiality and realize or actualize potentiality.

2.2 Individual Potentiality

For this, it is needed to define the potentiality of individual input neurons. The potentiality of an input neuron is defined by

$$v_k = \exp\left(R \sum_{j=1}^{M} (w_{jk} - w_k)^2 \right), \tag{1}$$

where w_{jk} denote connection weights from the kth input neuron to the jth hidden neuron and

$$w_k = \frac{1}{M} \sum_{j=1}^{M} w_{jk}. \tag{2}$$

The coefficient R determines the intensity of the variation of connection weights and should be experimentally determined. The potentiality is based on the variance of input neurons toward output neurons. It is natural to suppose that when input neurons respond to output neurons with large variation, the input neurons surely play important roles. This means that the neurons with large variation have high potentiality to represent input patterns. In addition, by the exponential function, when the variation of neurons becomes larger, the expected potentiality increases exponentially or excessively. This property is needed to intensify a few number of important neurons.

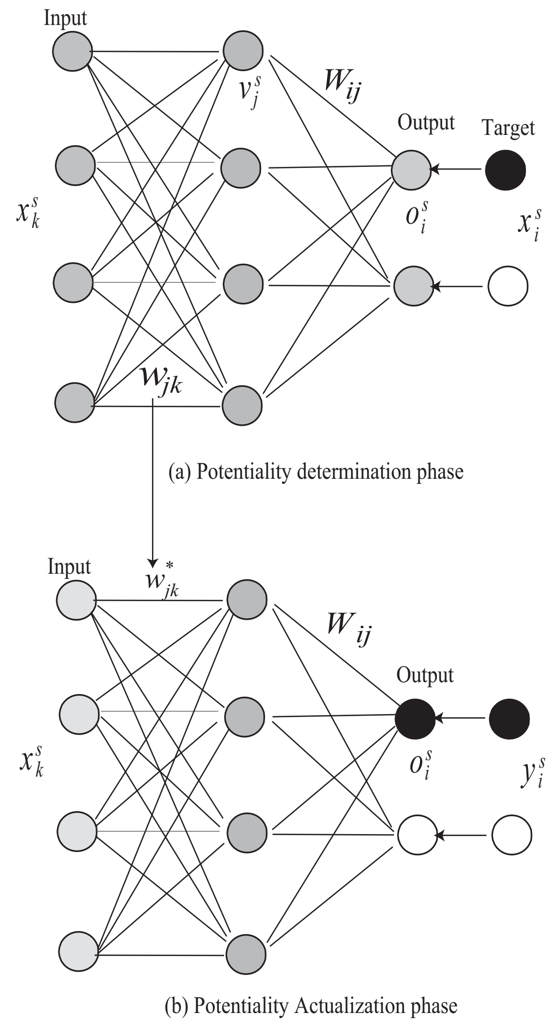


Fig. 1: Network architecture with the potentiality determination (a) and actualization (b) phase.

2.3 Selective Potentiality

The selective potentiality is defined by using the concept of information in the information-theoretic methods [15], [16], [17], [18], [19]. When the information increases in competitive learning, only one neuron finally fires, while all the other neurons cease to do so. This concept of information-theoretic competitive learning is directly translated into the potentiality. When the selective potentiality increases, finally only one neuron tend to have the maximum potentiality.

For using the information theoretic concepts, it is needed to normalize the individual potentiality

$$p(k) = \frac{v_k}{\sum_{l=1}^{L} v_l}. \tag{3}$$

The selective potentiality is defined by the decrease from maximum uncertainty to observed uncertainty

$$I \quad = \quad 1 + \frac{\sum_{k=1}^{L} p(k) \log p(k)}{\log L}. \tag{4}$$

When this potentiality increases, a smaller number of input neurons tend to have larger individual potentiality.

2.4 Potentiality Determination and Actualization Phase

The method is composed of the potentiality determination and actualization phase. In the determination phase, after finishing the learning, the potentiality is computed with the parameter

$$R = \frac{r}{L-1}, \tag{5}$$

where L and r denote the number of input neurons and the learning parameter.

Then, the relative potentiality is computed and with this potentiality, the potentiality actualization is initialized

$${}^{new}w_{jk} = {}^{old}w_{jk}p(k). \tag{6}$$

With these connection weights, the errors between targets and outputs are minimized

$$E = \frac{1}{2S}\sum_{s=1}^{S}\sum_{i=1}^{N}(y_i^s - o_i^s)^2, \tag{7}$$

where S and N denote the number of input patterns and output neurons, and y_i^s are the targets for the outputs o_i^s[1]. This means that the potentiality is incorporated into the leaning processes as initial weights. The experiments results show that the graduate decent learning is much affected by the initial conditions and this method is simple and effective to take into account the potentiality.

3. Results and Discussion

3.1 German Credit Approval Data Set

The first data set is the German credit data set from the machine learning database. The number of input patterns was 1000 with 24 input variables [20].

3.1.1 Selective Potentiality Increase

Figure 2 shows the selective potentiality as a function of the parameter r. As shown in the figure, the selective potentiality increased gradually when the parameter r increased.

Figure 3 show the relative potentiality when the parameter r increased from 1.0(a) to 5.0(h). When the parameter r is 1.0 in Figure 3(a), the relative potentiality distributed almost uniformly. Then, when the parameter r increased from 1.2 (b) to 1.6 (d), the potentiality became gradually differentiated. Then, the parameter increased further from 2.5 (e) to 5.0 (h), several input neurons tended to have much higher relative potentialities.

[1]The hidden and output activation function were the the hyperbolic tangent sigmoid and linear one and the early stopping method was used.

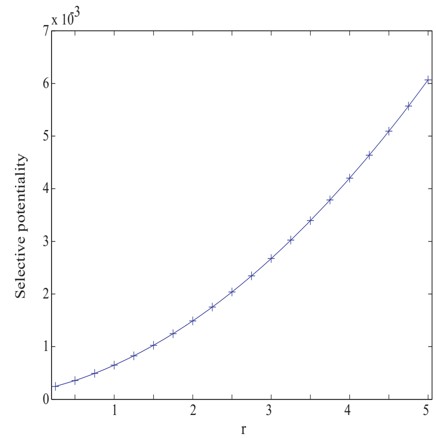


Fig. 2: Selective potentiality as a function of the parameter r for the German credit data set.

Table 1: Summary of experimental results of generalization for the German data set with ten different runs.

Methods	R	Average	Std dev	Min	Max
Potential	4.4	**0.2187**	0.0340	**0.1600**	**0.2600**
Early stopping		0.2367	**0.0273**	0.1867	0.2667
SVM		0.2613	0.0344	0.2067	0.3133
Logistic		0.2313	0.0308	0.1733	0.2733

3.1.2 Generalization Performance

Figure 4(a) shows generalization errors as a function of the parameter r. When the parameter r increased or the selective potentiality increased, the generalization errors tended to decrease and seem to reach the stable states. Figure 4(b) shows the standard deviation of the generalization errors. One of the main characteristics is that the standard deviation increased when the parameter increased. This means that the generalization errors fluctuated when the parameter R increased.

3.1.3 Summary of Results

Table 1 shows the summary of experimental results related to the generalization performance. In the table, the values in bold faces show the minimum values. As can be seen in the table, except the standard deviation, the potential method shows the best performance with the minimum values in the average, minimum and maximum values. On the other hand, the standard deviation was the largest by the potential method. As pointed out in the previous section, the standard deviation tended to be larger by the potentiality method.

Experimental results confirm that generalization performance was improved by increasing the potentiality but the errors tended to fluctuate for the larger parameter values.

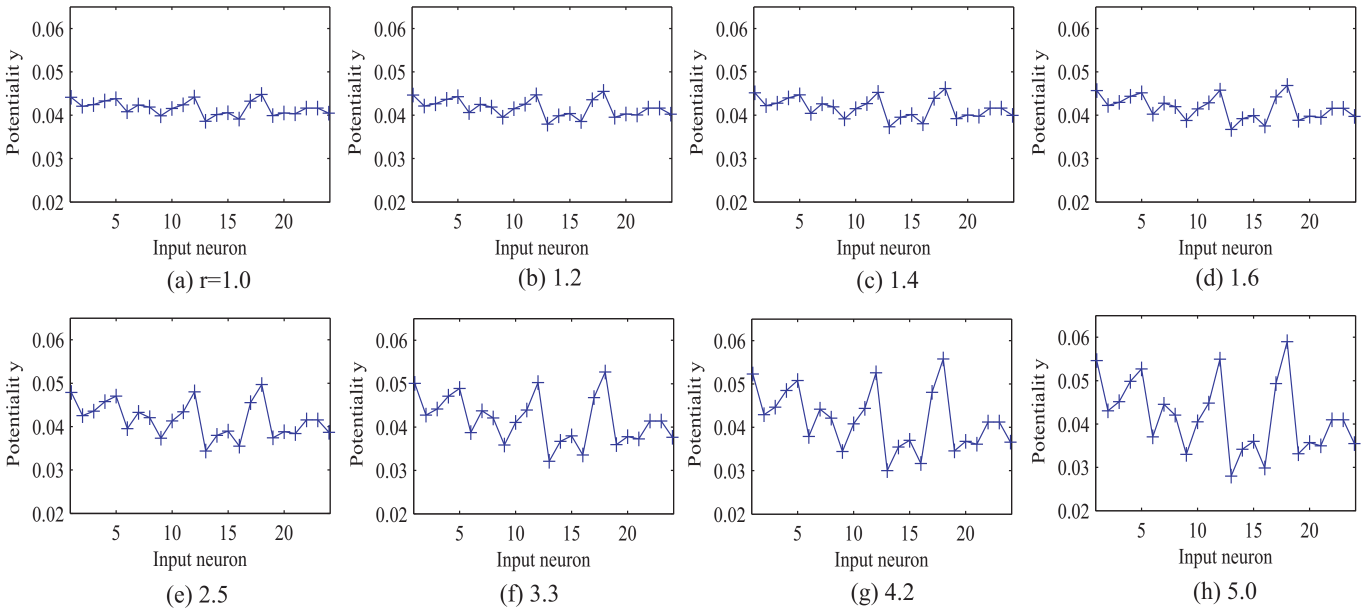


Fig. 3: Potentiality $p(k)$ of input neurons for four input neurons for the German credit data set.

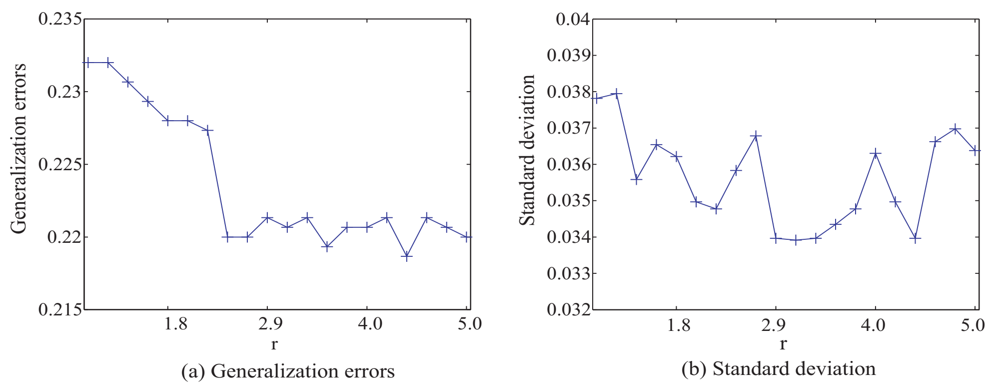


Fig. 4: Generalization errors (a) and the standard deviation of the errors (b) by the potentiality method for the German credit data set.

3.2 Biodegradation Data Set

The second data set is also from the machine learning data set where 41 attributes and 1055 patterns, which must be classified into 2 classes (ready and not ready biodegradable) [20].

3.2.1 Selective Potentiality

Figure 5 shows the selective potentiality as a function of the parameter r. The selective potentiality increased gradually when the parameter r increased.

Figure 6 shows the relative potentiality when the parameter r increased from 1.0(a) to 5.0(h). When the parameter r was 1.0 in Figure 6(a), the potentiality was almost uniform. Then, when the parameter r increased gradually, several potentialities became larger. Finally, when the parameter r was 5.0 in Figure 6(h), some potentialities were clearly differentiated.

3.2.2 Generalization Performance

Figure 7(a) shows generalization errors as a function of the parameter r. The generalization errors decreased for the smaller values of the parameter and then fluctuated. Figure 7(b) shows the standard deviation of the generalization errors. As can be seen in the figure, the standard deviation decreased gradually for the smaller parameter values. Then, the standard deviation became larger when the values became larger.

These results seem to suggest that the potentiality is not related to the improved generalization performance as shown in Figure 7(a) and 5. This can be explained by seeing the standard deviation of generalization errors. When the parameter R increased, the generalization errors fluctuated

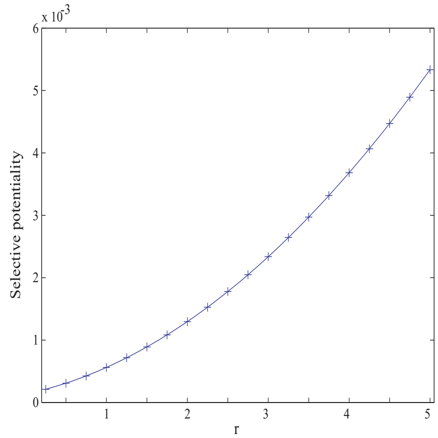


Fig. 5: Selective potentiality as a function of the parameter r for the bio-degeneration data set.

when the parameter R was larger in Figure 5(a). However, the standard deviation in Figure 7(b) greatly fluctuated when the parameter r became larger. This large standard deviation surely affected the overall generalization performance.

3.2.3 Summary of Results

Table 2 shows the summary of experimental results. The potentiality method showed the best performance in terms of the average and maximum errors. On the other hand, for the minimum errors, the logistic regression method showed the best performance and the potentiality method showed the second best performance. The potentiality method had the second largest standard deviation.

The experimental results also show that the present method of potentiality is good at improving generalization performance. The good performance is explained by two points, namely, the effectiveness of potentiality and separation of two phases. First, the potentiality as the variance of input neurons is effective in improving the generalization performance. When neurons respond to input patterns as differently as possible, the neurons play very important roles in learning. For example, naturally, neurons, responding only uniformly to input patterns, are considered to be unimportant. Second, in the method, the potentiality determination and use phase were separated. Only when the potentiality is determined, it is used in learning. This separation contributes to the improved performance.

4. Conclusion

The present paper proposes a new type of learning called "potentiality actualization learning". The potentiality implies the potentiality of input neurons, which is supposed to be realized in many different forms. In this paper, the potentiality is represented in terms of the variance of input neurons. The learning is conducted to realize this potentiality

Table 2: Summary of experimental results of generalization for the bio-degeneration data set with ten different runs. The logistic function was used to normalize the data.

Methods	R	Average	Std dev	Min	Max
Potential	3.7	**0.1184**	0.0221	0.0696	**0.1456**
Early stopping		0.1215	0.0206	0.0823	**0.1456**
SVM		0.1234	**0.0192**	0.0886	0.1519
Logistic		0.1316	0.0303	**0.0633**	0.1646

of input neurons. The potentiality actualization learning is composed of two phases. In the first phase of potentiality determination, the potentiality is determined. In the second phase of the potentiality actualization, the learning is conducted, incorporating the information on the potentiality.

The method was applied to two data sets, namely, German credit approval data set and bio-degradation data set. In both cases, the potentiality could be increased by changing the parameter. In addition, generalization performance was improved. Comparing with those by the other conventional methods like the SVM, performance was better. However, the standard deviation of the generalization errors tended to be larger than that by the other methods. If it is possible to reduce this large standard deviation by some methods, the generalization by the present method can be more improved. Thus, it is needed to develop a method to stabilize learning processes for the potentiality actualization learning.

References

[1] R. Kamimura and R. Kitajima, "Selective potentiality maximization for input neuron selection in self-organizing maps," in *Proceedings of IJCNN 2015 (to appear)*, IEEE, 2015.

[2] L. I. Nord and S. P. Jacobsson, "A novel method for examination of the variable contribution to computational neural network models," *Chemometrics and Intelligent Laboratory Systems*, vol. 44, pp. 153–160, 1998.

[3] A. Micheli, A. Sperduti, and A. Starita, "Analysis of the internal representations developed by neural networks for structures applied to quantitative structure-activity relationship studies of benzodiazepines," *Journal of Chemical Information and Computer Sciences*, vol. 41, pp. 202–218, 2001.

[4] G. G. Towell and J. W. Shavlik, "Extracting refined rules from knowledge-based neural networks," *Machine learning*, vol. 13, pp. 71–101, 1993.

[5] M. Ishikawa, "Structural learning with forgetting," *Neural Networks*, vol. 9, no. 3, pp. 509–521, 1996.

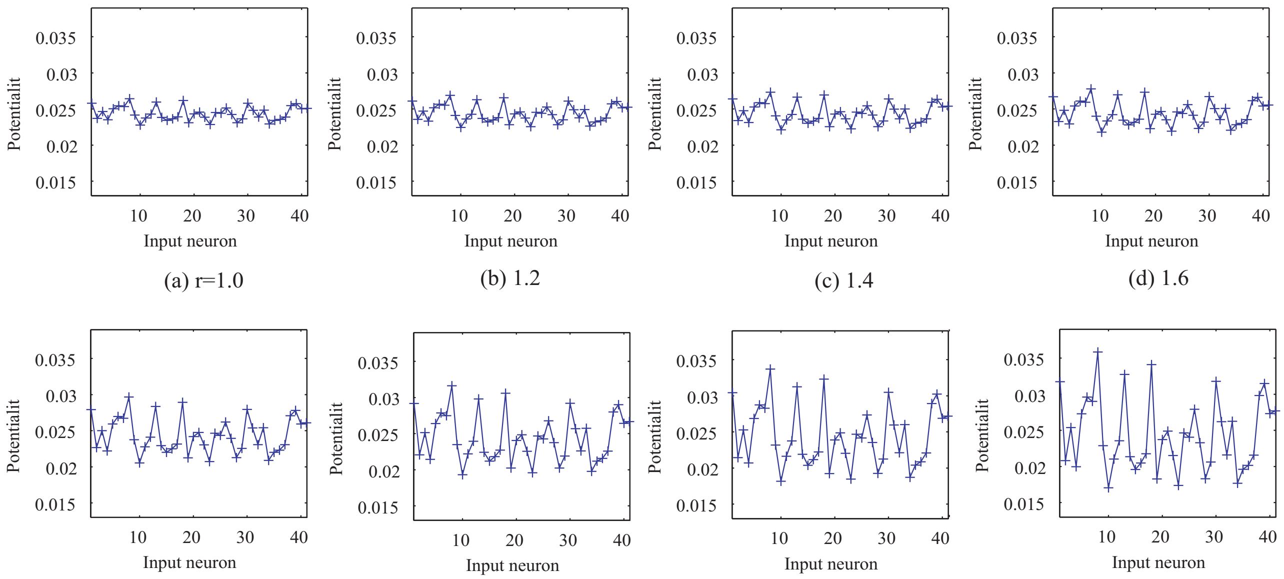
[6] M. Ishikawa, "Rule extraction by successive regularization," *Neural Networks*, vol. 13, pp. 1171–1183, 2000.

[7] P. Howes and N. Crook, "Using input parameter influences to support the decisions of feedforward neural networks," *Neurocomputing*, vol. 24, p. 1999, 1999.

[8] R. Feraud and F. Clerot, "A methodology to explain neural network classification," *Neural Networks*, vol. 15, pp. 237–246, 2002.

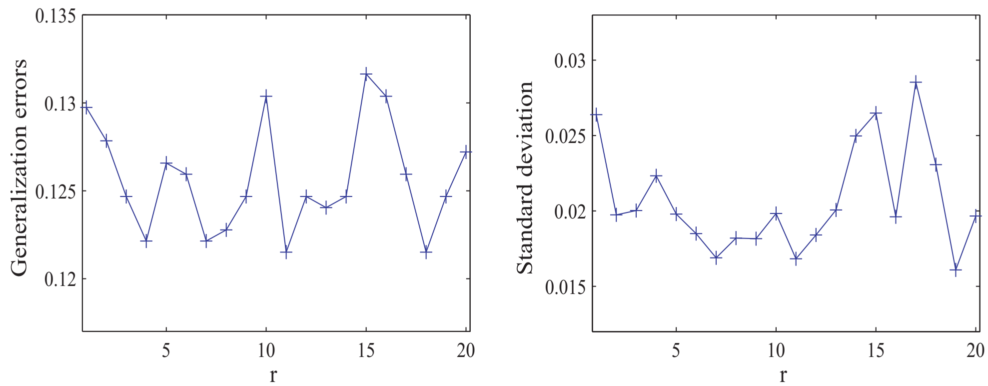
[9] R. Setiono, W. K. Leow, and J. M. Zurada, "Extracting of rules from artificial neural networks for nonlinear regression," *IEEE Transactions on Neural Networks*, vol. 13, no. 3, pp. 564–577, 2002.

[10] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.



(a) r=1.0 (b) 1.2 (c) 1.4 (d) 1.6 (e) 2.5 (f) 3.3 (g) 4.2 (h) 5.0

Fig. 6: Potentiality $p(k)$ of input neurons for four input neurons for the bio-degeneration data set.



(a) Generalization errors (b) Standard deviation

Fig. 7: Generalization errors (a) and the standard deviation of the errors (b), by the potentiality method for the bio-degeneration data set.

[11] A. Rakotomamonjy, "Variable selection using SVM-based criteria," *Journal of Machine Learning Research*, vol. 3, pp. 1357–1370, 2003.

[12] S. Perkins, K. Lacker, and J. Theiler, "Grafting: Fast, incremental feature selection by gradient descent in function space," *Journal of Machine Learning Research*, vol. 3, pp. 1333–1356, 2003.

[13] J. Reunanen, "Overfitting in making comparison between variable selection methods," *Journal of Machine Learning Research*, vol. 3, pp. 1371–1382, 2003.

[14] G. Castellano and A. M. Fanelli, "Variable selection using neural-network models," *Neurocomputing*, vol. 31, pp. 1–13, 1999.

[15] R. Kamimura and T. Kamimura, "Structural information and linguistic rule extraction," in *Proceedings of ICONIP-2000*, pp. 720–726, 2000.

[16] R. Kamimura, "Information-theoretic competitive learning with inverse Euclidean distance output units," *Neural Processing Letters*, vol. 18, pp. 163–184, 2003.

[17] R. Kamimura, "Teacher-directed learning: information-theoretic competitive learning in supervised multi-layered networks," *Connection Science*, vol. 15, pp. 117–140, 2003.

[18] R. Kamimura, "Progressive feature extraction by greedy network-growing algorithm," *Complex Systems*, vol. 14, no. 2, pp. 127–153, 2003.

[19] R. Kamimura, "Information theoretic competitive learning in self-adaptive multi-layered networks," *Connection Science*, vol. 13, no. 4, pp. 323–347, 2003.

[20] K. Bache and M. Lichman, "UCI machine learning repository," 2013.

A Cluster-based Algorithm for Anomaly Detection in Time Series Using Mahalanobis Distance

Erick Giovani Sperandio Nascimento[1a], Orivaldo de Lira Tavares[1], and Alberto Ferreira De Souza[1]
Erick@lcad.inf.ufes.br, tavares@inf.ufes.br, alberto@lcad.inf.ufes.br
[1] Department of Informatics, Federal University of Espirito Santo, Vitória, ES, Brazil
[a] Corresponding Author

Abstract - *We propose an unsupervised learning algorithm for anomaly detection in time series data, based on clustering techniques, using the Mahalanobis distance function. After a brief review of the main and recent contributions made in this research field, a formal and detailed description of the algorithm is presented, followed by a discussion on how to set its parameters. In order to evaluate its effectiveness, it was applied to a real case, and its results were compared with another technique that targets the same problem. The obtained results suggest that this proposal can be successfully applied to detect anomaly in time series.*

Keywords: time series, anomaly detection, clustering, unsupervised learning, mahalanobis distance, pattern recognition

1 Introduction

Nowadays, many processes, such as industrial plants, meteorological monitoring stations or stock markets, generate relevant time series data continuously. In general, these data are collected and stored by specific hardware, and later are analyzed and maintained by specialized people, who learn about these processes using the data, but are also responsible for verifying its correctness in representing the real state of the processes.

In many situations, it is critical for the process to identify unusual patterns that could be generated by unexpected behavior. And such unwanted behavior may be due to any problem that the related process might be experiencing. For example, an industry may monitor some variables of its current productive process to diagnose bottlenecks, violations of quality requisites, violation of environmental requisites such as a specific pollutant emitted to the environment over the permitted by law, or any other situation that could be harmful to its business. Another example is one certain environmental institute or agency, whom would need to monitor some meteorological or air quality parameters in order to evaluate the air quality of an urban area, might experience that some equipment were presenting failure, which could lead to misunderstand data monitoring. Either, a credit card company may monitor each user transaction to look for unusual behaviors that could point to fraudulent operations. These unusual, unwanted behaviors are often called as *anomalous* behaviors, and might be induced in the data due to a variety of reasons, all of them presenting a certain degree of importance to the analyst. And it is important that this analysis could take into account any changes in the parameter's behavior to identify opportunities to improve, prevent or correct any situation.

In this context we present an unsupervised learning algorithm based in clustering techniques using the Mahalanobis distance as its distance function targeting the problem of anomaly detection in time series, here called as *C-AMDATS*, which stands for *Cluster-based Algorithm using Mahalanobis distance for Detection of Anomalies in Time Series*. The paper is organized as follows: the remainder of this section presents a brief review of the recent research regarding anomaly detection in time series. Section 2 presents the foundations of the algorithm, and a detailed and formal description of the algorithm. Section 3 presents a real case with anomalous patterns that was evaluated in order to assess the ability of the *C-AMDATS* approach to detect these anomalies, in conjunction with a comparison with other technique targeting the same problem, i.e., the detection of anomalies in time series. Section 4 presents a conclusion and recommendations for future works. And in Section 5 we acknowledge our main contributors.

1.1 Related Work

Several works have been developed to identify patterns in time series data, and some of them were specialized to detect anomalous patterns in time series. We will briefly present a review about the most recent works in anomaly detection in time series in order to identify whether our proposed technique is introducing a new contribution to the community.

Some works uses distance-based techniques, like in [2,3,4,5,6,7,8] to detect outliers or anomalies in time series. Other works uses sliding windows and discretization techniques. In some cases, a single time series is converted to a time series database through the use of a sliding window incrementally [9,10,11,12,17] or in discrete steps according to a known period [13,14]. Specifically in [17], the authors present a technique, called *SAX*, which addresses anomaly detection using time series discords, and applies it to real cases. We chose to compare our technique with this one because there is a graphic visualization tool and user interface, called *VizTree* [18], which implements the technique. Unlike methods that seek anomalies of a pre-specified length, the method presented in [15] looks for anomalies at varying levels of granularity (i.e., day, month, year), using a tree structure called *TSA-Tree* that contains pre-computed trend and anomaly information in each node. The InfoMiner technique [13, 14] detects "surprising" patterns on periodic event sequence data. Thus, the data is already discretized, and the known period allows the authors to treat a single continuous time series as a set of smaller one period time series. The work presented in [1] introduces a technique to identify patterns in time series data using an algorithm called by them as *PCAD* – Periodic Curve Anomaly Detection, which is a clustering-based algorithm built above the basis of the k-means algorithm, that outputs a ranked list of both global and local anomalies. The technique developed in [16] proposes an approach that employs a kernel matrix alignment method to capture the dependence relationships among variables in the time series in order to detect anomalies.

Some of these works have been extracted from [19], which brings a literature survey about clustering time series data stream that we recommend to be read as a supplementary reference about the related work. We chose some of the most recent researches regarding anomaly detection in time series data. In the next section we present our proposal of an algorithm based in clustering techniques with some enhancements built to let it recognize anomalies in a single time series data. The results of its application will be further evaluated in this paper.

2 The Proposed Algorithm

Our algorithm, presented in Box 1, is a dynamic clustering procedure that, given (i) a time series $T = \{t_1, t_2, \ldots, t_{|T|}\}$ of real-valued, time-indexed variables sampled at a certain frequency and ordered by time, (ii) the initial clusters' size τ, and (iii) the clustering factor φ, computes a set of anomalous patterns in T, $\mathbf{P} = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_{|\mathbf{P}|}\}$, where $\boldsymbol{P}_j = \{C_1, C_2, \ldots, C_{|\boldsymbol{P}_j|}\}$ is an anomalous pattern of T, which is composed of a set of disjoint clusters $C_k = \{t_a, t_{a+1}, \ldots, t_b\}$, $1 \le a \le b \le |T|$, that are ordered subsets of T.

C-AMDATS (T, τ, φ)

```
1.  C ← ComputeInitialClusters(T, τ);
2.  while changes in C happen do
3.    C' ← C
4.    for i ← 1 to |T| do
5.      Move t_i from its cluster in C to the
6.      nearest cluster in C according to
       f(C')
7.    endfor
8.  endwhile
9.  repeat
10.     Add C_1 to P;
11.     Remove C_1 from C;
12.     k ← 1;
13.     while k <= |C| do
14.         for j ← 1 to |P| do
15.             if P_j is similar to C_k then
16.                 Add C_k to P_i;
17.                 Remove C_k from C;
18.             else
19.                 k ← k + 1;
20.             endif
21.         endfor
22.     endwhile
23. until |C|=0
24. for i ← 1 to |P| do
25.     Compute the Anomaly Rank r(P_i)
26. endfor
27. SortByAnomalyRank(P);
28. return P;
```

Box 1 – C-AMDATS Pseudo-Algorithm

The algorithm starts in line 1 by computing the set of equal sized initial clusters, $\boldsymbol{C}$. In this step, the set T is split into a set of sets, $\boldsymbol{C}$, where each subset $C_k = \{t_a, t_{a+1}, \ldots, t_b\}$ has size $|C_k| = \tau$, i.e., $b - a = \tau$ (except the last cluster, in cases where $|T|$ is not divisible by τ). After that, in lines 2-8, the algorithm rebuilds $\boldsymbol{C}$ iteratively using its copy, $\boldsymbol{C'}$, computed before the iteration (line 3). For that, in lines 5-6, the algorithm uses $f(\boldsymbol{C'})$ to determine which cluster in $\boldsymbol{C}$ is the nearest to t_i. In the algorithm, $f(\boldsymbol{C'})$ computes the average of the time series segments within each cluster C'_k, or the centroid of each C'_k, $m_k = (t_a + t_{a+1} + \ldots + t_b) / (b - a)$ (see **Fig. 1**), and the distance, $d(t_i, m_k)$, from t_i to each m_k, for $1 \le k \le |\boldsymbol{C'}|$. Using these distances, in lines 5-6 the algorithm moves t_i from its current cluster in $\boldsymbol{C}$ to the cluster C_k, where k is the index of the cluster C'_k whose centroid m_k is the nearest to t_i according to $d(.,.)$. Our choice of distance function $d(.,.)$ is the Mahalanobis distance [20]. We explain this choice below but, in our experiments, we also use the Euclidian distance for comparison.

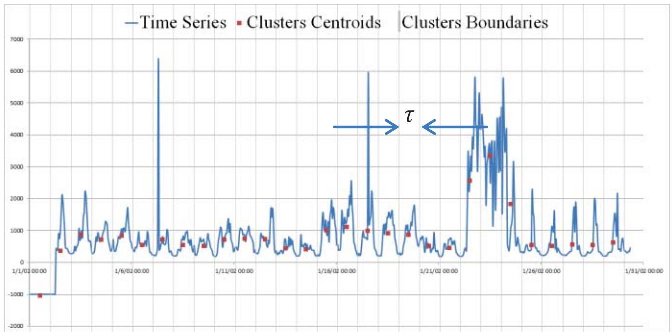


Fig. 1. A time series T split into equal-sized clusters C'_k, each of which of size t. The red dots are the centroids, m_k, of each cluster C'_k.

The loop of lines 2-8 terminates when no sample t_i is moved in lines 4-7. After this loop, we have the set of sets, $\boldsymbol{C}$ (see **Fig. 1**), composed of clusters of samples, C_k, that better group the samples according to their sample values distributed over time (please compare **Fig. 1** with **Fig. 3**; note that the size of each cluster C_k in **Fig. 3** is not the same). This happens because of our choice of distance function. In clustering problems, it is common to use the Euclidean distance function. Its use leads to clusters with circular format, due to the fact it does not take into account the variance of each dimension of the data set. However, it is possible that this circular shape may not be suitable to represent the cluster's shape. To solve this problem, another distance function should be used to build clusters that take in consideration variances in the x and y axes. The Mahalanobis distance differs from the Euclidean distance in that it takes into account the variances of each dimension (see **Fig. 2**). The **Equation (1)** presents the formulation of the Mahalanobis distance [20]:

$$d_m(x,\mu) = \sqrt{(x-\mu)^T S^{-1}(x-\mu)} \tag{1}$$

In **Equation (1)**, $x = (x_1, x_2, \ldots, x_n)^T$ is a specific variable in the data set, where n is the number of dimensions of the variables, $\mu = (\mu_1, \mu_2, \ldots, \mu_n)^T$ is a certain cluster centroid and S is the covariance matrix relative to that cluster.

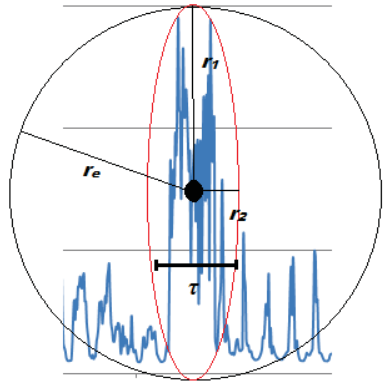


Fig. 2. A sample of a time series illustrating the differences between the application of the Euclidean (forming the circle) and Mahalanobis (forming the ellipse) distance functions.

In **Fig. 2**, τ is the initial clusters' size, r_e is the radius of the circle that fits the cluster, and r_1 and r_2 correspond to the radii of the ellipse that fits the same cluster as well. The r_e value is the Euclidean distance of farthest point in the cluster to its centroid, being big enough to embrace all the points in the cluster, while the r_1 and r_2 values are obtained by the application of the Mahalanobis distance. As we can note, the shape that best fits the cluster is the ellipse, while the circle is grouping regions that do not fall into the cluster. It is due to the fact that the Mahalanobis distance function takes into account both dimensions simultaneously, not separately. In order to show the real impact of using this distance function rather than the Euclidean distance, we will present the results of applying both to a real case in the Section 3.

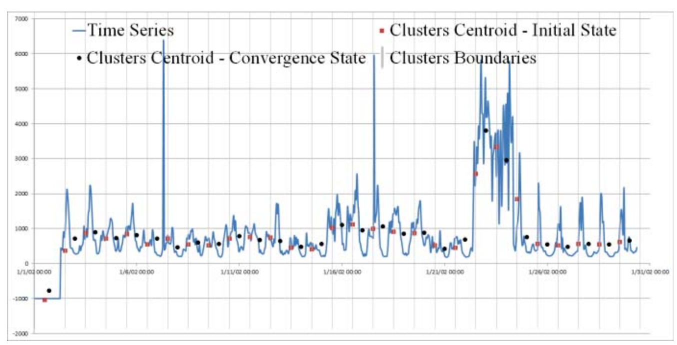


Fig. 3. A time series T split into clusters C'_k, each of which of variable size. The red dots are the centroids at the initial state of the algorithm, and the black dots are the centroids after the iterative process at lines 2-8

The following step (lines 9-23) performs the task of finding the final patterns **P** in the time series T. After all clusters have been found, the algorithm verifies which clusters are *similar*. This similarity is calculated using the standard deviation σ_y of the real-values of the variables in *T*, the y-coordinate of each cluster and the *clustering factor* φ. If the modulus of the difference between the y-coordinate of the centroids of two clusters is less than or equal to σ_y times φ, then these clusters can be *merged*, meaning that they will represent the same pattern ***P***. This task is performed till all the clusters have been analyzed.

In the last step (lines 24-27), the algorithm performs the detection of the anomalies. An anomaly is a pattern that does not conform to an expected behavior in *T*, i.e. an anomalous pattern. This detection is done by computing the anomaly score *r* for each pattern ***P*** found in the previous step, which is calculated as the ratio of the size of the entire time series by the summation of the sizes of the clusters present in ***P***. The anomaly score (or rank) *r* is a measure of how much ***P*** is interesting in terms of being an anomaly. Following, the entire set **P** is ordered by *r* in descending order, and the anomalous patterns will be those with the highest anomaly score values. The higher the anomaly score value for a pattern ***P***, the greater is its chance to be an anomaly in *T*. In **Fig. 4** we present the final state of the algorithm: all similar clusters have been merged into a pattern, as stated by the criteria described above. Three patterns have been found, and according to their anomaly score, the most anomalous are those highlighted in red and green color, while the blue pattern is the least.

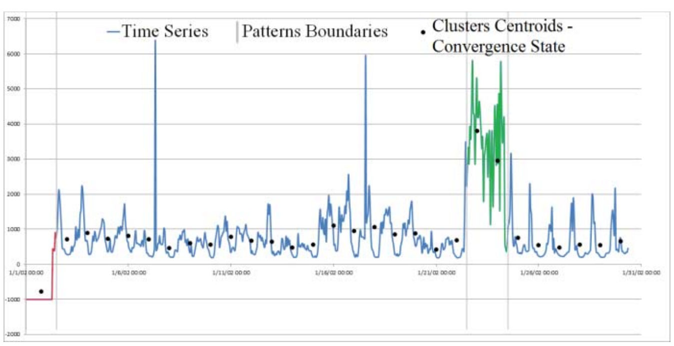


Fig. 4. A time series *T* divided into three patterns, at the final of the execution of the algorithm. The green and red are the most anomalous.

The complexity of the *C-AMDATS* is *O(nkz)*, where *n* is the number of variables, *k* is the number of initial clusters, and *z* is the number of iterations till the convergence state. Since it is a derivation of the k-means algorithm, it can also be classified as a NP-Hard problem [21], meaning that the algorithm will stop at the *z* iteration due to its stop criterion, but there is no guarantee that the absolute minimum of the objective function can be reached.

In Section 1.1 we presented a review of the related work. To the best of our knowledge, it was not possible to find any other clustering algorithm for anomaly detection in time series data that could be even similar to this technique here presented.

3 C-AMDATS – Applications and Results

To verify its ability to analyze real time series data, this technique was applied to real cases. Hence, a real case episode was selected. It will be further presented and discussed, as well as the results of the application of the C-AMDATS algorithm. During the tests, two versions of the algorithm were developed: one using the Euclidean distance ($C\text{-}AMDATS_E$) and the other using the Mahalanobis distance ($C\text{-}AMDATS_M$). The experiments showed that the application of the Mahalanobis distance led to better results, but it took more CPU time than the application of the Euclidean distance function due to the need to compute the inverse of the covariance matrix for each cluster, at each iteration step. We will present results using both distance functions.

To assess the algorithm's performance with respect to its ability to identify the same anomalous patterns identified by the human specialists, its results were compared to those patterns using the *precision*, *recall* and *accuracy* methodologies [22]. Also, a *confusion matrix* was built to show the differences between each approach.

3.1 Real Case – Carbon Monoxide

This case refers to the measurement of carbon monoxide during two months in the year of 2002. The data is hourly sampled, and was collected in a metropolitan area, by an automated monitoring system maintained. For reference, we will use its chemical representation, *CO*. Its cycle's length is of 24 hours.

The **Fig. 5** shows the result of the $C\text{-}AMDATS_M$ approach for this case. Three major patterns are highlighted: the red, green and blue, in order of anomaly score, which led us to select the patterns highlighted with red and green color as the anomalous patterns. In **Table 1** we present the confusion matrix for this case. The patterns are also delimited by vertical lines.

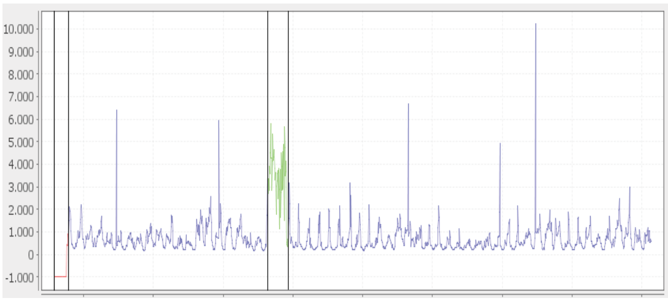


Fig. 5. Time plot for CO highlighting the anomalous patterns found by *C-AMDATS*$_M$

Table 1. Confusion matrix for this case

	Anomalous Pattern		*Precision*	*Recall*	*Accuracy*
***C-AMDATS*$_M$**	82	2	0.9762	0.9111	0.9931
	1 348	8			
***C-AMDATS*$_E$**	66	7	0.9041	0.7333	0.9785
	1 343	24			
SAX	40	8	0.8333	0.4444	0.9597
	1 342	50			

The values of the parameters for *C-AMDATS* were: *initial* clusters' *size* of 24 hours and *clustering factor* of 1.2. For SAX, we spent about 2 hours looking for a best combination of its parameters, and we found that a *window length* of 24h, *number of symbols per window* of 3 and *alphabet size* of 4 performed the best. Moreover, we also had to set one advanced option in the VizTree tool, called "*No Overlapping Windows*", which led to the best results we could experiment.

Similarly, both approaches were able to find, some partially, the region of the anomaly subsequence. However, the SAX approach was just able to give a clue about the second anomaly, as we can see in **Fig. 6**, and the *C-AMDATS*$_M$ could give a good result in comparison with the others.

For SAX, we extracted the most meaningful branches regarding these anomalous patterns, which corresponded to "ccc" and "acc".

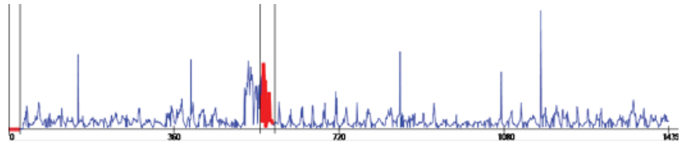


Fig. 6. Time plot for CO highlighting, in red, the anomalous region found by the SAX technique

4 Conclusions and Future Work Recommendations

In this work we presented a proposal of an algorithm for anomaly detection in time series. We showed that there is a plenty of applications, and also many contributions made to this research field. We identified the main contributions presented recently and analysed them to identify whether the algorithm proposed by these authors is indeed a new contribution to the community. We verified that there is no similar technique. We presented the concepts behind this work, and then we described our proposal. Finally, we applied the algorithm to a real data case, and have identified that it performed good results in comparison with the other approach, which shows that it can be applied as a tool to leverage the specialists' job in analyzing and identifying anomalies in time series data.

There are several future works to be developed in the following:

a) to write a full paper of this work, describing in more details some issues regarding the review of the related work, the description of the algorithm and the methods for assessing its performance, applying the algorithm to other real cases and compare with the same technique (SAX);

b) to use *C-AMDATS* in an operational environment, where the algorithm would be set up to work continuously, with the jobs of analysing time series data, finding patterns, and sending status report to specialists. Then, these specialists would be able to verify the results of the algorithm at real time;

c) to implement a function to analyze correlated parameters at once to find anomalies between them, e.g. ozone and solar radiation, which are different parameters but have an intrinsic correlation. This recommendation would demand creating a derivation of the *C-AMDATS* algorithm to be applied to analyze various time series data at the same execution;

d) to design a learning module to learn from the user what are the best values computed at a certain moment, based upon past applications of the algorithm that have been validated by the user.

Based on the results here presented, we think this work could be successfully applied in several areas of this research field to improve the way time series data are analyzed in order to detect anomalies.

5 Acknowledgements

We acknowledge the Espirito Santo Research Support Foundation (*FAPES*), which partially funded this work. We also acknowledge our contributors, who gave us access to real data and let us publish the results. Namely, we thanks the Environmental Institute of the state of Espirito Santo, Brazil – *IEMA* for the data that is related to the real case.

6 References

[1] Umaa Rebbapragada, Pavlos Protopapas, Carla Brodley, Charles Alcock. “Finding Anomalous Periodic Time Series: An Application to Catalogs of Periodic Variable Stars”, Spring Machine Learning Journal (Springer), Vol. 74, Issue 3, 281-313, Mar 2009. DOI: 10.1007/s10994-008-5093-3;

[2] Edwin Knorr, Raymond Ng. “Algorithms for Mining Distance-Based Outliers in Large Datasets”, In: Proceedings of the 24th International Conference on Very Large Data Bases – VLDB, VLDB International Conference, pp. 392–403, 1998;

[3] Sridhar Ramaswamy, Rajeev Rastogi, Kyuseok Shim. “Efficient Algorithms for Mining Outliers from Large Datasets”, In: SIGMOD ’00: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, SIGMOD, pp. 427–438, 2000;

[4] Fabricio Angiulli, Carla Pizzuti. “Fast Outlier Detection in High Dimensional Spaces”, In: Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery, pp. 15–26, 2002;

[5] Mingxi Wu, Christopher Jermaine. “Outlier Detection by Sampling with Accuracy Guarantees”, In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 767–772, 2006;

[6] Markus Breunig, Hans-Peter Kriegel, Raymond Ng, Jörg Sander. “LOF: Identifying Density-Based Local Outliers”, In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 93–104, 2000;

[7] Wen Jin, Anthony Tung, Jiawei Han. “Mining Top-N Local Outliers In Large Databases”, In: Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 293–298, 2001;

[8] Dongmei Ren, Baoying Wang, W. Perrizo. “RDF: A Density-based Outlier Detection Method Using Vertical Data Representation”, In: Proceedings of the 4th IEEE International Conference on Data Mining, pp. 503–506, 2004;

[9] Dipankar Dasgupta, Stephanie Forrest. “Novelty Detection in Time Series Data Using Ideas from Immunology”, In: Proceedings of the International Conference on Intelligent Systems, pp. 82–87, 1996. DOI: 10.1.1.57.3894;

[10] Eamonn Keogh, Stefano Lonardi, Bill Yuan-chi Chiu. “Finding Surprising Patterns in a Time Series Database in Linear Time and Space”, In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 550–556, 2002;

[11] Junshui Ma, Simon Perkins. “Online Novelty Detection on Temporal Sequences”, In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 613–618, 2003;

[12] Li Wei, Nitin Kumar, Venkata Lolla, Eamonn Keogh, Stefano Lonardi, Chotirat Ratanamahatana. “Assumption-Free Anomaly Detection in Time Series”, In: SSDBM’2005: Proceedings of the 17th International Conference on Scientific and Statistical Database Management, pp. 237–240, 2005;

[13] Jiong Yang, Wei Wang, Philip Yu. “Infominer: Mining Surprising Periodic Patterns”, In: Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 395–400, 2001;

[14] J. Yang, W. Wang, P. S. Yu. “Mining Surprising Periodic Patterns, Data Mining and Knowledge Discovery”, Data Mining and Knowledge Discovery (ACM), Vol. 9, Issue 2, 189–216, Sep 2004;

[15] Cyrus Shahabi, Xiaoming Tian, Wugang Zhao. “TSA-Tree: A Wavelet-Based Approach to Improve the Efficiency of Multilevel Surprise and Trend Queries on Time-Series Data”, In: Proceedings of the 12th

International Conference on Statistical and Scientific Database Management, pp. 55–68, 2000;

[16] Haibin Cheng, Pang-Ning Tan, Christopher Potter, Steven Klooster. “Detection and Characterization of Anomalies in Multivariate Time Series”, In: Proceedings of the 9th SIAM International Conference on Data Mining, pp. 413–424, 2009;

[17] Eamonn Keogh, Jessica Lin, Ada Fu. “HOT SAX: Efficiently Finding the Most Unusual Time Series Subsequence”, In: Proceedings of the 5th IEEE International Conference on Data Mining (ICDM), pp. 226–233, 2005;

[18] Jessica Lin, Eamonn Keogh, Stefano Lonardi, Jeffrey Lankford, Daonna Nystrom. “VizTree: a Tool for Visually Mining and Monitoring Massive Time Series Databases”, In: Proceedings of the 30th International Conference in Very Large Data Bases, pp. 1269-1272, 2004;

[19] V. Kavitha, M. Punithavalli. “Clustering Time Series Data Stream – A Literature Survey”, International Journal of Computer Science and Information Security (IJCSIS), Vol. 8, Issue 1, pp. 289-294, Apr 2010;

[20] Julius T. Tou, Rafael C. Gonzalez. “Pattern Recognition Principles”, Addison-Wesley, 1974;

[21] Meena Mahajan, Prajakta Nimbhorkar, Kasturi Varadarajan. “The Planar k-Means Problem is NP-Hard”, In: Proceedings of the 3rd International Workshop on Algorithms and Computation, Springer-Verlag, pp. 274–285, 2009. DOI: 10.1007/978-3-642-00202-1_24;

[22] David L. Olson, Dursun Delen. “Advanced Data Mining Techniques”, Springer, 2008.

Determining Signal Source Integrity Using a Semi-supervised Pattern Classification System

W. Satney[1], A. Carrington[2], T. Scantlebury-Manning[2] and A. Als[1]
[1]Computer Science, Mathematics and Physics, University of the West Indies (Cave Hill Campus), BB1100,
[2]Biological & Chemical Sciences, University of the West Indies (Cave Hill Campus), BB1100

Abstract - *This paper focuses on a unique signal classification problem that employs digital signal processing techniques to first, separate 57 audio signals into two signal-type categories (A and B) and second, to further classify the integrity of the category A signal sources. Short-Time Fourier Transform and central tendency analyses are employed to distinguish between the signals within the categories. FFT Welch Method and Kohonen's Self-Organizing Maps are then employed to determine the integrity level of the sources associated with the signals in category A. The overall results show 91.2% accuracy in classifying the signals into categories A and B. Additionally, the system was able to achieve 100% classification when distinguishing between the poor and (somewhat) good signal source integrity. The hybrid classification system proposed in this paper has direct application to real world problems where both signal isolation and the associated integrity of the signal source need to be determined.*

Keywords: Signal classification, Semi-supervised learning system, FFT Welch Method, STFT, SOM.

1 Introduction

The principle concern of computer scientists and engineers when developing expert systems is the ability for these systems to apply effective pattern recognition techniques to solve classification problems. Machine learning algorithms have revolutionized how computers classify related data to decipher patterns in the data presented. Indeed, the ultimate goal of these algorithms is to mimic the learning ability of humans.

Machine learning tasks are generally broken into two main categories: supervised learning and unsupervised learning. Supervised learning algorithms are used when training data is provided as part of the problem. In this case the training data is composed of set of training targets. The algorithm analyses the training data and deduces a classifier function or a regression function. The deduced function is then able to predict the correct output classification for valid input data. Conversely, unsupervised learning algorithms are used when there is no output target(s) associated with each input data sample. Instead, unsupervised learning algorithms learn how to delineate specific input patterns based on the statistical structure of the entire set of input patterns. Another machine learning mode that may be applied to pattern recognition problems is semi-supervised learning. Semi-supervised learning techniques basically encapsulate a supervised learning algorithm and an unsupervised learning algorithm.

This research paper proposes a hybrid classification system that utilizes information made available by Short-Time Fourier Transform (STFT) analysis to distinguish between two signal categories. Feature extraction is then performed thereby simplifying the problem domain by further reducing the amount of data passed on to the pattern recognition stage. Specifically, the simplified problem is applied to the Artificial Neural Network (ANN) system for further classification. These techniques will be highlighted in the next section.

2 Background

A complete pattern recognition system typically consists of a signal gathering mechanism to record the source signals; a feature extraction mechanism to reduce dimensionality through the creation of a feature vector; and a pattern classification mechanism to distinguish between the patterns present within the source signals. Designing an effective signal classification system to solve pattern recognition problems is challenging. In particular, feature vector creation and the design of the pattern classifier prove to be the most challenging aspects. As the extraction vector is often based on spectral or entropy analyses, a brief background on some of these spectral techniques and the Self-Organized Maps pattern classification algorithm will be presented in this section.

2.1 Discrete Fourier transform (DFT)

The DFT technique is able to decompose any perpetual periodic signal into sinusoidal waves [1]. The frequency and the associated amplitude information are the key aspects of DFT required for signal analysis. The equation below represents the DFT:

$$X[k] = \frac{1}{N}\sum_{n=1}^{N-1} x[n]e^{-j2\pi kn/N} \qquad (1)$$

Both the frequency domain, $X[k]$, and the time domain, $x[n]$, are arrays of complex numbers, with k and n running from 0 to N-1. Note that k represents the k^{th} frequency component, n represents the n^{th} sample and j represents the imaginary unit. The Fast Fourier Transform (FFT) is an improved method for calculating the DFT. Although both FFT and DFT produce the same results, the FFT is exceptionally efficient at significantly reducing the computation time required [1]. Unfortunately, using Fast Fourier Transform (FFT) based methods alone have inherent problems which can greatly affect the accuracy of the analysis. This is because FFT-based methods contain a finite record of data that relate to the frequency spectrum of the signal. Consequently, these methods are susceptible to spectral leakage effects due to the windowing that is ingrained in finite-length data records. Furthermore, smearing or smoothing the estimated spectrum is the predominate effect of windowing in FFT based methods [2, 3]. Finally, FFT methods are not suitable for real world signals in which the frequency content changes often over time. Thus, periodic waves with constant frequencies are best analysed using FFT methods [4]. As pattern recognition problems usually have some real world application in which non-stationary signals are investigated [5-9], Short-Time Fourier transform (STFT) and FFT Welch Method are techniques used to overcome the challenges associated with the FFT.

2.2 Short-Time Fourier transform (STFT)

The STFT technique is able to analyse non-stationary signals in the time domain through an algorithm that is applied to various sections of the signal. The signal is processed through a moving window which breaks down the signal into a set of overlapping or non-overlapping segments in which FFT is applied on each segment. Tokmakç et. al use STFT in the classification system for stenosis from mitral valve Doppler signals yielding excellent results [10].

2.3 FFT Welch method

The application of the Welch method on the FFT algorithm is used to estimate the power spectra. This technique involves sectioning signals into overlapping segments with each data segment termed a window. The principal advantage of using this technique is that it involves less computations that other methods and it is suitable for performing analysis on non-stationary signals [5, 11]. In this study, each signal is divided into windowed sections using the Hanning window. The modified periodograms are then calculated and averaged.

2.4 Self-organizing maps (SOMs)

SOMs 'learn' to detect the inconsistencies and correlations in the input vectors to classify these vectors into a perspective grouping on a two-dimensional grid [12]. The Neurons in the self-organizing map 'learn' to recognize adjacent clusters on the grid. Any four of the distance functions (i.e. dist, boxdist, linkdist and mandist) may be used to calculate the distance between the neurons from their position [12]. For instance, the dist function calculates the Euclidean distance from a home neuron to another. The neuron with the smallest Euclidean distance is selected as 'winner' and is moved towards the presented input data [13].

2.5 The Study

An assortment of fifty-seven (57) audio signals produced from two categories of sources were obtained from two independent parties[1] and stored in a signal-bank library. Each signal was labelled, A or B, based on its source category. A technical expert was instrumental in confirming the correct labelling (A or B) of the fifty-one signals supplied by the UWI. The signals in the standardized set were already pre-labelled, thus no confirmation was required.

The signals in category A represent the desired target classification for this study. Conversely, those in category B are to be discarded. Each signal in Category A contains data that may infer on the integrity of the signal's source along a continuum.

The primary objectives of this study are to employ digital signal processing (DSP) techniques to 1) separate the signals into the two categories and 2) inform on the integrity status of the sources in the desired target category. The methodology employed to accomplish these two objectives are described in the next section.

3 Methodology

3.1 Objective #1 – The signal separation process

Differentiation between the signal sources is accomplished by a two-stage filtering process as depicted in Figure 1. In stage one, a STFT analysis was employed to determine the mean (μ) spectral intensity at a predetermined reference frequency. The mean was then compared to a classification threshold value to determine the signal category. At the end of this stage the signals in category A begin to get filtered out from the entire pool of signals. The batch of signals discarded from the first stage invariably contains signals from both categories. Thus the objective of stage two is to attempt to completely separate the remaining category A signals from those in category B.

[1] Six (6) signals were obtained from a standardized data set and the remaining fifty-one (51) from the University of the West Indies (UWI) (Cave Hill Campus).

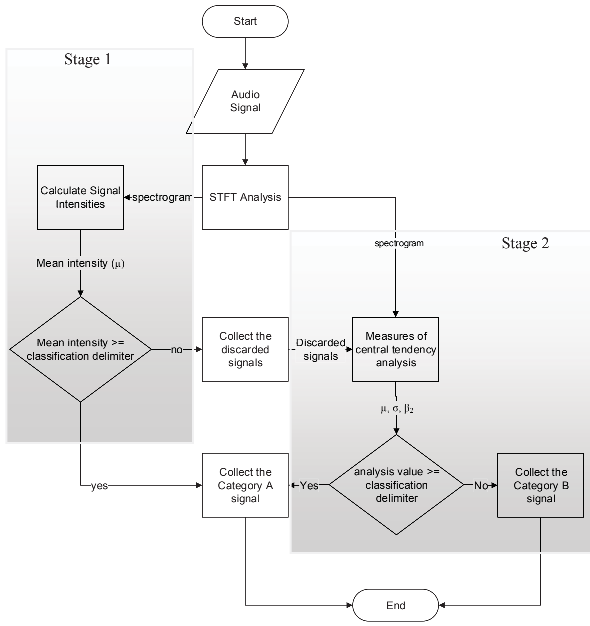


Figure 1 - Signal Separating Process

In this stage, further separation is accomplished via a central tendency analysis of the mean (μ), standard deviation (σ) and kurtosis (β_2) of the signal intensities associated with the spectrograms. More specifically, these values are used as input to a simple algorithm which compares them to predetermined thresholds in order to separate out the remaining category B signals. The pool of signals that remain after this separation represent category A signals.

3.2 Objective #2 – The determining integrity status

The use of statistical analysis (mean, standard deviation, min, max, kurtosis and skewness) derived from the FFT Welch method is input into the SOM network to classify each of the Category A signals obtained from stage one of the signal separation process. This classification may then be used to infer certain qualities about the signals' source. The description of the six qualities that can be determined based on the SOM classifications are provided in Table 1.

Table 1 – Signal Classification Listing

Classification Category	Description
1	Noisy Signal
2	Below average
3	Somewhat average
4	Average
5	Good
6	Very good

SOM categories two (2) through six (6) infer on the relative integrity of the signal source along a continuum. Those signals classified in category 1 are deemed to be too noisy and should be discarded.

It should be noted that all work was undertaken using MATLAB and the key parameters employed are provided in Table 2.

Table 2 - Key Experiment Parameters

Signal Characteristics	
Recording sample frequency	44100 Hz
STFT Analysis	
Hamming window size	2^{10} data points
Overlap	50%
FFT Welch Method	
Number of samples	ns = max(size(sig))
Number of windows	nw = 16
Hanning windows size	hanning(floor(ns/nw))
SOM Analysis	
Num. of Neurons	6 (2 x3 hexagonal network)
Num. of epochs	7000

4 Results

Figures 2 and 3 show two examples of spectrograms associated with signals from Category A and B, respectively. The proposed two-stage separation method proved to be 91.2% accurate in correctly distinguishing between the signals in these categories. Only three (3) signals from category A and two (2) signals from category B were miscategorised.

The technical experts are acute when distinguishing between sources that are either (very) good or poor. Example spectrograms associated with these signals classifications are provided in Figures 2 and 4, respectively. Indeed, even to the untrained ear, the audio signals associated with these two sounds are distinct. However, classification categories 2, 3 and 4 as described in Table 1 often prove difficult to distinguish even to trained experts. Therefore the value of the proposed system lies in its ability to determine the integrity of the signal's source along the entire continuum from 2 to 4.

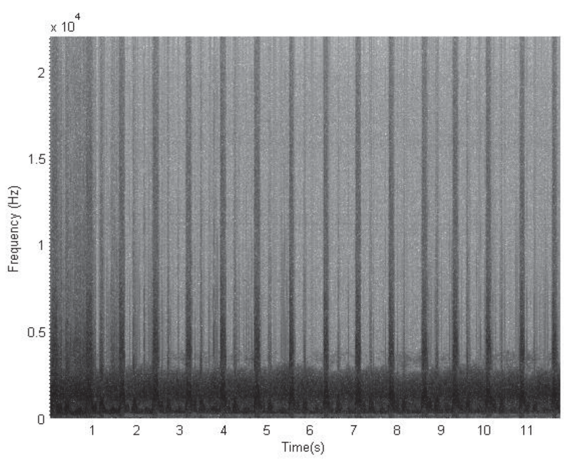


Figure 2 – Spectrogram of signal from Category A

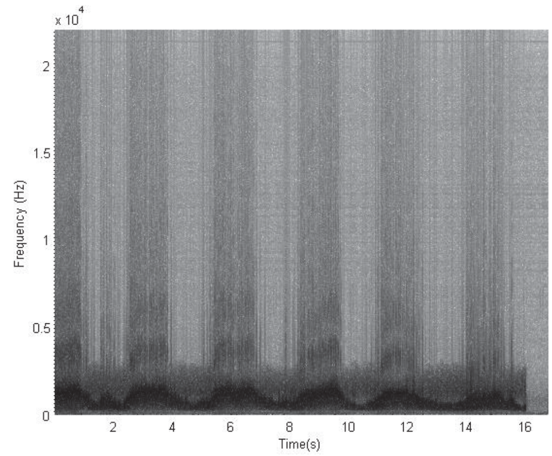


Figure 3 – Spectrogram of signal from Category B

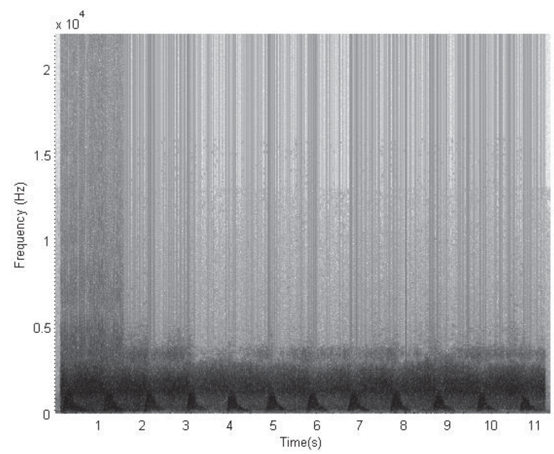


Figure 4 – Spectrogram of signal from Category A with poor structural integrity

A cluster analysis was done on all the signals obtained from stage 1 of the signal separation process using a fully connected self-organizing map. Figure 5 shows the location of the six (6) neurons in relation to the data and therefore, by extension, it also highlights the six associated clusters.

Cluster 6 represents the only signal whose source was considered to be very good. Cluster 5, which represents those sources considered to have good integrity, contained seven (7) signals. Cluster 2, 3 and 4 contained eighteen 18 signals in total and represent sources with integrity levels from average to below average along the continuum. Finally, noise dominated the signals in cluster 1. Thus this cluster represents the case where the signal samples that are unusable.

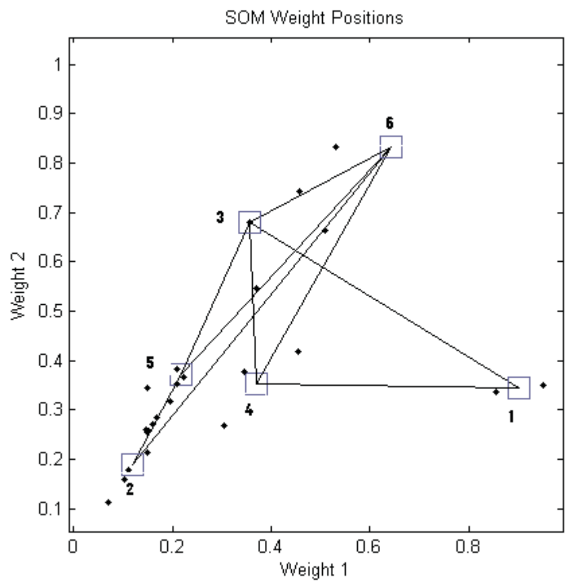


Figure 5 - SOM Weight Positions

Analysis of the pool of category A signals obtained from stage two of the separation process revealed that they were all from sources with poor integrity.

The two-stage separation process correctly identified all five (5) of the poor signal sources and the single good signal source within the standard data set. A UWI technical expert was used to classify the other fifty-one (51) signals which they supplied. This expert confirmed that the system correctly identified 18 poor cases.

5 Discussion

In this study, we investigated a total of 57 signals from two signal categories (A and B) obtained from two independent sources. The proposed STFT and FFT Welch method combined Kohonen's Self-Organizing Maps proved successful in both separating the signal categories and further classifying the integrity of each signal source in the target category. Specifically, the system is 91.2% accurate when distinguishing signal source categories. Furthermore, the system is 100% accurate in determining both the poor integrity category A source signals, the good integrity category A source signal and those recorded poor quality signals that are unusable due to background noise.

In the field technical experts have the ability to distinguish between signals within categories A and B. Moreover they are also capable of distinguishing between the good and the poor category A signal sources. The potential benefit of this work lies in the ability of the system to one step further and infer on the integrity of the signal's source along a continuum. This is seen in the 27 cases which proved difficult to place on the continuum using only the expert's auditory senses.

6 Conclusion

The proposed hybrid system was 91.2% successful in separating the two signal categories. Noise within the recorded signal is believed to be a major contributor to the categorisation errors. The system also shows great promise with its ability to determine the integrity of the signal's source along a continuum. Future work will examine DSP techniques that will serve to reduce the presence of this noise and raise the categorization success percentage and improve the signal source integrity analysis.

7 References

[1] S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. San Diego, California: California Technical Publishing, 1999.

[2] E. Ubeyli and I. Guler, "Spectral analysis of internal carotid arterial Doppler signals using FFT, AR, MA, and ARMA methods," *Computers in Biology and Medicine,* vol. 34, pp. 293-306, 2004.

[3] D. H. Evans, W. N. McDicken, R. Skidmore, and J. P. Woodcock, *Doppler Ultrasound: Physics, Instrumentation and Clinical Applications*. Chichester: John Wiley and Sons, 1989.

[4] D. L. Fugal, *Conceptual Wavelets in Digital Signal Processing*. San Diego, California: Space & Signals Technical Publishing, 2009.

[5] S. Kara and F. Dirgenali, "A system to diagnose atherosclerosis via wavelet transforms, principal component analysis and artificial neural networks," *Expert Systems with Applications,* vol. 32, pp. 632-640, 2007.

[6] F. Hardalac, H. Yildirim, and S. Serhatlioglu, "Determination of carotid disease with the application of STFT and CWT methods," *Comput Biol Med,* vol. 37, pp. 785-92, Jun 2007.

[7] E. D. Übeyli and İ. Güler, "Detecting variabilities of Doppler ultrasound signals by a modified mixture of experts with diverse features," *Digital Signal Processing,* vol. 18, pp. 267-279, 2008.

[8] J. H. Houtveen and P. C. M. Molenaa, "Comparison between the Fourier and Wavelet methods of spectral analysis applied to stationary and nonstationary heart period data," *Psychopysiology,* vol. 38, pp. 729-735, 2001.

[9] R. Rock, A. Als, P. Gibbs, and C. Hunte, "The 5th Umpire: Automating Cricket's Edge Detection System," *Systemics, Cybernetics and Informatics,* vol. 2, pp. 4-9, 2013.

[10] M. Tokmakçi, "A Classification System for Stenosis from Mitral Valve Doppler Signals Using Adaptive Network based Fuzzy Inference System," *Journal of Medical Systems,* vol. 31, pp. 329-336, 2007.

[11] P. D. Welch, "The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms," *Audio and Electroacoustics, IEEE Transactions on,* pp. 70-73, 1967.

[12] M. H. Beale, M. T. Hagan, and H. B. Demuth. (2014). *Neural Network Toolbox User's Guide (2014b ed.)*.

[13] Y. Liu, R. H. Weisberg, and C. N. K. Mooers, "Performance evaluation of the self-organizing map for feature extraction," *Journal of Geophysical Research,* vol. 111, 2006.

Adaptive WiFi Positioning System with Unsupervised Map Construction

Ka-Sing Lin[1], Albert Kai-Sun Wong[2], Tsz-Lung Wong[2] and Chin-Tau Lea[2]
[1]ECE Department, The Hong Kong University of Science and Technology, Hong Kong

Abstract— *A major problem in indoor localization based on WiFi RSS (Received Signal Strength) is the tedious offline surveying process for fingerprinting. To address this problem, we propose an unsupervised Simultaneous Localization and Mapping (SLAM) system for automatic floor map and radio map construction. All it takes to set up this system is for a surveyor to walk through the coverage area randomly several times to collect traces of WiFi RSS measurements. Based on similarity matching of these measurements, a floor map in form of a graph is automatically constructed, together with a radio map which associates individual WiFi RSS distributions with sample points along the graph. In the online stage, this system can be dynamically updated using crowdsourced RSS data. Algorithms are also discussed to increase the computational efficiency, as the size of the database can increase squarely with the scale of the test bed.*

Keywords: WiFi Localization, Unsupervised Data Mining, Pattern recognition, Crowdsourcing

1. Introduction

The capability to accurately localize people carrying mobile smart gadgets have become very important in many applications. The Global Positioning System (GPS), while ubiquitously available outdoor [2], [3], is generally not applicable in indoor. As IEEE 802.11 WiFi has become a standard feature of mobile gadgets and is widely deployed within buildings, much research effort has been directed on WiFi-based indoor localization. While some radio localization systems can be based on measurements such as time of arrival (TOA) or time difference of arrival (TDOA) of signals [4], [5], for WiFi, received signal strength (RSS) based localization, specifically the fingerprint approach for which knowledge of locations of APs is not required, is generally regarded as most suitable because of its low-cost and low-complexity. The fingerprint approach, however, requires a time-consuming surveying process to establish a radio map. Many research works [6], [7], [8], [11] have been done to increase the accuracy for location estimation but relatively few works have been done to address the surveying process. [9] reports that 18.3 hours were used to survey a 70m * 23m area. Further, WiFi Access Points (APs) may be turned on and off or changed in deployment over time, rendering the radio map obsolete.

1.1 Related works

Because of the survey-effort problem, the concept of Simultaneous Localization and Mapping (SLAM) [1], [10], used in robotic systems is generating interests in personal localization . In SLAM, sensors are used to track the agent and for a map to be built concurrently so that a tiresome surveying process to map the environment manually can be avoid. [1] applies the SLAM as well as the crowd-sourcing concept to WiFi localization so that a graphical floor map of a coverage area can be automatically constructed based on many randomly collected WiFi RSS traces. The graphical map represents a grid-like coverage area by vertices and edges. However, because of random variations in WiFi signals, superfluous vertices and edges are often produced in the graphical map. Further, to support localization, a radio map needs to be constructed on top of the floor map.

1.2 Main contributions

In this paper, the work in [1] is extended to complete a novel Adaptive WiFi Positioning System (AWPS). Assuming a grid-like indoor environment, AWPS first constructs a floor map of the coverage area in form of a graph by extracting similarity patterns from unsupervised RSS data. Then a radio map is constructed on top of the floor map so that AWPS is ready for online localization. Our contributions can be summarized as follow:

1. Enhanced floor map construction - To increase the accuracy and robustness of the floor map construction process, we supplement the completely crowd-sourced approach in [1] by bootstrapping the system with RSS traces collected through a casual surveyor walk-through. The floor map of the coverage area is discovered by observing the correlation pattern of the RSS measurements. We propose a new approach, which is the use of a median filter, to smooth RSS sequences and to impute missing values. We propose also a new Pattern Closing algorithm to smooth the correlation patterns so that our Adaptive Correlation Pattern Recognition (ACPR) module can more accurately and efficiently detect Break Points (BPs) and Atomic Location Segments (ALSs) than the system in [1]. With the enhanced BPs and ALSs Purification algorithm, our system can eliminate the superfluous BPs and ALSs problem.

2. Points of Interest (POIs) included in map - In this paper, POIs are assumed to be rooms alongside corridors. Marking

POIs on the map enriches the information content of the map and is useful for navigation applications.

3. Radio Map Construction - We completed the localization system in [1] by adding the algorithms to construct a radio map on top of the floor map so that online localization can be performed.

Finally, we also briefly show how the system can be adaptively updated by crowd-sourced data after it has been bootstrapped.

The paper is organized as follows: The system overview and the experimental setup are stated in Section 2. Section 3 demonstrates the detailed methodology of the AWPS. The results are illustrated in Section 4. Finally, the conclusion is presented in Section 5.

2. System Design

This section describes the system flow of the AWPS and the experimental setup.

2.1 System Overview

The proposed AWPS consists of an offline and an online stage. The offline stage is made up of five processing steps as shown in Fig.1.

In [1], a large number of crowd-sourced RSS traces are collected during the offline stage. This system, on the other hand, begins by collecting only a couple of reference traces by a surveyor casually walking through the coverage area. After data preprocessing and ALSs and BPs detection, a graphical map and radio map can be constructed to support localization in the online stage. Traces collected during the online stage can be used as crowd-sourced data to complete any missing part of the floor map and to update the radio map. Thus the system is adaptive to changes in the environment; e.g. changes in WiFi deployment. The overall system flow is illustrated in Fig.1.

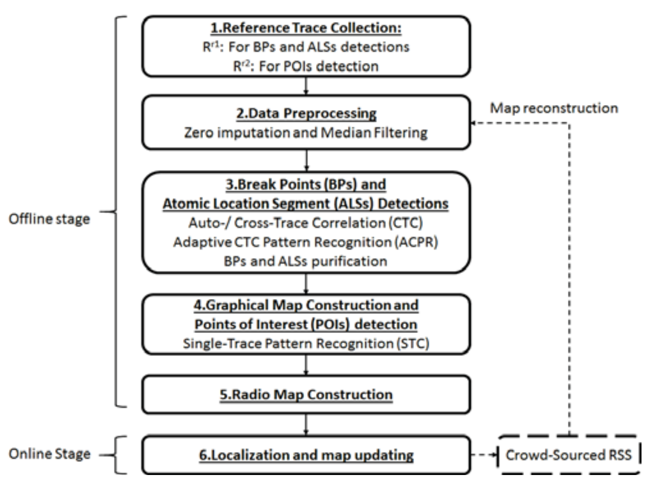


Fig.1 System flow of AWPS

The first step in the offline stage is reference trace collection. Two reference traces are collected by a surveyor to bootstrap the map construction. Each trace is a series of WiFi RSS measurements, each measurement being a list of RSS values from hearable WiFi APs and their corresponding MAC addresses. RSS values from hearable APs typically fall between -60 to -90 dBm. For convenience, RSS values are recorded as the measured dBm plus 100 so that RSS values for hearable APs are typically in the range of 10 to 40 and RSS values for non-hearable APs are recorded as 0. The first reference trace, $\mathbf{R^{r1}}$ is recorded by the surveyor walking randomly through the coverage area. It is used to detect Break Points (BPs) and Atomic Location Segments (ALSs). ALSs refer to corridor segments which users travel along. BPs indicate the T-junctions or crossroad intersections where corridor segments meet and where users may turn into different directions. Samples of ALSs and BPs are illustrated in Fig.2. To incorporate points of interest (POIs) into the map, the surveyor will collect a second reference trace $\mathbf{R^{r2}}$. While collecting $\mathbf{R^{r2}}$, the surveyor will stop at each POI for a few second before continuing the causal walk.

Next, the reference traces are preprocessed for missing value imputation and noise removal using median filtering. Then, an autocorrelation matrix for trace $\mathbf{R^{r1}}$ is computed, followed by the application of Adaptive Correlation Pattern Recognition (ACPR) to detect the BPs and ALSs in the trace. Purification algorithms are used to remove superfluous BPs and ALSs which may appear because of noises in the data. Then, a graphical map is constructed based on the set of unique ALSs and BPs identified. POIs are also included into the graphical floor map by applying a Single-Trace Pattern Recognition (STPR) algorithm on $\mathbf{R^{r2}}$. Finally, a radio map is generated and a floor map is constructed by integrating the radio map and graphical map. Further details are described in section 3.

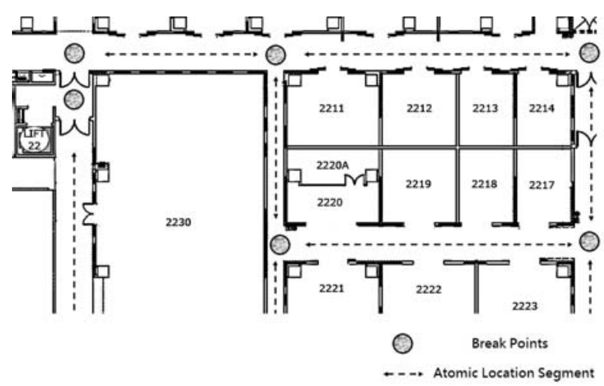


Fig.2 Example of ALSs and BPs

AWPS is ready for the online stage after the floor map and radio map are constructed. Users are localized by using a Maximum a posteriori (MAP) estimator. In point-by-point localization, AWPS matches the real-time RSS observed by the user with the database using likelihood function.

3. Methodology

The proposed AWPS applied multiple algorithms to construct the physical map. Median filtering are used to preprocess the traces. Auto- and cross-correlation computation (CTC), Adaptive Correlation Pattern Recognition (ACPR) and Single-Trace Pattern Recognition (STPR) are applied to identify BPs, ALSs and POIs. Furthermore, BP and ALS Purification ensure uniqueness of BPs and ALSs so that a graphical floor map accurately reflecting the physical environment can be constructed, using a graph algorithm described in [1]. The radio map is constructed and the user is localized by a MAP estimator. Details of all algorithms involved are described in this section.

3.1 Important Notation

In this subsection, we summarize the important variables used in this paper. Details descriptions of these variables are introduced in the corresponding subsections to follow.

Symbol	Meaning
$\mathbf{R^{r_1}}$	The Reference Trace by random walking
$\mathbf{R^{r_2}}$	The Reference Trace with stopping at POI
$\mathbf{R^X}$	A reference trace or cloudsourced trace
$\hat{\mathbf{R}}$	Concatenation of all traces, $\hat{\mathbf{R}} = [\mathbf{R^X}$ ...]
$\nu_{\mathbf{n}}^{\mathbf{X}}$	The n^{th} measurement in the X^{th} trace
$\mathbf{Q}$	Quantized matrix of $\mathbf{C}$
$\mathbf{C}$	The Cross-Trace (or Auto) Correlation (CTC) matrix
$\mathbf{C}^{\{\mathbf{X},\mathbf{Y}\}}$	Submatrix of CTC matrix corresponding to $\mathbf{R^X}$ and $\mathbf{R^Y}$
$C_{r,s}^{\{X,Y\}}$	CTC value of $\nu_{\mathbf{r}}^{\mathbf{X}}$ and $\nu_{\mathbf{s}}^{\mathbf{Y}}$
m^X	Number of hearable APs in $\mathbf{R^X}$
n^X	Number of measurements in $\mathbf{R^X}$
$\mathcal{M}$	Total number of hearable APs in the target area
$\mathcal{N}$	Total number of measurements in $\mathbf{R}'$

Table 1: Important Notations

3.2 Data Preprocessing

In the online and updating stage, the raw data available to the system is a super-trace that is the concatenation of WiFi measurements from many traces collected by many users $\hat{\mathbf{R}}$ = [$\mathbf{R^{r_1}}$ $\mathbf{R^{r_2}}$ $\mathbf{R^1}$ $\mathbf{R^2}$ ...]. While there is only one reference trace $\mathbf{R^{r1}}$ during the bootstrapping of the system, for convenience we will continue our discussion for the general case when there can be many different traces. Each trace is a series of WiFi RSS measurements, $\mathbf{R^X}$ = ($\nu_{\mathbf{1}}^{\mathbf{X}}$ ... $\nu_{\mathbf{r}}^{\mathbf{X}}$ ... $\nu_{\mathbf{n^X}}^{\mathbf{X}}$), where n^X indicates the total number of measurements in trace X. Every measurement is organized as a vector of the RSS value according to the AP index, $\nu_{\mathbf{r}}^{\mathbf{X}}$ = ($\nu_{\mathbf{r,1}}^{\mathbf{X}}$ ... $\nu_{\mathbf{r,p}}^{\mathbf{X}}$ ... $\nu_{\mathbf{r},\mathcal{M}}^{\mathbf{X}}$).

Due to the randomness of the propagation channel, WiFi RSS is not stable even when the observer is stationary. While most APs are non-hearable at a given location, RSS from a hearable AP can also be missing from time to time. We need data preprocessing to smooth the RSS values and to impute RSS values for non-hearable and missing APs. The above can be effectively accomplished by a median filter:

$$\nu_{\mathbf{r,p}} = \mathbf{median}[..., \nu_{r-1,p}, \nu_{r,p}, \nu_{r+1,p}, ...] \quad (1)$$

The effect of the median filter is shown for the RSS sequence for one AP (from a walking observer) in Fig.3 and compared to that of the missing value imputation algorithm used in [1], [6]. We observe the median filter approach is much more effective than other linear filtering in removing random deep fading in the original signal while retaining the general contour of the RSS profile.

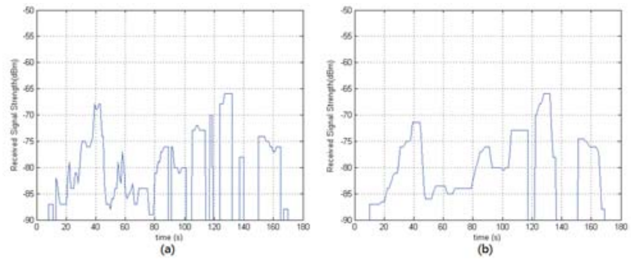


Fig.3 Preprocessing result of RSS sequence by:
(a) zero imputation in [1], [6] (b) median filtering in AWPS

3.3 BPs and ALSs Detection

A Correlation as Similarity Measure

From here on we will refer to $\hat{\mathbf{R}}$ as the filtered version. The key to our map construction algorithm is that repeated path segments exist in the reference trace since the surveyor or users walk randomly in the coverage area over a period of time. Repeating path segments can be identified from similarity in the RSS values. Detection of intersections is done based on the fact that repeating path segments can become disjoint when the surveyor or users move forward to or come in from different directions at a given intersection. Our approach is to detect intersections by detecting points at which repeated path segments become disjoint. If all intersections are detected, then the corridor segments between them can be detected.

We identify repeated path segments by finding similarity in the RSS measurements recorded. We use the Pearson product-moment coefficient as the similarity metric. The correlation value between the r^{th} measurement in $\mathbf{R^X}$, (ν_r^X) and the s^{th} measurement in $\mathbf{R^Y}$, (ν_s^Y), is

$$C_{r,s}^{\{X,Y\}} = corr(\nu_{\mathbf{r}}^{\mathbf{X}}, \nu_{\mathbf{s}}^{\mathbf{Y}}) = \frac{\sum_{p=1}^{\mathcal{M}} [(\nu_{\mathbf{r,p}}^{\mathbf{X}} - \mathbf{m_r^X})(\nu_{\mathbf{s,p}}^{\mathbf{Y}} - \mathbf{m_s^Y})^T]}{\mathcal{M}\sigma_r^X \sigma_s^Y}, \quad (2)$$

where $\mathbf{m_r^X}$ stands for the mean RSS value of measurement ν_r^X over all $\mathcal{M}$ APs in the coverage area:

$$\mathbf{m_r^X} = \frac{\sum_{p=1}^{\mathcal{M}} \nu_{r,p}^X}{\mathcal{M}} \quad (3)$$

$$\sigma_{\mathbf{r}}^{\mathbf{X}} = \sqrt{\sum_{p\epsilon\mathcal{M}} (\nu_{r,p}^X - m_r^X)^2}. \quad (4)$$

Mobility patterns can be observed in a correlation matrix. A stationary user or two users stationary at the same location is expected to generate a rectangularly shaped cluster of high or correlation values. Two pathways traversing the same corridor segment will create an elongated high correlation pattern pointing at approximately 45° if both are heading in the same direction and 135° if in the opposite directions. We illustrate several examples of correlation patterns in Fig.4.

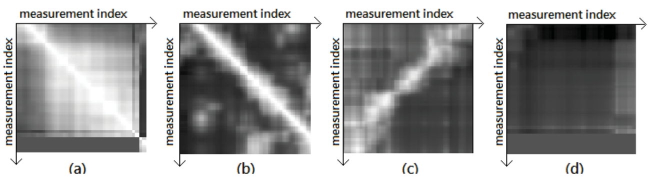


Fig.4 Examples of correlation patterns indicating users are: (a) stationary at the same location (b) in the same corridor segment heading in the same direction (c) heading in opposite directions (d)in unrelated locations

B Adaptive Correlation Pattern Recognition (ACPR)

The purpose of Adaptive Correlation Pattern Recognition (ACPR) is to determine the breaking points (BPs), which are the starting and ending points of high correlation patterns, for detecting intersections in the floor map. For ACPR, $\mathbf{C}$ is first quantized into a binary matrix $\mathbf{Q}$ by a threshold. Fig. 5 illustrates the differences between $\mathbf{C}$ and $\mathbf{Q}$.

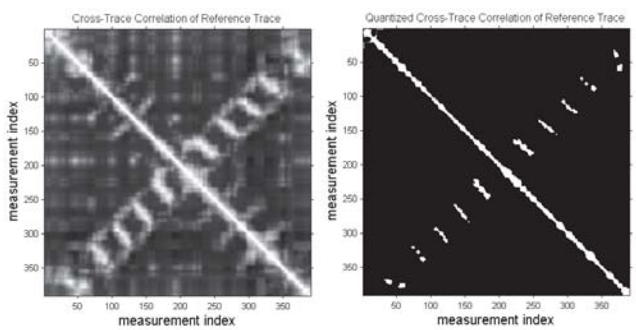


Fig.5 Comparison of Quantization result

We observe that most of the noise patterns are removed by the quantization process. However, incidental discontinuities in the correlation pattern still exist and may contribute to superfluous BPs. A pattern closing algorithm is applied to remove incidental discontinuities. Pattern closing consists of two operations: dilation ($\oplus$) followed by erosion ($\ominus$).

$$\mathbf{Q_{closed}} = (\mathbf{Q} \oplus \mathbf{mask}) \ominus \mathbf{mask} \tag{5}$$

$$\mathbf{Q} \oplus \mathbf{mask} = \bigcup_{\forall m \in mask} \mathbf{Q_m} \tag{6}$$

$$\mathbf{Q} \ominus \mathbf{mask} = \bigcap_{\forall m \in mask} \mathbf{Q_{-m}}, \tag{7}$$

where a square matrix is used as the mask.

The quantized correlation pattern is expanded by dilation to remove the cleavages which may accidentally break a high correlation pattern. The pattern is restructured into the original size by erosion afterward. The effect of the proposed pattern closing is demonstrated in Fig.6.

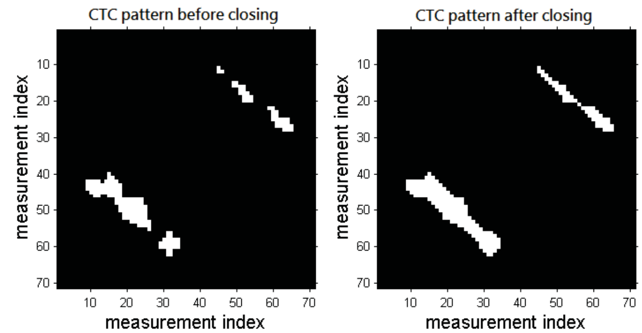


Fig.6 Comparison of pattern closing result

The proposed ACPR detects a high correlation pattern by adaptive searching. The search starts from $(0, 0)$ of $\mathbf{Q}^{\{X,Y\}}$ and continues with a zigzag path. The scanning proceeds until a high correlation value is found at (r, s), which equals 1 due to the quantization steps. Counter variables for the column and row, r_0, s_0, are initialized as 0. Then the sum of the sub-matrix $(r : r + r_0, s + s_0)$ is calculated to detect whether a high correlation pattern exists in the next column. r_0 is increased by 1 if the sum of the sub-matrix exceeds a certain threshold. The same procedure is applied to rows and s_0 by summing the sub-matrix, $(r + r_0, s : s + s_0)$. $r, r + r_0$ and $s, s + s_0$ are labeled with a BP_{ID} which essentially indicates the starting or ending time of a repeated path segment in $\mathbf{R^X}$ and $\mathbf{R^Y}$. The measurements between two break points are labeled with a unique ALS ID.

Algorithm 1 Adaptive Correlation Pattern Recognition (ACPR)

procedure ACPR($Q^{\{X,Y\}}$)
▷ Detect ALSs and BPs in $Q^{\{X,Y\}}$
$x_0 = 0$; $y_0 = 0$; $ID_{BP} = 0$;
for r = 1:n^X **do**
for s = 1:n^Y **do**
if $Q^{\{X,Y\}}_{r,s}$ = 1 **then** $indicator = 1$
while $indicatior \neq 0$ **do**
if $sum(Q_{r:r+r_0,s}) > threshold$ **then**
$r_0 = r_0 + 1$; $indicator = 1$;
else $indicator = 0$;
end if
if $sum(Q_{r,s:s+s_0}) > threshold$ **then**
$s_0 = s_0 + 1$; $indicator = 1$;
else $indicator = 0$;
end if
end while
$BP[ID_{BP}] = r; ID_{ALS} += 1$
$BP[ID_{BP} + 1] = r + r_0; ID_{BP} += 2;$
end if
end for
end for
$Q_{ID_{BP}:ID_{BP+1}} = ID_{ALS}$;
end procedure

C Break Points Purification

Assume K_1 BPs are detected over the super-trace $\hat{R}$ in ACPR. Each BP is identified by its index in $\hat{R}$. Since a corridor segment may be repeated multiple times in $\hat{R}$, multiple high correlation patterns are generated. Looking downward (or sideway) at the correlation matrix, the end point of a corridor segment may produce multiple end points of a high correlation patterns, each occurring at a slightly different time because of variations in the WiFi signal. BP purification is done to remove BP that are very close to each other in time. BPs with similar sequence index are clustered and labeled with the same ID_{BP}.

Algorithm 2 Break Points Purification

```
procedure BPREMOVE(BP_redundant)
                                   ▷ Detect and remove redundant BPs
    for ID_BP = 1 : end do
        diff[ID_BP] = BP[ID_BP + 1] − BP[ID_BP]
        if diff[ID_BP] < threshold then
            BP[ID_BP + 1] = BP[ID_BP]
        end if
    end for
end procedure                      ▷ Return purified discrete BPs
```

After the BP purification, we assume that K_2 BPs are left, and these are used to divide the RSS trace into $K_2 + 1$ segments, each potentially representing a unique atomic location segment in the coverage area.

D Atomic Location Segment Purification

The high correlation patterns are recognized as blocks by ACPR. Each high correlation pattern occurs when an ALS is repeated. Assume L_1 ALSs are detected in ACPR. Every ALS is labeled with an ID_{ALS}. We apply ALS Purification to unify the ID_{ALS} for repeated corridor segments and to decompose ALS consisting of multiple corridor segments. ALS Purification first examines the Correlation Pattern in the horizontal direction. If there exists two ALSs sharing the same starting and ending BP, we assign the same ID_{ALS} to the two ALSs. In case the starting BPs of the two ALSs (ALS_1 and ALS_2) are the same while the ending BPs are not, the longer ALS will be decomposed into two shorter ALSs. That is, $ALS_1 = [ALS_2\, ALS_{L_1+1}]$, where ALS_{L_1+1}

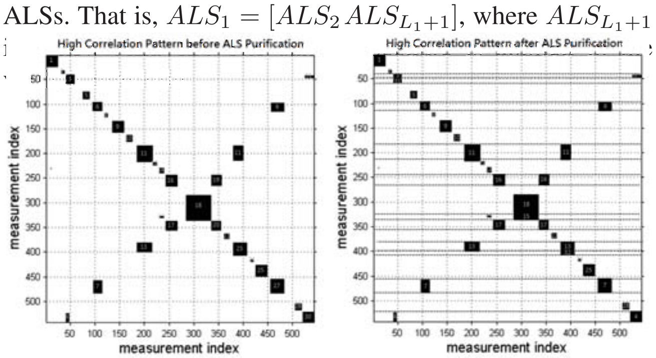


Fig.7 Re-labeled High Correlation Pattern

Algorithm 3 Atomic Location Segment Purification

```
procedure ALSPURI(ALS_redundant)
                                   ▷ Detect and re-label redundant ALSs
    for r = 1:n^X do
        for s = 1:n^Y do
            if starting_ALS_a = starting_ALS_b then
                if ending_ALS_a = ending_ALS_b then
                    ID_ALS_a = ID_ALS_b
                else
                    label(ALS_a − ALS_b) = ID_ALS_{L+1}
                end if
            end if
        end for
    end for
end procedure                      ▷ Return purified unique ALSs
```

3.4 Graphical Map Construction

A graphical map is constructed by Planar Embedding Algorithm described in [1]. We label the interesting locations as POIs to enrich the information of the graphical map.

A Points of Interest (POIs)

Points of Interest (POIs), such as lecture rooms or attractions, are labeled on the map automatically by using a second reference trace, $\mathbf{R^{r2}}$. Single-trace Pattern Recognition (STPR) is applied on $\mathbf{C^{\{r2,r2\}}}$ to examine the mobility of $\mathbf{R^{r2}}$. The user is stationary at time i for j seconds if $\mathbf{R^{r2,r2}_{i,i+j}}$ is high. Single-trace Correlation (STC), is calculated by a moving average across the diagonal of $\mathbf{C^{\{r2,r2\}}}$. Fig. 7 shows an example of STC. POIs are detected by 50% of the rising edges since STC starts to stabilize when user stops.

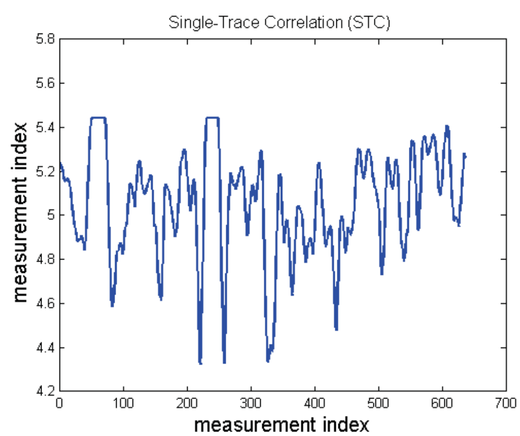


Fig.8 Single-Trace Correlation Pattern

3.5 Radio Map Construction

A radio map is the fingerprint for online localization. It is a description of the distribution of RSS values at different locations in the coverage area. The description can be parametrized or data-driven. In AWPS, for simplicity, we first assume the radio map is made up of a discrete set of location points and the RSS distribution for each AP

Algorithm 4 Single-Trace Pattern Recognition (STPR)

procedure STPR($\mathbf{R^{r2}}$) ▷ Detect POIs from $\mathbf{R^{r2}}$
  $C = C^{r2,r2}$
  ▷ Use single-trace correlation as the metric
  for $c = 1 : m^{r2}$ **do**
    $STC[count] = \frac{C[c-M:c+M, c-M:c+M]}{M}$ ▷ M is the size of the moving average mask
    $diff_{1st}[c] = STC[c+1] - STC[c]$
    ▷ Calculate the first difference of the STC.
  end for
  for $c = 1 : m^{r2}$ **do**
    $diff_{2nd}[c] = diff_{1st}[c+1] - diff_{1st}[c]$
    ▷ Calculate the second difference of the STC to detect troughs and peaks.
  end for
  $tough = find(diff_{2nd} > 0)$;
  $peak = find(diff_{2nd} < 0)$;
  $POIs = Measurement\ index\ at\ 50\% rising\ edge$
end procedure ▷ Return index of potential POIs

at each location point can be parametrized as a Gaussian distribution:

$$p(\nu \mid l_i)) = \frac{exp(- \mid \nu - \nu_{m,i} \mid / \sigma_{m,i})^2}{\sqrt{2\pi}\sigma_{m,i}} \tag{8}$$

where $\nu_{m,i}$ and $\sigma_{m,i}$ are the mean and variance of the RSS of the m^{th} AP at location l_i respectively.

The repeated ALSs in the reference trace and in crowd-sourced measurements are used to generate the RSS distributions. Repeated ALSs must first be unified there are the same number of RSS measurements presumably collected at the same set of location points distributed presumably uniformly along the corridor segment. Instead of simple interpolation, we accomplish the above by the Dynamic Time Warping (DTW) and ALS length alignment algorithm. DTW is applied to measure the similarity between two ALSs with different walking speeds. A cost function, $\mathbf{D}$ is computed based on the Euclidean distance of the measurements between two repeated ALS^{rep1} and ALS^{rep2}:

$$D(i,j) = \sqrt{\sum_{m \epsilon \mathcal{M}} (\nu_{r,m}^{rep1} - \nu_{s,m}^{rep2})^2}. \tag{9}$$

Then an algorithm is used to align the length of repeated ALS. It first initializes counter variables $r = s = 0$. It searches for the minimum cost between $D(r+1, s+1)$, $D(r+1, s)$ and $D(r, s+1)$. If $D(r+1, s+1)$ is the minimum cost, it implies the user is walking at the same speed in the two repeated ALSs at times r and s. No alignment is needed in this case. If $D(r+1, s)$ is the minimum cost, it implies that the user is walking slower between time r and $r+1$ in ALS^{rep1} than at time s in ALS^{rep2}. Then the ALS^{rep1} is compressed by taking the average of ν_r^{rep1} and ν_{r+1}^{rep1}. Similarly, the ALSs is compressed by taking the average value of ν_s^{rep2} and ν_{s+1}^{rep2} if $D(r, s+1)$ has the lowest cost. The search continues until the length of ALS^{rep1} and ALS^{rep2} are the same.

Algorithm 5 Atomic Location Segments Alignment

procedure ALSCOMBINE($ALS_{redundant}$)
  ▷ Align the length of repeated ALSs
  $D = DTW(ALS^{rep1}, ALS^{rep2})$
  $r = s = p = 0$
  for $r = 1 : size_{rep1}; s = 1 : size_{rep2}$ **do**
    if $D(r+1, s) = minimun$ **then**
      $ALS^{rep1}(r) = mean(ALS^{rep1}(r)\&(r+1))$
    end if
    if $D(r, s+1) = minimun$ **then**
      $ALS^{rep2}(r) = mean(ALS^{rep2}(s)\&(s+1))$
    end if
    $p = p + 1$
  end for
end procedure ▷ Return the purified ALSs

3.6 Localization

During the online stage, a real-time RSS measurement, ν, is recorded at an unkown location. The posterior probability of the user location can be calculated by Bayes rule:

$$p(l_i \mid \nu)) = \frac{p(\nu \mid l_i)(p(l_i))}{p(\nu)} \tag{10}$$

By using the likelihood function, $p(\nu \mid l_i)(p(l_i))$, a MAP estimator is implemented to predict the user location, L:

$$\hat{X} = \arg\max p(\nu \mid l_i)(p(l_i)) \tag{11}$$

During the online stage, unlabeled RSS measurements can be added to the database as crowdsourced traces. Cross correlation of a new crowd-sourced trace with existing traces can be computed, and individual segments in the new trace are identified. RSS measurements in each segment can be used to update the radio map for the corresponding segment. The system can also regenerates the floor map by the algorithms presented before, updating the floor map when a new ALS or intersection is identified.

4. Experiment Result

Experiment is conducted in a grid-like area of $120m * 21m$ located on the 2^{nd} floor of a university academic building. The actual floor map is shown in Fig.7. A Samsung Android Tablet $GT - P3100$ is used for collecting the traces. Two reference traces, R^{r1} and R^{r2} are recorded to bootstrap the system. The total offline surveying time is about 22 minutes, which, as shown in Table 2, is much shorter than the surveying times reported from previous systems. The computational efficiency of ACPR, which might be of concern during the crowd-sourcing phase, is 2.7 times better than the algorithm in [1], $0.0353\, Trace/s$ compared to $0.0979\, Trace/s$.

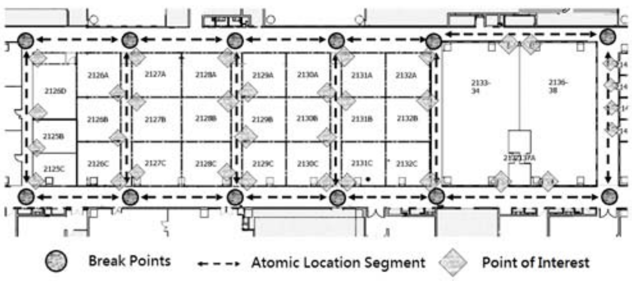


Fig.9 Experiment area of the AWPS

SYSTEM	Surveying Time(minutes)	Covered area
Proposed AWPS	$21'16''$	$120m * 21m$
$A.Varshavskya$[9]	$1098'$	$70m * 23m$
$Z.VictoriaYing$[11]	$75'$	$46.5m * 21m$

Table 2: Comparison of surveying time

The surveying time is reduced since only two traces are used for offline map construction. A floor map, which describes the relation between ALSs and BPs, is constructed. All POIs in the covered area including Laboratories and classrooms are also into the floor map by STPR. Corridors are recognized as ALSs and labeled with a unique ALS_{ID}. The intersections are identified as BPs. Fig.9 demonstrates the map constructed by AWPS. In total, 12 BPs, 16 ALSs and 27 POIs are drawn in the covered area. Users are localized at the corresponding ALS during the online phase. Some ALSs are not walked purposely in the reference trace to test the environment adaptability of the system. Since crowdsourced traces are collected during the online phase, the un-recorded ALSs are detected after they are walked by users.

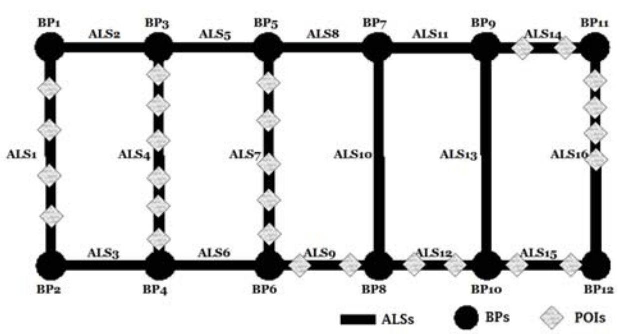


Fig.10 The constructed map of experiment area

5. Conclusions

AWPS is presented as a WiFi RSS-based indoor localization system. AWPS greatly simplifies the process of offline surveying for radio map construction. The system can be bootstrapped by two casual walks of a surveyor. The automatic floor map construction capability of the system eliminates the need for a reference floor map of the coverage area. We have also proposed better approaches to data smoothing and missing value imputations (median filter) and pattern recovery (closing filter). The enhanced intersection and corridor segment detection algorithms enable an accurate floor map to be automatically reconstructed. A parametrized radio map is developed for online localization. Continuous matching of crowd-sourced RSS traces will allow the radio map to be adaptively updated, and the floor map to be completed/updated if necessary. Point of Interest labels are added into the floor map and can be made use of for navigation applications.

References

[1] Luo, Xin, et al. "Automatic Floor Map Construction for Indoor Localization." AICT 2014, The Tenth Advanced International Conference on Telecommunications. 2014.

[2] Ouyang, Robin Wentao, AK-S. Wong, and Kam Tim Woo. "GPS localization accuracy improvement by fusing terrestrial TOA measurements." Communications (ICC), 2010 IEEE International Conference on. IEEE, 2010.

[3] Zhang, Victoria Ying, and AK-S. Wong. "Combined AOA and TOA NLOS localization with nonlinear programming in severe multipath environments." Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE. IEEE, 2009.

[4] Venkatraman, Saipradeep, and James Caffery. "Hybrid TOA/AOA techniques for mobile location in non-line-of-sight environments." Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE. Vol. 1. IEEE, 2004.

[5] Al-Jazzar, Saleh, Mounir Ghogho, and Desmond McLernon. "A joint TOA/AOA constrained minimization method for locating wireless devices in non-line-of-sight environment." Vehicular Technology, IEEE Transactions on 58.1 (2009): 468-472.

[6] Ouyang, Robin Wentao, AK-S. Wong, and Chin-Tau Lea. "Received signal strength-based wireless localization via semidefinite programming: Noncooperative and cooperative schemes." Vehicular Technology, IEEE Transactions on 59.3 (2010): 1307-1318.

[7] Wang, Xiaoli, AK-S. Wong, and Yongpmg Kong. "Mobility tracking using GPS, Wi-Fi and cell ID." Information Networking (ICOIN), 2012 International Conference on. IEEE, 2012.

[8] Evennou, Frédéric, and François Marx. "Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning." Eurasip journal on applied signal processing 2006 (2006): 164-164.

[9] A. Varshavskya, E. de Laraa, J. Hightowerc, A. LaMarcac, and V. Otsason, "Gsm indoor localization," Pervasive and Mobile Computing, vol. 3, 2007.

[10] Ferris, Brian, Dieter Fox, and Neil D. Lawrence. "WiFi-SLAM Using Gaussian Process Latent Variable Models." IJCAI. Vol. 7. 2007.

[11] Zhang, Victoria Ying, Albert Kai-sun Wong, and Kam Tim Woo. "Histogram based particle filtering with online adaptation for indoor tracking in WLANs." International Journal of Wireless Information Networks 19.3 (2012): 239-253.

A Reconfigurable System Learning for Data Classification Using Parallel Processing

E. M. Moreira[1], C. D. Maciel[2], E. M. Moreira[3], and F. S. Zanoni[2]
[1]Federal Institute of Education, Science and Technology of São Paulo, São João da Boa Vista, São Paulo, Brazil
[2]Signal Processing Laboratory, Department of Electrical Engineering / University of São Paulo, São Carlos, São Paulo, Brazil
[3]Systems Engineering and Information Technology Institute / Federal University of Itajubá, Itajubá, Minas Gerais, Brazil

Abstract— *This paper presents a System Learning with a task scheduler, which makes possible the utilization of several classification and validation methods, allowing the distribution of tasks between the module systems. This architecture is structured of such way that the classifications obtained through a specific technique can be reutilized in parallel by the same algorithm or by other techniques, producing new classifications through the refinement of the results achieved and expanding the use in databases with different characteristics. The results demonstrated that through a learning system, the complexity of the analysis of great databases is minimized, allowing to verify basis with different structures and to increase the methods applied in the analysis of each structure. It favors the comparison between the methodologies and provides more reliable results.*

Keywords: Machine Learning; System Learning; Clustering; Data Partitioning.

1. Introduction

The procedure called *data clustering* constitutes in a complex technique, which has being studied in several areas aiming to find patterns in data, considering different theories and methodologies [1]. This technique is focused on the development of partitioning procedures of data sets in order to join similar objects, generating subgroups [2]. In this partitioning procedure, there is a significant relationship between arguments, metrics of similarities and structure of data. Considering the specific characteristics of each database, it is not trivial to reach a satisfactory result in the partitioning procedure in many cases. For this reason, the manipulation or configuration of these elements is a fundamental pre-requisite to obtain suitable results. In this way, aiming the elaboration of subgroups through the evaluation of important properties to the total group, it is necessary a previous knowledge regarding the context in which the major group is involved in each analysis; its specific insertion areas; and its general relevance, independently of any subdivisions. Furthermore, it is interesting to identify elements that, in fact, have important characteristics to the large group, in such way these subdivision can contribute to the understanding of the inherent properties to the same subgroups as well as the original total group. Usually, these studies require great statistic evaluation and significant technical analysis, including, in several times, a contribution of subjective studies from specialists of the subject evaluated in each work.

In turn, the information of interest are contained in large database, wherein the data manipulation involves specific tasks with higher computational overhead. During the data manipulation, the results obtained by a classification methodology are frequently sensitive to a specific data base. Several works are focused on distincts techniques of classification which are dependents of parameters related to the data originated from the respective study [3], [4], [5]. Considering the peculiarities of each base and the impact referent to a small alteration in the method employed regarding the final partitioning, it is important to make a rigorous evaluation of the classifications obtained through these partitioning tasks.

In this context, it is interesting to consider the possibility of creating a refinement method which uses the characteristics of several other methods, since there is no guarantees that a particular method will clustered the data correctly, according to the understanding of the specialists that are contributing with the subjective task of interpretation of the data. In fact, various papers present this difficult about the classification procedure [6], [3], [7].

As function of this fact, works that associated and/or compare different classification methodologies become essential procedures to the choice of the best data treatment, depending of the several factors inherent to each case [8], [6].

Considering the difficulties of creation of a unique methodology to a satisfactory classification of distinct data bases and the difficult of interpretation of the implications related to the large volumes of data in each base, the classification efforts can, frequently, to be correlated to a significant increase in the inherent computational cost [9].

In this way, the present work proposes the development of a parallel learning system that has the objective to administrate classification methods, partitioning refinement methods as well as the validation methodology of results to be applied in the data bases.

This article is structured in the following way: the next section presents the structure of the proposed system with the respective implemented methodologies; subsequently, it is presented the section focused on the experiments and results obtained. In the last section, the final considerations and proposal to the continuation of this work are presented.

2. Materials and Methods

2.1 System Learning Proposal

The System Learning described in this document presents four modules: `Initialization Module`, `Validation Module`, `Refinement Module`, and `Special Scheduling Module`. In each module can be used several algorithms to meet their goal.

Figure 1 shows the design of this system learning, allowing you to see how these modules interact. The activities of each module will be detailed in the following subsections.

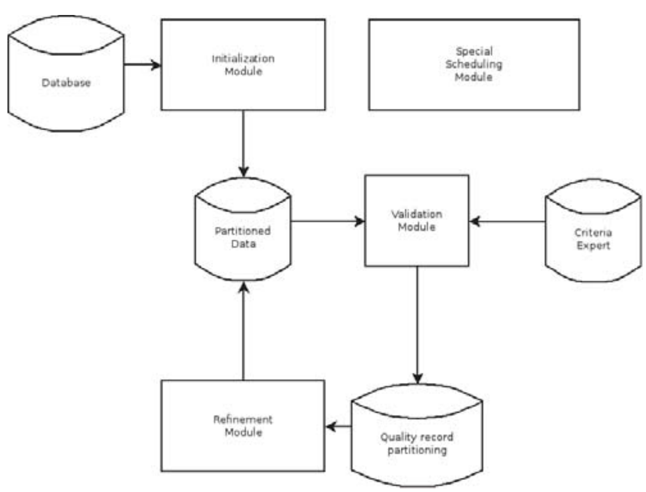


Fig. 1: Figure shows the structure of the system and the form in which the modules are related. The data set is firstly treated by the `Initialization Module` generating the first set of partitioned data. The partitioned data sets are assessed by the `Validation Module`, which employs different criteria and expert rules to record the quality of the partition. Parallely to this process, `Refinement Module` generates more partitions to be evaluated from the partitioned data set by `Initialization Module` and Refinement own module. `Special Scheduling Module` is responsible for the administration of the activities

2.1.1 Initialization Module

The goal of the `Initialization Module` is to prepare the original data to be classified. This module create an initial partitioning and organize the data in a pattern understood by the *Validation* and *Refinement* modules. In the `Initialization Module` were implemented four algorithms to generate initial partitions: *Binary Tree*, *Random*, *Quasi-random*, and *Fiedler Method*, that is a spectral method.

Noteworthy is the method of *Binary Tree* which, due to the analysis that is performed on the data during its execution, creates a partition with great potential to be the end result or next to it. Methods *Random* and *Quasi-random* are algorithms based on random mechanisms which, although simple, may achieve good results due to parallel processing and integration between modules for validation and refinement.

All classification produced in `Initialization Module` or `Refinement Module` receives a value that represents the quality of the created partition. To this value was given the name of **Quality Attribute**. The way of this attribute is calculated can vary with the type of data or according to the methods used.

The *Binary Tree* Method utilizes the *Binary Tree* data structure for construction of partitions. Each partition is represented by a tree that stores alike objects. The algorithm starts with the construction of a similarity matrix from the attributes of the original data file. When analyzing a set of data, or attributes, such as dimensions of a multi-dimensional space, the description of each figure corresponds to a point in that space. Therefore, the distance between pairs of objects can be used as a degree of similarity between them. In implementing this method, it was used the Euclidean distance between the geometric data [10].

The *Binary Tree* method performance basically the construction of a dendogram [11], that are convenient ways of depicting pairwise dissimilarity between objects. The construction of the *Binary Tree* is done in order *bottom/up*. Initially, it has n distinct sets or partitions, where n is the cardinality of the original data set. From the similarities matrix, two closest objects are identified. These objects will be merged into a single tree and thus become the left and right subtrees of the new tree. Thus, the number of partitions decreases by one. The matrix of similarities is redone by excluding the rows and columns referring to the objects, and by inserting a new line whose reference point in space is the average distance between the two objects chosen for the other partitions. This method is known as *Average Link* [12].

Importantly, the definition of the new coordinate location of the partition can be accomplished in several ways, for example, the use of the coordinate of the midpoint between the objects belonging to the partition or by using the coordinate of the object, which belongs to the partition that is more close relative to the other partitions, a method known as *Single Link* [13]. These changes were implemented and the proposed system run in parallel by providing, to `Validation Module`, distinct partitions for analysis. The procedure for choosing the partitions closer repeats until it reaches the desired number of partitions.

The *Random* method is very simple. From an initial seed, there is a random choice of the objects that will compose each partition. Obviously, this method does not start of any knowledge of the data and therefore produces clusters whose **Quality Attribute** is inferior. The biggest advantage of this method, besides its simplicity, is the speed with which the partitions are created from new seeds of random numbers.

Turn, the semi-random method is a variation of the *Random* method. Despite using random choices, the objects are not chosen within the entire universe of the original data. A pre-assessment is conducted to the choices that are made of objects into subsets with the highest similarity among its components. The starter sets are constructed from a matrix of similarities. Initially, two objects are chosen randomly for each subset. The distance between these objects will be used as reference in the choice of other objects. The remaining objects are also randomly chosen, however, are inserted into the set consisting of the smallest distance between objects in the set and itself. Every time an object is inserted in the set, the reference center of the set is recalculated to be used in the next comparisons.

In 1973 and 1975, M. Fiedler published papers on the properties of the Laplacian matrix of the eigensystems [14], [15]. His studies began with the contribution of the paper by Anderson and Morely on the eigenvalues of the Laplacian matrix [16]. In 1990, Pothen, Simon, and Liou published a paper Applying Fiedler's ideas to the field of clustering [17], being the beginning of the theory for spectral graph partitioning. The implemented version of this method partitions the dataset into two subsets, based on positive and negative values of the eigenvector matrix Laplacian. The Laplacian matrix was obtained from the matrix of similarities, so it is a weighted matrix corresponding to a complete graph. In order to achieve the desired number of partitions, each subset was created recursively subdivided by applying the same method again.

After calculating the **Quality Attribute** of a partitioning, the `Validation Module` records that information in the '`Register Quality Partitioning`'. At the criterion of the specialist, we can set minimum quality values for a partitioning to be handled by the `Refinement Module`. However, validation of an entire partition from the `Initialization Module` must necessarily pass through the `Refinement Module`.

2.1.2 Validation Module

Once a classification has been created by `Initialization Module`, or enhanced by the `Refinement Module`, it is necessary to check the quality of the partition. Usually, this task requires a specialist who will use the results of partitioning. However, based on information from these professionals, it is possible to create a rule base (`Criteria Expert`) that automates this process or at least deletes the partitions that are below a minimum standard of quality. For example, instead of using all attributes related to an object of the base for calculating the Euclidean distance, specific information about the data could be used to optimize the calculation of similarity. Thus, the specialist will assess only those partitions that have achieved a good value on **Quality Attribute**.

For the tests performed with the data in this article, the **Quality Attribute** used was the sum of the mean metric distances among all pairs of objects in the same partition. Obviously, other statistical parameters can be used according to the rules laid out on the basis of '`Criteria Expert`'.

After calculating the **Quality Attribute** of a partitioning, the `Validation Module` records that information in the '`Register Quality Partitioning`'. At the discretion of the specialist, minimum quality values can be set for a partitioning to be handled by the `Refinement Module`. However, validation of an entire partition from the `Initialization Module` must necessarily pass through the `Refinement Module`.

2.1.3 Refinement Module

Refining is to improve the quality of a partitioning previously realized. This is done by exchanging objects between partitions. This exchange helps to increase the value of **Quality Attribute**.

Once the exchange is performed and a new classification obtained, the `Validation Module` verifies the impact of this exchange. When the exchange process is successful, the partitioning obtained is available as a result achieved and as a started point to a new refinement. On the other hand, if the **Quality Attribute** value indicates a quality deterioration of original classification, it is discarded.

The integration of `Validation Module` and `Refinement Module` provides application of the evolutionary method that is the base of the genetic algorithms. The evolutionary paradigm has been used in research of the machine learning.

According Zhang [18], the steps performed by an evolutionary algorithm have two basic stages: **initialization**, which from a default configuration or a random configuration generates the first population, and **generation**, that is based on three functions: fitness, crossover and mutation.

In this work, the population of individuals is represented by a classification obtained and its **Quality Attribute**. The main genetic operators are crossover and mutation. The crossover operation is implemented through specific techniques developed in the `Refinement Module`, described below. The mutation operation is not used at this work, since changes always occur in the same individual. Fitness function is used to measure the quality of an individual and, as already discussed, this is done by the `Validation`

Module, through the calculation of **Quality Attribute**.

The main problem of the genetic algorithm is its computational overhead, especially due to the fitness function, which has to be repeatedly assessed using the evolutionary process. This signals the importance of the application of this technique on a cluster of computers [19].

As the essence of Refinement Module are the mechanisms that promote the exchange of objects, several mechanisms can be used for this purpose, an example used as base is the classic method *k-medoid* [20], because it is based on the most centrally located object in a cluster, it is less sensitive to outliers in comparison with classic method *K-means* [21].

For the initial composition of this system, three methods was created, *Radius Coverage*, *Query by Range*, and algorithm variation of the Kernighan and Lin [22] for partition graph.

By means of partitions defined in the Initialization Module, or previous results of partitioning, this method computes radius of coverage from of centroid of each partition, depending of method used, that is, the radius is distance between the center point and the most distance object that belongs to partition. In this sense, it can identify spheres involving objects of each partition, however, it is possible that objects of different partitions are reached by the radius of other partitions. These objects are in the areas of intersection between two or more partitions. From the identification of candidate partitions, ie, partitions which their coverage radius reach the object, which is chosen the partition where distance between the object and its *centroid* is the smallest. The object is displaced from its original partition to the partition chosen. A variation of this method is the possibility of increasing the radius of coverage to identify other objects in the intersection set. This increase variations have been implemented. The rate of increase is directly related to the value of **Quality Attribute**, i.e., dependent on the data of the problem.

The Coverage Radius Method performs a query on all data, identifying nearest *centroid*. If the *centroid* found is not the same, the method transfers the data to the partition of this *centroid*. After method verify all data, the system accesses the Validation Module for a new **Quality Attribute**. An interesting feature of this method is that, depending on the data set, there is a convergence of found partitions, since as they happen refinements, *Radius Coverage* tend to decrease, helping the Special Scheduling Module in the criterion stop.

Importantly, these methods were applied in sets ever built by the algorithms implemented in the Initialization Module. Thus, the choice of candidate objects to exchange the set was made from range queries in relation to the cluster center (centroid or medoid). Also, experiments were performed by increasing the radius of coverage with the aim of expanding the sets of intersection.

The Kernighan and Lin [22] proposed one of the earliest methods for graph partitioning, and more recent local improvement methods are often variations on their method. This method works primarily evaluating the gain in reducing the cost cuts between partitions during an exchange between pairs of vertices of different partitions. The algorithm developed in this study randomly selects two partitions and chooses an object that is farthest from the centroid of the first partition and does the same on the second partition neighbor, selecting the most distant object of its respective centroid. After selection, the algorithm calculates the gain to be obtained from the **Quality Attribute** if this change is made. If there is no gain, a new object is chosen from the second partition, ie, the second most distant object of its centroid, and so on. When two objects are chosen, a new object from the first partition is selected and the procedure repeats. When there are more gains the algorithm stops. It is important to note that the algorithm stops before replacing all the objects (which would be to simply rename the two partitions), as by changing all the objects, there would be no gain.

At the end of the algorithm, a new partitioning is obtained and therefore is ready for a new analysis by the Validation Module.

2.2 Special Scheduling Module

As presented, the architecture of this system learning enables its implementation on a parallel computer or a distributed architecture, for example, a cluster of computers.

The Special Scheduling Module features, in this context, two main functions: scheduling jobs produced by the other modules and identifying the time to stop the system learning.

The Initialization Module maintains a continuously running process (*trigger process*). Basically, this process is responsible for communication with the Special Scheduling Module. Its function is to wait for the signal of availability (when there is any ready processor to perform the task) and create a new task (the execution of an algorithm for a specific method, which has not yet been used, or the execution of a method already used with new values for its configuration attributes).

In Special Scheduling Module there is a *Scheduler Process*. This process maintains a multilevel priority queues system. Tasks are grouped into three classes of priority: tasks of the Validation Module, tasks of the Refinement Module and tasks of the Initialization Module. Figure 2 presents the structure of multilevel queues. The queue refers to the processes of Validation Module has the highest priority, i.e. no process to other queues that are waiting will be choose if there is any process in this queue. This is necessary because, since there are classifications already made, the validation should be performed with priority over the development of new partitioning. The queue relating to the `Initialization Module` has the lowest priority, i.e., its processes can only be chosen if the other queues are empty. It is important to note, however, that in the initial operation of the system learning, all queues are empty and therefore the tasks of generating initial classifications will be executed, because there will be no ratings to be validated or refined, with their queues being empty.

The `Validation Module` has a unique type of process (*Quality Evaluator*), which is responsible for assessing the quality of a partitioning obtained. Every time a process of `Initialization Module` or `Refinement Module` finishes its execution, there is a new partition available to evaluation. In this context, a new process *Quality Evaluator* is created to analyzed the classification obtained. This process is thus inserted into their priority queue, waiting for the availability of a processor to select the scheduler.

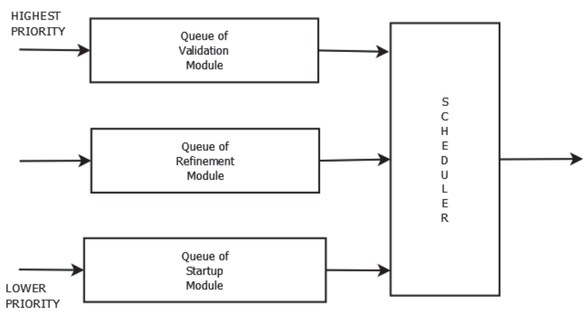


Fig. 2: Scheduling with Multilevels Queues - `Special Scheduling Module` defines priorities in the administration of the activities of the system. `Validation Module` has higher priority system that `Refinement Module` and this has a higher priority than the `Initialization Module`

An important issue to be considered concerns the criteria that the `Special Scheduling Module` uses to stop scheduling new tasks. When the system verifies that the partitioning presented by `Refinement Module` has been produced by a previous method (in `Initialization Module` or another refinement by the same method), there is no need to submit it for processing because it would result in data replication, the unless it is used another refinement method.

Another stop criteria used verifies the analysis of historical values for the **Quality Attribute** using the same method of refinement. When the values surrounding a reference value, indicating the proximity of a local minimum, `Special Scheduling Module` interrupts the scheduling of new tasks. It is noted that approaches to identify the local minimum can vary depending on the used database.

Obviously there are databases which partitions produced in the `Refinement Module` is not repeated because of combinatorial explosion caused by the amount of information. Furthermore, the permutations obtained from each refinement can significantly change the **Quality Attribute** and, in this case, it is not possible to identify the approach of a local minimum. In these situations, it is possible to define as stop criterion a maximum number of refinements from a partition obtained by one method of the `Initialization Module`.

3. Experiments Results

In order to obtain comparison, it was elaborated the table 1 using Iris data set, which presents the values of the **Quality Attribute** initially obtained in the methods already implemented in the system proposed, in agreement with description in the section 2.1.1. This table presents also the **Quality Attribute**, in agreement with UCI repository [23]. This value is **2,82201**, which corresponds to the original partitioning of Fisher [24]. Considering the method of *Binary Tree*, it was obtained a **Quality Attribute** that is still better, **2,7228**. This result, which was considered significantly better than the original one, demonstrates the existence of superpositions between the classes.

After the generation of initial groups, these selections were utilized by algorithms of the `Refinement Module`. In agreement with previous discussion in section 2.1.3, the choice of the candidate objects to develop the exchange of partitioning was made, following the two methods described.

Table 1 presents the atualizated values of the **Quality Attribute** after the treatment of the **Refinement Module** with method of *Radius Coverage* and the method of *Query by Range*.

Data Set *Iris* - Quality Attribute			
Repository	*Binary Tree*	*Radius Coverage*	*Query by Range*
	2,69466	**2.70427**	**2.72368**
	Quasi-random	*Radius Coverage*	*Query by Range*
2,82201	**2.71932**	**2,70231**	**2.85162**
	Random	*Radius Coverage*	*Query by Range*
	5,68444	**2.71379**	**4.18003**

Table 1: Determination of the **Quality Attribute** realized by the `Validation Module`, which was applied in the partitionings generated by the methodologies of the initializing module in Iris data set

In the execution of the *Random* and Semi-Random methods, the results presented in the Table 1 corresponds to the better value obtained by the `Initializing Module` in **3.000.000**, considering the partitionings obtained presently.

Table 2 constitute abstracts of some results of the proposed system. The first column has the method utilized to the generation of the first partition as well as the index of correction encountered. The second column corresponds to the methods of refinement containing, to each one, two informations. The first one is the quantity of iterations that was required until to reach some stop criterium; the second one corresponds to the correct index of the better partition obtained in the evaluation.

Data Set *Iris*					
Initialization Module		Refinement Module			
Method	Index Correct	*Radius Coverage* Iteraction	*Radius Coverage* Index Correct	*Query by Range* Iteraction	*Query by Range* Index Correct
Binary Tree	91,33%	1	88,66%	2	93,33%
Random	54,66%	3	88%	6	55,33%
Quasi-random	90,66%	1	91,33%	2	92%

Table 2: Analysis of data set *Iris* after partitionings generated by the **Initialization Module** and treatment realized by the **Refinement module**

Corresponding to the information presented by the Table 2, it is possible highlight the refinement reached by the *Radius Coverage* Method in the partitioning generated by the *Random* Method, in which the selection founded is proximal of the best partitioning obtained by the others methodologies. Considering the method of *Query by Range*, the **Refinement Module** originated tasks to the **Special Scheduling Module**, starting from the own radius of approach of the original selections and increasing the radius step by step in **0,02**, in the distance of Euclides approach.

In relation to the stop criteria, the behavior was different in each one of the methods evaluated. In the method of *Query by Range*, the interruption occurred when a new selection did not decreased the value of quality assignment. In its time, in the case of the *Radius Coverage*, the interruptions, in all bases, happened when the result of the refinement produced a group already in previous partitions.

It is important to notice that in the method of *Query by Range*, the test to the *Radius Coverage* increase was of **15%** of the original radius of the class.

All the modules implemented were simulated sequentially e parallely to the performance analysis of the system. This experience was performed in a cluster of computers with the following structure:

- Four computers with following specifications:
  - Intel(R) processor Core(TM)2 Quad CPU 2,66GHz;
  - Cache L2 8Mb;
  - 1066 MHz Frontal bus;
  - 2 GBytes RAM.
- The network has the following features:
  - 100 Mbits/s Ethernet;
  - *Switch*: 3Com Baseline Switch 2948-SFP Plus 48-Port Gigabit (3CBL SG48).

The implementations of the programs were developed in the C++ language with the version 1.5.4 of library OpenMPI. The operational system was Linux Fedora 13 with kernel version 2.6.33.5-112.fc13.

It is clear that because of the parallel system characteristic and structure of the scheduler, it can possible to make various refinements on the data already partitioned while there are other tasks carrying out the first partition in others data sets.

To evaluate the performance of parallel system methods, it was calculated the Speedup, which measures the reduction factor of runtime on P processors and finally the efficiency that is determined by the ratio between the speedup obtained and the number of processors used [25].

To strengthen the contribution presented in this article, further illustrating the behavior of the implemented system, the **Special Scheduling Module** worked with 25 Databases for classification, at the same time. The databases were taken from the UCI repository (Iris, Zoo, Glass, Balance and Wine) and replicated five times, totaling 25 databases in parallel. Were carried on the following methods for each instance: Binary Tree, Random, Quasi-random, Radius Coverage, and Query by Range.

Table 3 shows the system behavior manipulating **6335** attributes.

Table 3: Performance Learning System in the cluster of computers

Clusters	Time	**SpeedUp**	**Efficiency**
1	32m13.849s	-	-
2	16m14.887s	1,98	99,18%
4	9m7.396s	3,53	88,32%

To standardize the data originated of distinct bases, a data modelling system was elaborated, which will favor the uniform scalability of the system. In this sense, the implementations of other methods, that were added to the system, should handling the data as described in the model created, requiring converting the original data to adjust and insert them in the system database.

4. Conclusion

The main feature of the system learning that was presented in this paper was its flexibility in the composition of algorithms, which must be employed in data classification. Moreover, through the configuration files it is possible to easily change the data sets.

The examples in this paper were chosen in order to demonstrate the accessibility involving the practical employment of the present methodology, that is, in using this system learning. With few changes, new methods may be included in each module of the system. Considering the results presented, the increase of processors permits increasing the data sets to be analyzed simultaneously.

In the results presented, this system is efficient because it searching good partition, even when the `Initialization Module` get a bad partition. The use of `Refinement Module` can correct a bad choice of initialization seeds. This example is showed in `Refinement Module` when it works with the results presented by Randon Method in `Initialization Module`.

In addition, a graphical user interface is being developed to permit in the configuration of this system by the users. In this way, the perspectives of application of this proposed in several areas of knowledgment are very auspicious, which can achieve great methodological advancements in various protocols utilized by research groups of several areas.

References

[1] L. Xutao, Y. Yunming, L. J. Mark, and K. N. Michael, "On cluster tree for nested and multi-density data clustering," *Pattern Recognition*, pp. 3130–3143, 2010.

[2] M. Meila, "Comparing clusterings - an information based distance," *Journal of Multivariate Analysis*, no. 98, pp. 873–895, 2007.

[3] Y. Liu, X. Wu, and Y. Shen, "Automatic clustering using genetic algorithms," *Applied Mathematics and Computation*, 2011.

[4] B. Zahraie and A. Roozbahani, "Sst clustering for winter precipitation prediction in southeast of iran: Comparison between modified k-means and genetic algorithm-based clustering methods," *Expert Systems with Applications*, 2011.

[5] M. E. Celebi, H. A. Kingravi, and P. A. Vela, "A comparative study of efficient initialization methods for the k-means clustering algorithm," *Expert Systems with Applications*, 2013.

[6] R. K. Blashfield, "Mixture model tests of cluster analysis: Accuracy of four agglomerative hierarchical methods," 1976.

[7] F. Cao, J. Liang, and L. Bai, "A new initialization method for categorical data clustering," *Expert Systems with Applications*, no. 36, pp. 10 223–10 228, 2009.

[8] G. W. Milligan, "An examination of the effect of six types of error perturbation on fifteen clustering algorithms," *PSYCHOMETRIKA*, 1980.

[9] W. Yan, U. Brahmakshatriya, Y. Xue, M. Gilder, and B. Wise, "p-pic: Parallel power iteration clustering for big data," *J. Parallel Distrib. Comput.*, 2012.

[10] E. Eirola, G. Doquire, M. Verleysen, and A. Lendasse, "Distance estimation in numerical data sets with missing values," *Information Sciences: an International Journal*, vol. 240, pp. 115–128, 2013.

[11] C. Iyigun and R. T. S. U. of New Jersey New Brunswick. Graduate School New Brunswick, *Probabilistic Distance Clustering*. Rutgers The State University of New Jersey - New Brunswick, 2008. [Online]. Available: http://books.google.com.br/books?id=GRWIu1jc884C

[12] O. I. R. Handl, J. Knowles, J. Knowles, M. Dorigo, and U. L. D. Bruxelles, "Ant-based clustering: A comparative study of its relative performance with respect to k-means, average link and 1d-som," 2003.

[13] L. Hubert, "Approximate evaluation techniques for the single-link and complete-link hierarchical clustering procedures," *Journal of the American Statistical Association*, vol. 69, no. 347, pp. 698–704, 1974.

[14] M. Fiedler, "Algebric connectivity of graphs," *Czechoslovak Mathematical Journal*, no. 23, pp. 298–305, 1973.

[15] ——, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory," *Czechoslovak Mathematical Journal*, no. 25, pp. 619–633, 1975.

[16] W. N. Anderson and T. D. Morley, "Eigenvalues of the laplacian of a graph," 1968.

[17] A. Pothen, H. D. Simon, and K. P. Liou, "Partitioning sparse matrices with eigenvectors of graphs," *SIAM Journal on Matrix Analysis and Applications*, vol. 11, no. 3, pp. 430–452, 1990.

[18] J. Zhang, Z. hui Zhan, Y. Lin, N. Chen, Y. jiao Gong, J. hui Zhong, H. S. Chung, Y. Li, Y. hui Shi, and Xian, "Evolutionary computation meets machine learning: A survey," *IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE*, pp. 68–75, 2011.

[19] P. Espejo, S. Ventura, and F. Herrera, "A survey on the application of genetic programming to classification," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 40, no. 2, pp. 121–144, march 2010.

[20] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for k-medoids clustering," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3336–3341, Mar. 2009. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2008.01.039

[21] *Data Mining and Analysis: Foundations and Algorithms*. Cambridge University Press, 2010.

[22] B. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *The Bell System Technical Journal*, vol. 2, no. 49, 1970.

[23] A. Frank and A. Asuncion, "Uci machine learning repository," 2010. [Online]. Available: http://archive.ics.uci.edu/ml

[24] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[25] I. FOSTER, "Designing and building parallel programs: Concepts and tools for parallel software engineering. mathematics and computer science division - argonne national laboratory," pp. http://www–unix.mcs.anl.gov/dbpp/text/node4.html, 1995.

Synchronous Emotion Pattern Recognition with a Virtual Training Environment

B. Senzio-Savino[1], M. R. Alsharif[1], C. E. Gutierrez[1], K. Yamashita[2]
[1]Department of Information Engineering, University of the Ryukyus, Nishihara-shi, Okinawa, Japan
[2]Graduate School of Engineering, Osaka Prefecture University, Sakai-shi, Osaka, Japan

Abstract - *The use of Brain-Computer Interface (BCI) has been increasing exponentially in the recent years due to the use of low-cost commercial Fast Fourier Transform (FFT) based EEG headsets. The aim of this paper is to present a method and the results of applying binary Density Based Support Vector Machine (DBSVM) Classifiers in a 3D virtual environment designed for interacting with EEG predefined signal patterns. The environment trains the classifier by taking 180 second EPOCHs and classifying them into a successful/unsuccessful attempt per test subject. The test had a success rate of 80% for some of the obtained classifiers. Further work includes the study of different classifier features and implementation of a dynamic classifier. The target is to recognize a specific emotion pattern generated by a subject after learning his common wave pattern generations with the use of a 3D training application.*

Keywords: BMI, MindWave, DBSVM, pattern recognition, Machine learning

1. Introduction

The use of FFT-BCI based 3D virtual environments has been implemented effectively in different application fields; it is rather a new technological tool, which may be exploited to enhance motor retraining [1].

A simple development with software like Unity3D can be combined with other assistance technologies or interfaces [2] in order to improve a specific system's performance with an end user's feedback.

These interfaces are either interacting with the preprocessed behavior signal output data stream from a processing IC with the aid of "slow adaptive algorithms" [3] like the TGAM1, or the RAW-EEG value obtained from an active electrode and then decomposed into a specific behavior according to the spectral density of the sample .

These main brain waves and frequency range for processing are specified in Table 1 [4].

There are algorithms that focus on detecting a specific emotion from the end user according to their mind-wave patterns.

Some of them can map and help to differentiate emotions like attention/inattention and meditation/uneasiness based on a simple beta/theta wave ratio [5].

Table 1: Basic human brain waves

Name	Frequency range [Hz]
Delta [δ]	0.5-2.75
Theta [θ]	3.5-6.75
Low alpha [l-α]	7.5-9.25
High alpha [h-α]	10.0-11.75
Low Beta [l-β]	13-16.75
High Beta [h-β]	18-29.75
Low Gamma [l-γ]	31-39.75
Mid Gamma [m-γ]	41-49.75

This paper describes the implementation and results of a Density Based Support Vector Machine Classifiers from a series of training data sets obtained from a 3D environment with the use of a different set of common-use kernels.

The classifiers' input data are a binary set of 180 long signal vectors (EPOCHs) corresponding to the preprocessed attention and meditation outputs from a one dry electrode EEG headset with the configuration as shown in Figure 1.

The aim of this research is to elaborate a method for recognizing mind behavior patterns for a synchronous password application development.

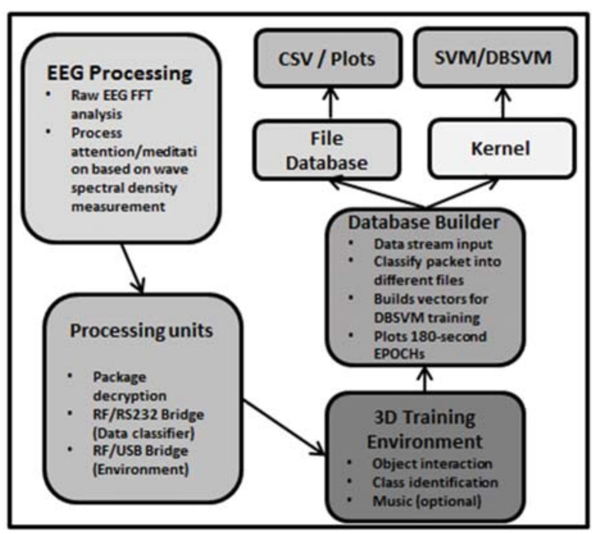


Figure 1: System description

2. Density Based Support Vector Machines

Machine learning techniques are generally helpful in feature extraction among EEG signals, since the recorded signals usually tend to have variable bias and uncertain changes, which make the feature classification difficult. The Support Vector Machine (SVM) model proposed by Dr. Vapnik has been used as a form of supervised learning models (SLM) in detecting certain features in the captured EEG data and in many other systems [6].

However, the sensitivity of SVMs to outliers is a weak point for this algorithm. One of the approaches for reducing this sensitivity is the Density Based Support Vector Machine (DBSVM) method, which is based in the simple expression of population density according to Equation (1) [7]:

$$Population\ density = \frac{number\ of\ population}{area} \qquad (1)$$

Although the concept of population density is used to develop the DBSVM model, the formula is actually based on two separate algorithms. The removal of outliers is based on comparing a vector's total Euclidean or Mahalanobis distance between the average distances of the mapped vectors [8].

2.1 Outliers

Outliers are unusual data points that are inconsistent with other observations. In statistics an outlier is an observation with an abnormal distance from most observations.

Generally presence of an outlier may cause some sort of problems. An outlier may be due to gross measurement error, coding/recording error, and abnormal cases, but a frequent cause of outliers is a mixture of two distributions and they can be occur by chance in any distribution[9].

2.2 DBSVM with Mahalanobis Distance

The Mahalanobis distance is the distance between the x from the quantity μ. This distance is based on the correlation between variables or the variance-covariance matrix S.

For instance, let's suppose a set of n numbers of two-dimensional data $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ is given. The Mahalanobis distance between two points is given by Equation (2):

$$D_m(x) = \sqrt{(x - y)^T S^{-1} (x - y)} \qquad (2)$$

The next step is to sum up all the distances for each point (i.e. $d_1 = [D(1,2) + D(1,3) + \cdots + D(1,n)]$) where n is the number of data points in the data set.

The total distance for all data points of one data set is needed to calculate the average distance that will be used to determine data points inside and outside of densely populated area. Equation (3) shows the expression to calculate this:

$$Average\_d = \frac{\sum_{i=1}^{N} \sqrt{(x_i - y_i)^T S^{-1} (x_i - y_i)}}{n} \qquad (3)$$

$$if\ d_i > Average\_d \rightarrow x_i = outlier$$

After calculating $Average\_d$ by formula, those points that are not considered as outliers will become the new training data set in which the points will be considered important data points.

2.3 DBSVM with Euclidean distance

The Euclidean distance between two points is given by Equation (4):

$$D(1,2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \qquad (4)$$

Therefore, as used for the Mahalanobis Distance average calculation, the average distance for excluding outliers is as described in Equation (5):

$$Average\_d = \frac{\sum_{j=1}^{N} \sum_{i=1}^{N} \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}}{n} \qquad (5)$$

The data sets obtained from the training environment will be tested with both population density methods for success rate comparison with hard margin SVMs. The application of this method can help for having a maximized margin and a better generalization of the EPOCH mapped values from the signal vector transformation.

3. The 3D training environment

The use of simulators for training with EEG Biofeedback is a common application and has been proven effectiveness in solving problems such as attention enhancement [10].

The proposed training environment was designed with the use of Blender for the animated models and structures and Unity3D for its ease of use and deployment. The environment has a series of features such as music and interaction with the experiment variables that are described as follows.

The environment setup was created for the test subjects to familiarize with the objects and understand the way of interacting with them; an additional relaxation music track can be used for aiming in obtaining higher meditation behaviors.

The purpose of the training task is to guide the player (training subject) throughout the predefined path, as specified in Figure 2.

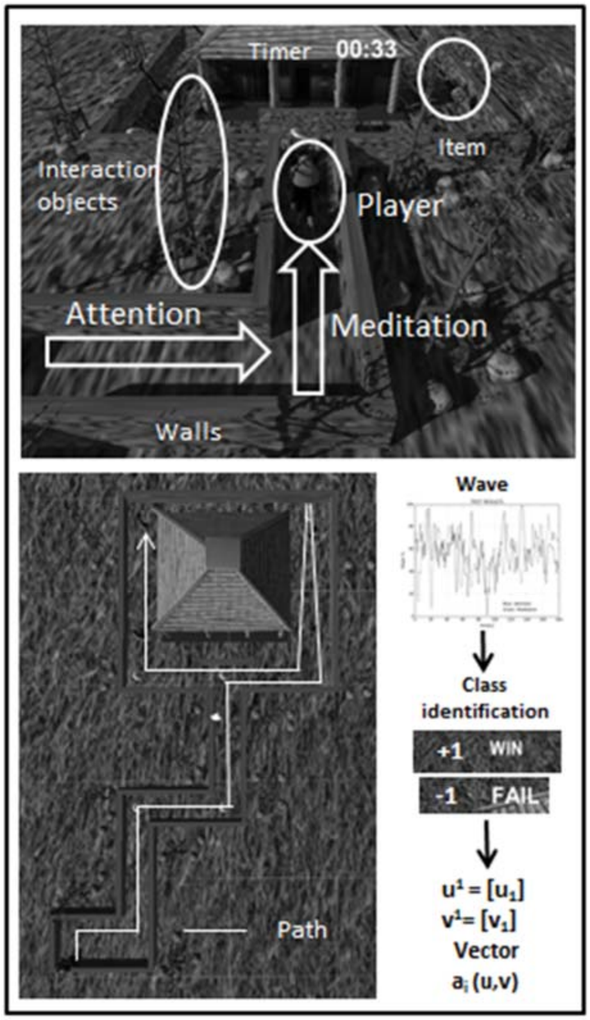


Figure 2: Environment description

When concentrated, the user can move the player forward and then move the player to the right if the user begins to enter meditation. In either case, the user can stop the player by blinking the eyes constantly until the player stops due to the artifact noise produced across the path, there are four main items that the subject requires to obtain.

If the series of mind-waves came into the expected pattern in the required time, the attempt will be classified as successful and the class *+1* will be labeled for the output EPOCH. However, if the test subject is not able to perform the entire track along the timeframe, an unsuccessful attempt is recorded with the *-1* label. The resulting vectors will be used as samples for later mapping in the SVM classifier.

Brain waves are dependent on each test subject, and so the performed wave patterns and headset calibration. Due to this factor, the training requires to be performed individually. Fortunately, the device has a fast self-calibration routine [11] and a trained subject can easily adapt in controlling the headset in minutes. In order to have a good classifier, 30 samples where obtained from the test subject. In the case it is complicated to generate the attention or meditation patterns, the use of the additional music tracks can stimulate the brain for generating the necessary beta or alpha waves for the device.

The testing area was a quiet room with the test subject sitting in front of a screen displaying the 3D environment and the data processing script running in the background for obtaining the corresponding data sets. Whenever the test subject was ready to perform a test a red button was pressed to restart the environment and script.

4. Results

Once the test was performed, as there are two outputs from each result, matrix *A* consisting of attention values and matrix *M* with meditation values. Feature *F* from Equation (6) was used to relate both matrices:

$$F = \left[\frac{A^2(A-M)}{A+M}\right] \tag{6}$$

Assuming the condition if $A_i = -M_i$, $F_i = 0$ for avoiding division by zero. Once the matrices have been associated with the feature, the *PCA* matrix *P* is obtained for matrix *F*. The two main components the transform matrix *B* generated by Equation (7) were used to map the outputs in the two-dimensional space.

$$FP = B \tag{7}$$

For example, the first component of the first row of the matrix will be the *x* coordinate and the second component will be the *y* coordinate. The resulting hard margin SVM classifiers are presented in Figure 3.

The kernels that were used are the linear, RBF and polynomial degree 3. The LinearSVC kernel uses a different code library that allows a more flexible modeling and better choice of penalties and loss function scaling [12].

Once the DBSVM has been set, there was a test vector consisting of five test elements F_t, with their respective class label. This time the mapping coordinates were obtained by using the PCA matrix P described by Equation (7).

The corresponding success rate chart is described by Table. The performed classification had a success rate of 80% success for the test vector on the LinearSVC implementation.

The classification success rate for the test data set could be augmented if a soft margin is implemented.

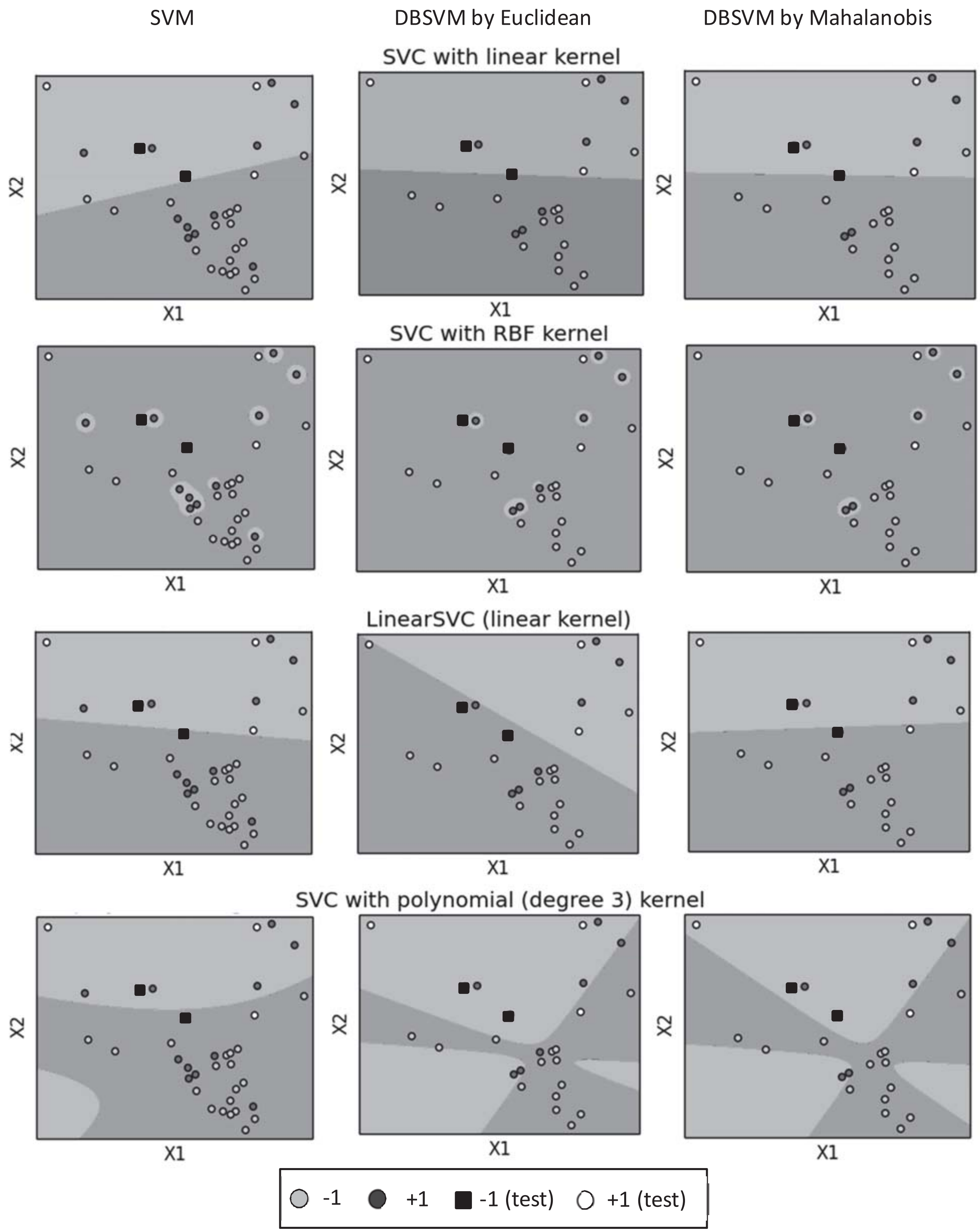


Figure 3: SVM and DBSVM classifiers

Table 2: SVM Classification success rates

Kernel	Normal	*Euclidean*	*Mahalanobis*
Linear	20%	60%	60%
RBF	40%	40%	40%
LinearSVC	80%	80%	60%
Polynomial	60%	20%	20%

5. Conclusions

This paper presents a way to use the Density Based Support Vector Machine Classifiers for a synchronous mind pattern classification problem. One of the main disadvantages the SVMs is the selection of a particular feature that can reduce the classification error. By choosing the correct feature, data sets can be condensed in specific regions and clusters in which the Euclidean and Mahalanobis outlier discrimination algorithm can perform as expected.

For this particular problem, the selected feature was the one that worked best after trying various models with the use of PCA. There can be other mapping approaches that can be performed based on the Attention/Meditation signal correlation. The use of a more flexible linear kernel for the training could obtain a higher learning rate without soft margin. If soft margin was included, the learning success would be 100%. However, the number of test vectors should be increased for a better test accuracy.

However, the use of the raw EEG data sets could represent a better way for classification derived from training.

The performed test demonstrates the effective use of low-cost, single electrode commercial FFT-based EEG headsets for simple classification tasks with the aid of a simple training environment with optional music feedback. Nonetheless, it is important to choose a specific feature to reduce the error.

6. Further work

From the DBSVM, a more specific feature will be determined based on the Attention/Meditation wave behaviors on the obtained EPOCHs. The resulting classifier will be used with different test vectors that will be based on performing in the training environment; by either moving the player or imaging the player's relative position and movement.

If successful, the end user will be performing an Attention/Meditation pattern recreation without the aid of the environment. This will be performed by the end used with the use of single and multiple channels FFT based EEG tools for performance analysis.

Also, the use of Independent Component Analysis for Raw EEG data will be implemented for different brain regions in order to generate a better feature based on specific active electrode positions.

References

[1] D. C. Jangraw, A. Johri, M. Gribetz and P. Sajda, "NEDE: An open-source scripting suite for developing experiments in 3D virtual environments", Journal of Neuroscience Methods, ELSEVIER, pp. 2-5, 2013.

[2] J. Kim and W. Lee, "EEG Signal feature analysis of Smartphone Game User", Advanced Science and Technology Letters, Science & Engineering Research Support Society (SERSC), pp. 2-5, 2013.

[3] NeuroSky Inc., "MindSet Communications Protocol", Step-By-Step Guide to Parsing a Packet, NeuroSky, Inc., pp. 6-8, June 2010.

[4] W.Salabun, "Processing and spectral analysis of the raw EEG signal from the MindWave", PRZEGLĄD ELEKTROTECHNICZNY, pp. 169,-172, February 2014.

[5] O. Cho and W. Lee, "BCI Sensor Based Environment Changing System for Immersion of 3D Game", International Journal of Distributed Sensor Networks, Hindawi Publishing Corporation, pp. 3-4, May 2014.

[6] J. Light, Kalaiselvi T., X. Li and A. R. Malali, "Fall Pattern Classification from Barin Signals using Machine Learning Models", Journal of Selected Areas in Telecommunications (JSAT), Cyber Journals: Multidisciplinary Journals in Science and Technology, pp. 1-3,5, December 2013.

[7] National Geographic, http://education.nationalgeographic.com/education/activity/calculating-population-density/?ar_a=1

[8] Z. Nazari, D. Kang, and H. Endo, "Density Based Support Vector Machines", The 29th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), July 2014.

[9] D. Ripley, Robust Statistics, M.Sc. in Applied Statistics, MT2004.

[10] B.H. Cho, J.M. Lee, J.H. Ku, D.P. Jang, J.S. Kim, I.Y. Kim, J.H. Lee and S.I. Kim, "Attention Enhancement System using Virtual Reality and EEG Biofeedback", Proceedings of the IEEE Virtual Reality 2002 (VR'02),IEEE, pp. 4-7, 2002.

[11] NeuroSky Inc., "MindWave User Guide", NeuroSky, Inc., pp. 18, July 2011

[12]http://scikitlearn.org/stable/modules/generated/sklearn.svm.LinearSVC.html, 2010-2014

Automatic Recognition of Speech Patterns of Numeric Digits Using Support Vector Machines: A New Approach

G. Batista[1], **and W. Silva**[2]
[1]Student of Electrical Engineering, Federal Institute of Maranhao, Sao Luis, Maranhao, Brazil
[2]Department of Electrical and Electronics, Federal Institute of Maranhao, Sao Luis, Maranhao, Brazil

Abstract— *This paper proposes the implementation of a Support Vector Machine (SVM) for automatic recognition of numerical speech commands. Besides the pre-processing of the speech signal with mel-ceptral coefficients, is used to Discrete Cosine Transform (DCT) to generate a two-dimensional matrix used as input to SVM algorithm for generating the pattern of words to be recognized. The Support Vector Machines represent a new approach to pattern classification. SVM is used to recognize speech patterns from the mean and variance of the speech signal input through the two-dimensional array aforementioned, the algorithm trains and tests those data showing the best response. Finally it shows the experimental results in speech recognition applied to Brazilian Portuguese language process.*

Keywords: Support Vector Machines; Classification; Pattern Recognition; Statistical Learning Theory; Application in Speech Recognition.

1. Introduction

1.1 Digital Processing of the Speech Signal

Digital speech processing is a specialty in full expansion. There are numerous applications of this research area, we can refer to automatic speech recognition for purposes of interpretation of commands by machines or robots, automatic speech recognition for the purpose of biometric authentication, recognition of pathology in the mechanism of speech production for biometric and or medicinal purposes. The speech processing systems are divided basically into three sub-areas: speech coding, speech synthesis and speech automatic recognition. Regardless of the specific purpose, the initial stages of a system for processing digital speech is sampling followed by segmentation of words or phonemes [1] for short-term analysis by Fourier transform or by spectral analysis. The speech signal processing first involves obtaining a parametric representation based on a certain model and then applying a transformation to represent the signal in a more convenient form for recognition. The last step in the process is the extraction of important characteristics for a given application. This step can be performed either by human listeners or automatically by machines [14]. Among the techniques that have been developed for segmentation of speech, those based on Hidden Markov Models (HMM) are quite traditional. Hybrid methods based on artificial neural networks and criteria such as average energy, selection of voiced phonemes and non voiced, Mel Frequency Cepstral Coefficients (MFCCs), spectral metrics, and others, are also used. Speech coding systems include those cases in which the purpose is to obtain a parametric representation of the speech signal, based on the analysis of the frequency, average power and other characteristics of the spectrum of the signals. The techniques of encoding the speech signal are used both for transmission and for compact storage of speech signals. One of the main applications of speech coding is to transmit the speech signal efficiently [2]. Systems for automatic speech recognition or Speech Recognition Systems (SRS) are focused on the recognition of the human voice by intelligent machines.

1.2 Methodology Proposed

This article uses as a recognition default locutions from Brazillian Portuguese of the digits $'0','1','2','3','4','5','6','7','8','9'$. The speech signal is sampled and encoded in mel-cepstral coefficients and coefficients of Discrete Cosine Transform (DCT) in order to parameterize the signal with a reduced number of parameters. Then, it generates two dimensional matrices referring to the mean and variance of each digit. The elements of these matrices representing two-dimensional temporal patterns will be used to classify by machines (Support Vector Machine).

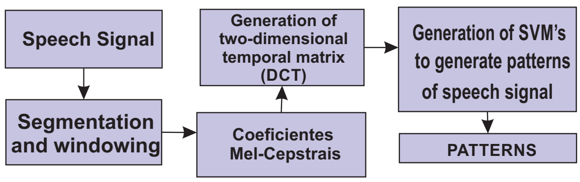


Fig. 1: Flowchart Blocks of Training System.

1.3 Pre-processing of Speech Signal

Initially, the speech signal is sample and segmented into frames, after segmentation of the speech signal passing through a process of windowing and it is encoded in a set of mel-cepstral parameters. The number of parameters obtained is determined by the order of mel-cepstral coefficients. The obtained coefficients are then encoded by Discrete Cosine Transform (DCT) in a two dimensional matrix that will represent the speech signal that will be recognized. The process of windowing in a given signal, aims to select a small portion of this signal, which will be analysed, named frame. A short-term Fourier analysis performed on these frames is called signal analysis frame by frame. The length of the frame T_f is defined as the length of time upon which a parameter set is valid. The term frame is used to determine the length of time between successive calculations of parameters. For speech processing, normally, the time frame is between 10ms and 30ms [13].

1.4 Generation of two-dimensional DCT-temporal matrix

After being properly parameterized in mel-cepstral coefficients, the signal is encoded by DCT performed in a sequence of observation vectors of mel-cepstral coefficients on the time axis. Thus, a two-dimensional temporal array DCT is generated for each m (m=1,2,3,...,10 number of samples to generate each pattern) example of model ***P***, represented by C_{kn}^{jm}. Finally, arrays of mean CM_{kn}^{j} (1) e variance CV_{kn}^{j} (2) are generated. The parameters of CM_{kn}^{j} and CV_{kn}^{j} are used as datas of input in SVM algorithm.

$$CM_{kn}^{j} = \frac{1}{M}\sum_{m=0}^{M-1} C_{kn}^{jm} \quad (1)$$

$$CV_{kn}^{j} = \frac{1}{M-1}\sum_{m=0}^{M-1}\left[C_{kn}^{jm} - \left(\frac{1}{M}\sum_{m=0}^{M-1} C_{kn}^{jm}\right)\right]^2 \quad (2)$$

where $k, 1 \leq k \leq K$, refers to the $k-$th line (number of Mel frequency cepstral coefficients) of $t-$th segment of the matrix n, $1 \leq n \leq N$ component refers to the $n-$th column (order of DCT) and j=0,1,2,...,9 is the number of patterns to be recognized.

1.5 Generation of machines

In the technical literature about SVMs, the standards are called classes. The mean and variance matrices are transformed in two column vectors, CMe (vector with means) and $CVar$ (vector with variances).

$$\begin{aligned} CMe_i^j \;=\; & \langle CM_{11}^0, CM_{12}^0, ..., CM_{1N}^0, CM_{21}^0, CM_{22}^0, ..., CM_{2N}^0, \\ & CM_{KN}^J \rangle \end{aligned} \quad (3)$$

$$\begin{aligned} CVar_i^j \;=\; & \langle CV_{11}^0, CV_{12}^0, ..., CV_{1N}^0, CV_{21}^0, CV_{22}^0, ..., CV_{2N}^0, \\ & CV_{KN}^J \rangle \end{aligned} \quad (4)$$

For example, in the case of a matrix CM_{22}^j, that is, where K=2 e N=2, the matrices CMe and $CVar$ take the following form:

$$\begin{aligned} CMe_i^j \;=\; & \langle CM_{11}^0, CM_{12}^0, CM_{21}^0, CM_{22}^0, CM_{11}^1, \\ & CM_{12}^1, CM_{21}^1, CM_{22}^1, ..., CM_{22}^J \rangle \end{aligned} \quad (5)$$

$$\begin{aligned} CVar_i^j \;=\; & \langle CV_{11}^0, CV_{12}^0, CV_{21}^0, CV_{22}^0, CV_{11}^1, CV_{12}^1, \\ & CV_{21}^1, CV_{22}^1, ..., CV_{22}^J \rangle \end{aligned} \quad (6)$$

Each class in this example is represented by 4 elements in the vector of mean and 4 elements in vector of variance according to (5) and (6), that is, the first 4 elements of the vector of mean and of the vector of variance refer into class 0, the following 4 elements of each vector to the class 1, and so on. Figure 2 shows data of the peers of mean and variance of the speech signals from the examples of (5) and (6). The set of functions mapping of type input-output is

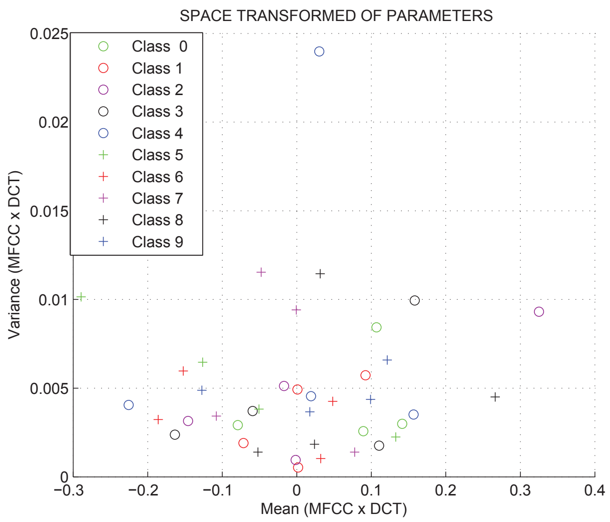


Fig. 2: Classes and their different points.

given by Equation 7:

$$\Omega = f\Big([CMe_i^j; CVar_i^j], w\Big) \quad (7)$$

where Ω is the real response produced by the learning machine associated with the entry of pairs of means and variances, and w is a set of free parameters, called weights for weighting, selected from the parameter space related to patterns. Figure 3 shows a general model of the supervised learning from the examples, having three components:

The **Environment** is the fixed input system, this yields x_i (points that come from the pairs of coordinates $(CMe, CVar)$) from the response of the DCT matrix of speech signals. The **Supervisor** returns a value of the desired output d_i for each input vector x_i in accordance with a conditional distribution function $F\,(d_i|x_i)$), also set. **Machine of Learning (ML)**, or algorithm capable of implementing a set of functions $f\Big([CMe_i^j; CVar_i^j], w\Big)$, where $\omega \in W$, where W is a set of parameters belonging to the set of desired

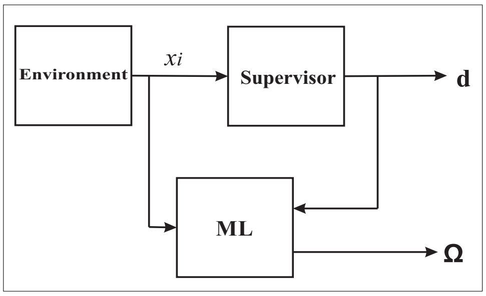


Fig. 3: Model of Learning.

responses. In this context, the learning problem can be interpreted as a **problem of approximation**, which involves finding a function $f\left([CMe_i^j; CVar_i^j], w\right)$ that generates the best approximation to the Ω output of the supervisor. The selection is based on a set of independent training examples I and identically distributed (iid), generated according to:

$$F(x, d) = F(x)F(d|x) : (x_i, d_i) \tag{8}$$

where (x_i, d_i) are peers with desired input and output with $d_i \in R^n$ and $i = 1, ..., I$.

1.5.1 SVM (Support Vector Machine)

Based on Statistical Learning Theory, a Support Vector Machine was developed by Vapnik [3], in order to solve problems of pattern classification, from studies initiated in the work "On the uniform convergence of relative frequencies of their probabilities to events" [5]. The Theory of Statistical Learning aims to establish mathematical conditions that allow the selection of a classifier with good performance for the data set available for training and testing. In other words this theory seeks to find a good classifier with good generalization regarding the entire data set. The desired performance of a classifier f is that it gets the smallest mistake during training, with the error being measured by the number of incorrect predictions of f. Therefore it's defined as Empirical Risk $Remp(f)$ the extent of loss between the desired response and the actual response and restrictions on Risk Functional use the concept of VC dimension [5]. Theory of uniform convergence of functional of empirical risk to functional of real risk includes limits on the rate of convergence, which are based on an important parameter called the Vapnick-Chervonenkis dimension, or simply VC dimension, named after its creators, Vapnick and Chervonenkis. The VC dimension is a measure of the capacity or power of expression of the family of classification functions performed by the learning machine [9]. Haykin [6] furthers more details about the Functional of Risk and VC dimension.

SVM is a classifier that separate linearly the data through a hyperplane. And to determine the optimal hyperplane separability, as it was assumed that the training set is linearly separable. The separating hyperplane the follows equation of a decision surface below:

$$\omega^T x + b = 0 \tag{9}$$

where x is an input vector, ω is a vector of adjustable weight (maximum separation possible between true and false examples) and b is a *bias* [16].

For the case of a non-linear set, the SVM's create another feature space from the original space, and the concepts and calculations of linear optimal hyperplane are applied in this new space [6].

The SVM is a dichotomic algorithm, that is, for pattern classfication based on two classes [6]. However, it is possible to obtain a classifier for multiple classes using the SVM algorithm. Scholkopf et al. the proposed classifier model of type "one vs. all" [10]. Clarkson and Brown have proposed classifier model of the "one vs. one" [4]. However, both models are indeed classifiers of only two classes: Class +1 and Class -1 [6]. On system "one vs. all" it is used one machine for each group, in which each group is trained separately from the rest of the set. In the system "one vs. one" it is used only three machines, in which a group is classified against another, then this one is rated against another group and so on, until the whole set is trained.

The decision surface of the SVM, which in the feature space is always linear, usually is nonlinear in the input space. As seen earlier, the idea of Support Vector Machine depends on two mathematical operations:

1) Nonlinear mapping of an input vector into a feature space of high dimensionality, which is hidden from the entry and exit [16];
2) It's necessary to build an optimal hyperplane to separate the features discovered in the first step. To design the optimal hyperplane it is needed a kernel function, or core of the inner product. A Kernel function is a funcion that receives two points of the input space and calculates the scalar product of the data in the feature space [16].

To ensure the convexity of the optimization problem and introduce the Kernel mapping in which the calculation of scalar products is possible, you must use a kernel function that follows the conditions set by Mercer's Theorem [11], [12]. In general, the three most important of the kernel functions are Polynomial, RBF kernel and Perceptron (MLP) [6].

2. Experimental Results

After performing the pre-processing of the speech signal coding and generation of temporal matrices CM_{kn}^j and CV_{kn}^j, the models were trained by SVM machines CM_{22}^j and CV_{22}^j, that is, *K*=2 and *N*=2 as shown in Figure 4, for CM_{33}^j and CV_{33}^j, that is, *K*=3 and *N*=3 as shown in Figure 5 , and CM_{44}^j e CV_{44}^j, ie , *K*=4 e *N*=4 as shown in Fig 6.

With the result of the best machines from training, the tests were made from voice banks where the speakers are independent and classified with the best machines of training. The speakers 1 and 2 are male and the speaker 3 is female. The Tables 1, 2 and 3 show the rates of successes. The Figures 4, 5 and 6 are from the best results that were generated by *RBF* function of sigma 0.03. Besides the best results, good results also were obtained from *Polynomial* of order 2, as show the tables 4, 5 and 6. To improve the tests results, training was made from 20 examples of each pattern.

The generated hyperplane during classification with *RBF* funtion with sigma 0.03 is very small. This is because as smaller the sigma, smaller the coverage area of the hyperplane is. This explain why it is impossible to observe the data that were classified as true in its classes in the Figures 4, 5 and 6.

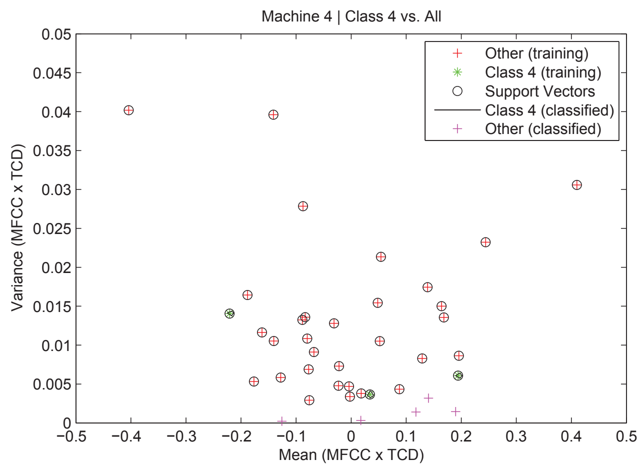


Fig. 4: Machine generated for class 4 from matrices CM^4_{22} and CV^4_{22}.

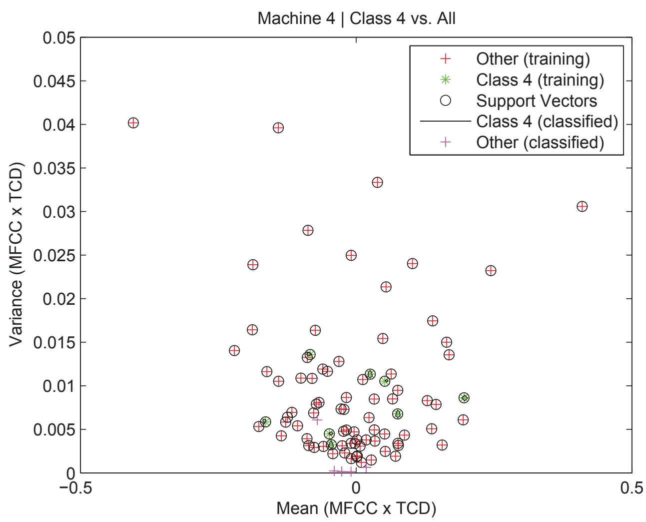


Fig. 5: Machine generated for class 4 from matrices CM^4_{33} and CV^4_{33}.

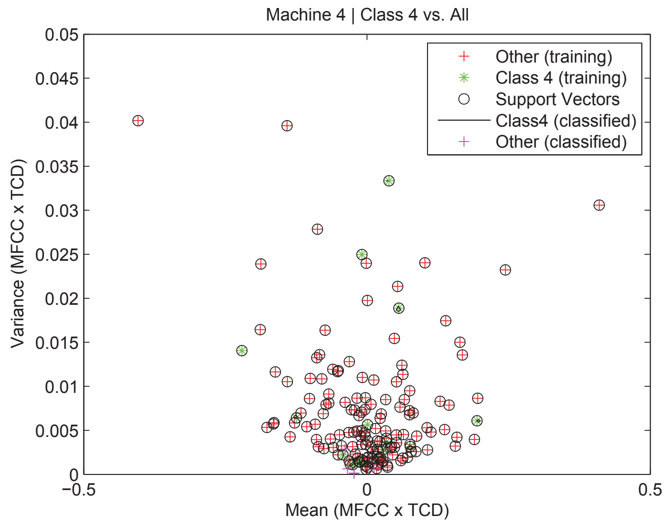


Fig. 6: Machine generated for class 4 from matrices CM^4_{44} and CV^4_{44}.

Table 1: Test performed from matrices CM^j_{22} and CV^j_{22} and *RBF* of sigma 0.03

Machines	Training	Test		
		Speaker 1	Speaker 2	Speaker 3
Class 0	16	8	8	8
Class 1	16	8	8	8
Class 2	16	8	8	8
Class 3	16	6	10	6
Class 4	16	8	6	6
Class 5	16	8	6	8
Class 6	16	8	8	8
Class 7	16	6	8	6
Class 8	16	8	8	8
Class 9	16	8	8	8
TOTAL	160	76	70	74

Table 2: Test performed from matrices CM^j_{33} and CV^j_{33} and *RBF* of sigma 0.03

Machines	Training	Test		
		Speaker 1	Speaker 2	Speaker 3
Class 0	16	8	8	6
Class 1	16	8	8	8
Class 2	16	8	8	6
Class 3	16	8	8	8
Class 4	16	8	10	8
Class 5	16	8	6	8
Class 6	20	8	6	8
Class 7	16	8	8	8
Class 8	16	8	8	8
Class 9	16	8	8	8
TOTAL	164	80	78	76

Table 3: Test performed from matrices CM_{44}^{j} and CV_{44}^{j} and *RBF* of sigma 0.03

Machines	Training	Test		
		Speaker 1	Speaker 2	Speaker 3
Class 0	20	8	8	8
Class 1	20	8	8	8
Class 2	16	8	8	8
Class 3	20	10	8	6
Class 4	20	8	8	8
Class 5	16	8	8	8
Class 6	20	8	8	8
Class 7	16	8	8	8
Class 8	16	8	8	8
Class 9	16	8	8	8
TOTAL	180	82	80	78

Table 4: Test performed from matrices CM_{22}^{j} and CV_{22}^{j} and *Polynomial* of order 2

Machines	Training	Test		
		Speaker 1	Speaker 2	Speaker 3
Class 0	20	1	4	4
Class 1	20	6	4	0
Class 2	20	8	8	8
Class 3	16	8	8	8
Class 4	16	6	1	1
Class 5	16	6	8	8
Class 6	16	6	6	6
Class 7	16	10	1	4
Class 8	20	6	8	8
Class 9	20	8	8	8
TOTAL	180	54	56	55

Table 5: Test performed from matrices CM_{33}^{j} and CV_{33}^{j} and *Polynomial* of order 2

Machines	Training	Test		
		Speaker 1	Speaker 2	Speaker 3
Class 0	20	8	6	6
Class 1	20	6	6	4
Class 2	20	4	8	4
Class 3	20	4	4	6
Class 4	16	6	6	8
Class 5	20	6	8	10
Class 6	16	6	8	8
Class 7	16	6	4	6
Class 8	20	6	4	8
Class 9	20	8	8	6
TOTAL	188	60	62	66

Table 6: Test performed from matrices CM_{44}^{j} and CV_{44}^{j} and *Polynomial* of order 2

Machines	Training	Test		
		Speaker 1	Speaker 2	Speaker 3
Class 0	16	8	6	4
Class 1	20	6	4	4
Class 2	20	10	8	8
Class 3	16	6	6	10
Class 4	20	10	8	8
Class 5	20	6	8	8
Class 6	16	1	6	4
Class 7	20	6	8	8
Class 8	20	10	10	4
Class 9	20	6	6	6
TOTAL	184	69	70	68

3. Conclusions

Analysing the methodology and applications of SVM, one realises that it is a technique with excellent response time of computational execution. Despite being a dichotomic method of classification, this also has possible means to work with a larger number of classes of different data types to be separated. In the standards classification proposed in this work, the SVM presented problems to correctly classify points very close among each other, because of the form of generalization one versus all. However, as it has a very wide scope in relation to the classification functions during learning process of the machines, the SVM ends up compensating the problem of generalization with the use of more points for classification. That is, the greater the number of points to represent the class the higher the amount of hits. Regarding the recognition of patterns in general were well classified, except with the digit '9'. The digits '1' and '8' obtained the highest classifications. And between *Polynomial* and *RBF* functions, the second one presented the best results. The use of mean and variance chosen as characteristics of the data to be generated patterns was the most appropriate way to find a better separability between points and therefore a better classification.

References

[1] P. C. Fantinato, *Segmentacao de Voz baseada na Analise Fractal e na Transformada Wavelet*, USP, 2008.

[2] A. A. Bresolin, *Reconhecimento de voz atraves de unidades menores do que a palavra, utilizando Wavelet Packet e SVM, em uma nova Estrutura Hierarquica de Decisao*, UFRN,2008.

[3] V. N. Vapnik, *The nature of statical learning theory.*, Spring-Verlag, 2ed, 2000.

[4] P. Clarkson and P. J. Moreno, *Acoustics, Speech and Signal Processing.*, IEE Internacional Conference, Vol 2, 1999.

[5] V. N. Vapnik and A. Y. Chervonenkis, *On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities*, Dokl, 2ed, 1968.

[6] S. Haykin, *Redes Neurais:Principio e pratica*, Bookman, 2002.

[7] C. H. Ding and I. Dubchak, *Multi-class protein fold recognition using support vector machines and neural networks*, Bioinformatics, 4ed, 2001.

[8] R. S. Urena et al., *Real-time robust automatic speech recognition using compact support vector machine*, IEEE Transaction on audio, speech, and language processing, 4ed, May, 2012.

[9] C. J. C. Burges, *A tutorial on Support Vector Machines for Pattern Recognition.*, Kluwer Academic Publishers, Boston, 1998.

[10] B. Scholkopf, O. Simard, A. Smola and V. Vapnik, *Prior knowledge in support vector kernels.*, The MIT Press, Vol 2, 1999.
[11] J. Mercer, *Functions of positive and negative type, and their connections with theory of integral equations*, Proceedings of the Royal Society of London - Philosophical Transactions of the Royal Society, Royal Society of London, November 3, 1909.
[12] C. De-Gang, Y. W. Heng and E. C. C. Tsang, *Generalized Mercer theorem and its application to feature space related to indefinite kernels*, IEEE Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, Kunming, 12-15 July 2008, 2008.
[13] J. W. Picone, *Signal Modeling Techniques in Speech Recognition*, IEEE Transactions on Computer, Vol 81, 9th edition, Apr. 1993, pp. 1215-1247, doi: 10.1109/5.237532.
[14] L. Rabiner and J. Biing-Hwang, *Fundamentals of Speech Recognition*, Prentice Hall, New Jersey, 1993.
[15] P. L. L. Fissore and E. Rivera, *Using word temporal structure in HMM Speech recongnition*, ICASSP 97, Vol 02, April, 1997.
[16] G. C. Batista, W. L. S. Silva and O. D. R. Filho, *Classification of Pattern using Support Vector Machines: An Application for Automatic Speech Recognition*, ADVCOMP, 2014.

Case Base Size and Overall Competence: Incremental Increase and Similarity Threshold Selection on a Data Set

Spencer Knight and John Hastings
Dept. of Computer Science & Information Technology, University of Nebraska-Kearney, Kearney, NE, USA

Abstract— *This paper builds on prior case-based reasoning (CBR) research into the effect of incremental increase of case library size on overall system competence. We explore the effect of gradually increasing the size of the case base on the total accuracy of a CBR algorithm, using a large numeric data set as an example. We use a standard mathematical definition of system competence in our study of the data set and compare these results to those obtained in prior research on several other sets.*

We also use a strategy for adjusting the size of the similarity threshold, or maximum distance between a test case and a training point for which the match between the two can be considered successful. This makes use of both data normalization and consideration of maximum possible error. As expected, increasing the size of the case library had the effect of increasing competence and accuracy, but this technique did not always produce the most intuitive results.

Keywords: case-based reasoning, machine learning, system competence, knowledge base size

1. Introduction

Machine learning techniques are ubiquitous. The rise of smartphones and other devices capable of adjusting their behavior to suit the user's habits, preferences, and styles without being explicitly informed means that the reader may have no further to look to find applications of machine learning techniques than his or her own pocket.

Much research has been done in the area of machine learning over the last six decades. Case-based reasoning (CBR) [1][2] is a field of research within machine learning in which systems rely on a knowledge base of cases (i.e., past or prototypical scenarios) that are captured either manually or automatically. This paper builds upon prior CBR research which utilized a basic but popular and mature approach, the nearest neighbor technique [3]. Using the definition of competence referenced by Leake and Wilson [4], we also use the nearest neighbor approach.

The nearest neighbor technique was applied to a large data set repeatedly and exhaustively to make a determination of the CBR system's competence as the size of the case base is increased.

2. Competence of a CBR System

For the purposes of this research, the mathematical definition of the competence of a CBR system is the same as laid out in Leake and Wilson [4] which defines competence as the proportion of "correct" matches to the number of cases in the current case base. More specifically, for each case in the current case library, the case is matched, adapted, and subjected to a similarity threshold to determine correctness. We use this approach to competence as well but omit the adaptation stage by assuming that a matched case which is sufficiently similar can be successfully adapted.

The competence of a CBR system is a measure of its accuracy, i.e., through an exhaustive examination of the data set, how well can we expect the algorithm to perform based on the given data. For this paper, we use complete leave-one-out testing or cross validation [5] with each increase of the size of the case library. That is, each time the case library size is increased, the algorithm runs through each case and measures the algorithm's competence at matching that test case using only the cases currently available to it.

This approach to competence has an obvious limitation, however. Since we are using only the data contained within the data set, our competence calculations can only truthfully be expected to apply to these data; new cases could potentially lie far outside the realm of the data we have. Sampling bias during the data collection phase, for example, may have reduced the applicability of the resulting CBR system to the target problem. Assuming that the case base is not biased and that it sufficiently covers the problem space is to make the representativeness assumption, i.e., that the case base is a representative sample of the target problem ([6]).

3. Competence and Case Library Size

Intuitively, it is easy to see that with a larger case library the competence and accuracy of a CBR system should be expected to increase (assuming no conflicting experiences). This is analogous to a human having more experience or more examples on which to draw experience.

However, the number of cases needed for a CBR system to reach a desired level of competence is highly dependent on the problem domain and the selection of attributes in the data set. A particular problem space, for example, may not be linearly separable and may therefore not be well-suited

Algorithm 1 Calculate competences for case base sizes 1 through max constructed from data set D

```
function CalculateCompetences(D,max)
    comp[ ] = float[max]                          ▷ Define float array comp of size max
    for i = 1 to max do
        S = select i cases at random from D       ▷ Construct a test case base of size i
        comp_i = LeaveOneOutTest(S)               ▷ Calculate competence for case base size i
    end for
    return comp
end function
```

to certain machine learning techniques. Another technique may be more appropriate in that instance.

The size of the case library is therefore very important, but acquiring new data in many situations may be time-consuming or expensive. An interesting topic in case-based reasoning is the determination of the necessary case library size long before case acquisition has been completed. This would hopefully limit wasted resources.

4. Similarity Thresholds

Another interesting question in this field is the notion of a threshold of similarity. For binary matching (determining whether a match returned by a CBR system is a success or a failure, with no other possibilities) performed on non-Boolean data such as discretized or numeric attributes in a data set, a threshold must be set to distinguish success from failure. The experimental algorithms described in the next section were run with both 5% and 10% matching error thresholds. The thresholds used here are percentages of the maximum possible match distance.

Because the details of the data set's domain were not known (namely, the maximum possible values for each of the white wine samples' attributes), a method was needed to determine the maximum possible matching error. For this reason, the data in the data set were normalized on a scale from zero to one, inclusive. The maximum error was then easy to calculate by considering a test case in which each attribute's value was one and a matched training case in which each attribute's value was zero. The normalization is itself a separate algorithm; adding or deleting cases to/from the data set will cause the program to re-normalize it.

5. Experimental Design

5.1 The Data Set

The data set used in this research was obtained from the University of California-Irvine's Machine Learning Repository [7]. It contains 4898 instances pertaining to various attributes of Portuguese "Vinho Verde" white wine ([8]). Each instance includes twelve numeric attributes of varying ranges and data types. An example instance from this data set is shown in Table 1.

Table 1: Sample Data Instance.

Feature	Value
1 - fixed acidity	7.8
2 - volatile acidity	0.76
3 - citric acid	0.04
4 - residual sugar	2.3
5 - chlorides	0.092
6 - free sulfur dioxide	15
7 - total sulfur dioxide	54
8 - density	0.997
9 - pH	3.26
10 - sulphates	0.65
11 - alcohol	9.8
12 - quality	5

To simplify calculations, the data were normalized, each on a range from zero to one. This made the maximum possible matching error easy to determine.

5.2 Algorithms

The algorithms used in this research are described in Algorithms 1 and 2. The main algorithm, displayed in Algorithm 1, calculates the competences for the case base sizes 1 through max constructed from the data set (or library) D. The program begins with a case library S consisting of one instance of the data set. It then increases the size of case library S by one with each iteration. In each iteration, the algorithm performs a nearest-neighbor, leave-one-out matching, using each case in case library S, as described in Algorithm 2. If the match returned is within the similarity threshold (explained above), it is considered a success. Otherwise, it is marked as a failure. The competence is then the proportion of successes to total matching attempts for the current case library size.

6. Results

The output of the algorithm described above was similar to the results obtained by Leake and Wilson. As expected, the overall system competence as determined with each increment of the case library size increased rapidly while the size of the case library remained small. Eventually, the benefit of adding more cases became marginal as the system's competence approached its maximum.

Algorithm 2 Calculate the average competence for case base S over leave one out testing for each case in S

```
function LeaveOneOutTest(S)
    totalCompetence = 0          ▷ Count the number of individual cases in S which are close to other cases in S
    for (each case C_i ∈ S) do                                   ▷ Use C_i as a test case
        T = S − C_i                                              ▷ Construct a test set T by removing C_i from S
        distance = NearestNeighborDist(C_i, T)                   ▷ Calculate the distance from C_i to the closest case in T
        if (distance < threshold) then                           ▷ If sufficiently close, then...
            totalCompetence = totalCompetence + 1                ▷ consider C_i already handled by the cases in T
        end if
    end for
    avgCompetence = totalCompetence/|S|
    return avgCompetence
end function
```

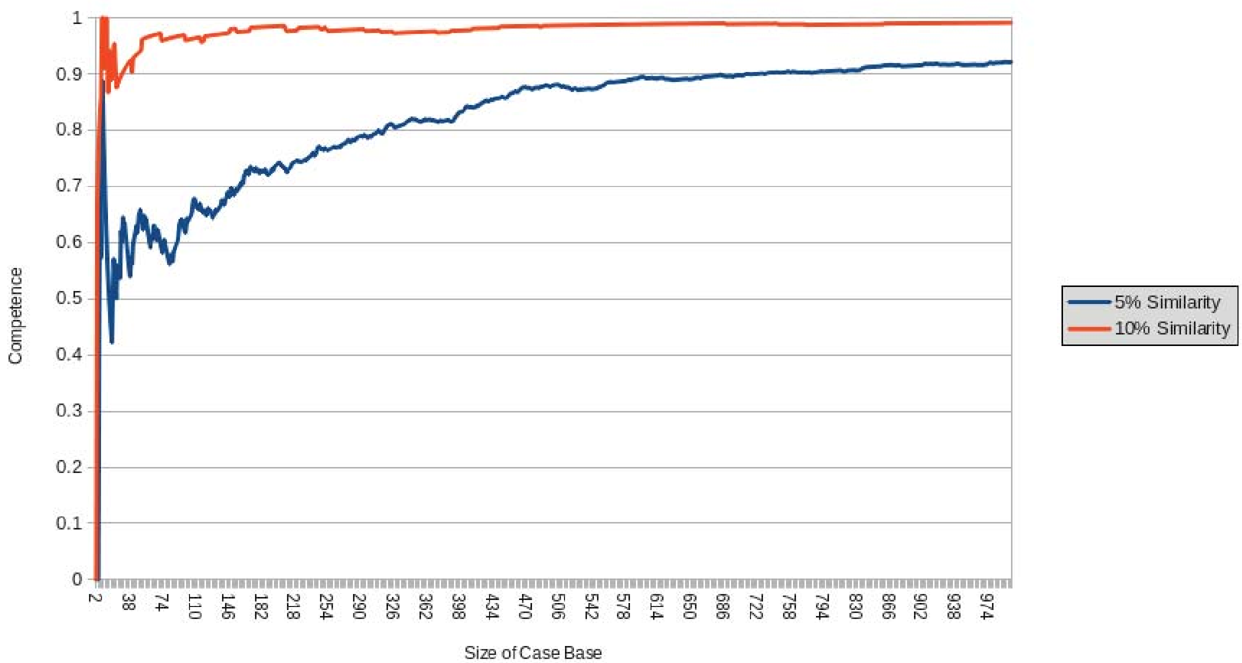


Fig. 1: Competence at Two Similarity Thresholds (Raw Data).

Figure 1 shows the competence of the system at each case library size for the two similarity thresholds without an associated trend line. The reader will notice anomalous outliers in this graph; the system occasionally reaches 100% accuracy with a very small case library size. This is, of course, not consistent. Figure 2 includes a logarithmic fit for the results for the two similarity thresholds.

The maximum competences obtained by the system at each similarity threshold are shown in Table 2. It should be noted that these are not asymptotic maximums; these are the competences achieved when all but one of the cases in the case library are used. Future research, also building on the work of Leake and Wilson, may be effective in predicting maximal competence a priori.

Note that the higher difference allowed for a successful match (with a 10% versus 5% threshold) increases the likelihood of "successful" matches by matching cases which are more distant. This results in a higher perceived system competence, which may or may not actually be the case depending on the ability of the adaptation component. In practice, allowing a higher matching threshold would likely require increased adaptation effort to account for the larger differences between cases. Going to the extreme, a 100% threshold (or difference) would result in a perceived competence (or matching success) of 100% starting as early as a case base size of one, but permitting such distant matches would need to be offset with a very effective adaptation component. Note that the adaptation component of a CBR

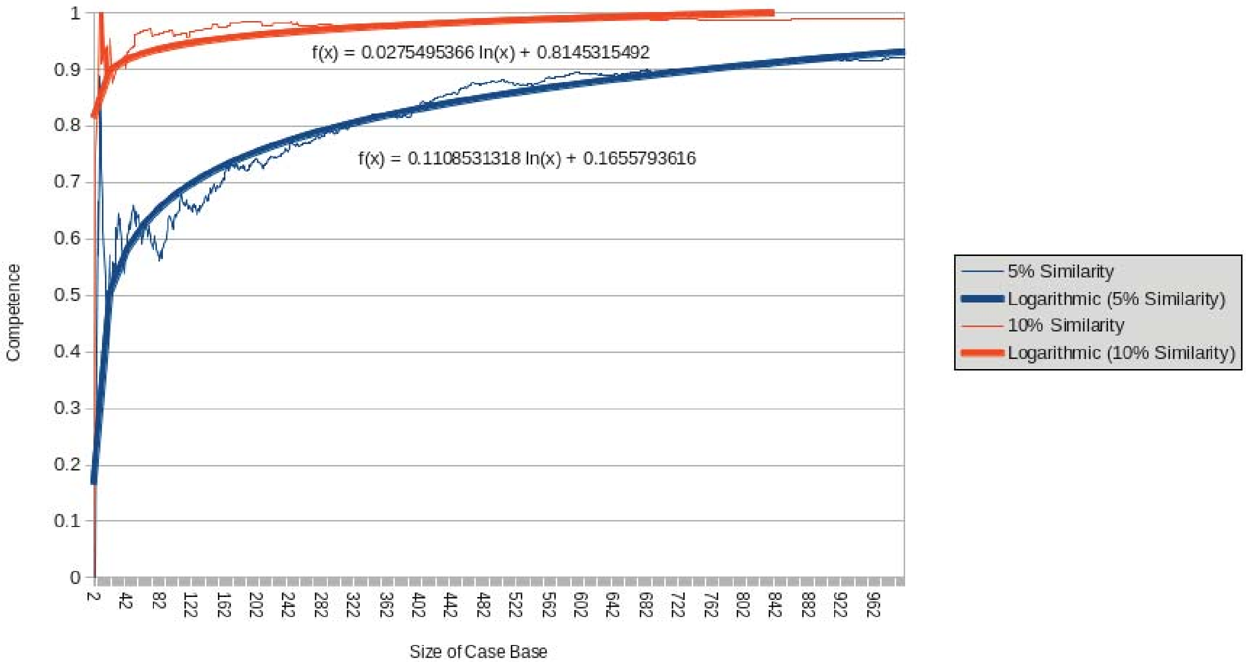


Fig. 2: Competence at Two Similarity Thresholds (Logarithmic Fit).

system is typically the most difficult portion to construct.

Table 2: Maximum Observed Competences.

Threshold	Maximum competence
5% of maximum error	0.894264
10% of maximum error	0.994897

These results confirm some of the findings of Leake and Wilson, for an additional data set. We obtained similar nearest-neighbor algorithm behavior: a rapid increase in accuracy followed by leveling and approach to a maximum competence. As the authors point out, findings such as these can be used to predict the size of the case base needed to achieve satisfactory competence.

7. Conclusion

These results show that the basic technique of nearest-neighbor matching can be very effective on a numeric data set, but that the choice of the similarity threshold greatly influences the perceived competence of a CBR system. The results from this algorithm also show that a similarity threshold choice based on a percentage of maximum possible error can be used to judge successful matches. Future research may expand on similarity thresholds and could yield a method to determine a threshold beforehand to give a particular competence.

We did not perform the adaptation phase during competence determination. This will be an interesting topic for future research. Various adaptation strategies, such as a weighting system via linear regression, could potentially be used to boost the accuracy of a CBR system, given a particular similarity threshold.

8. Acknowledgments

This work was supported in part by the University of Nebraska-Kearney Summer Student Research Program (SSRP).

References

[1] J. Kolodner, *Case-based reasoning*. San Mateo, CA, USA: Morgan Kaufmann, 1993.

[2] A. Aamodt and E. Plaza, "Case-based reasoning: Foundational issues, methodological variations, and system approaches," *AI Communications*, vol. 7, no. 1, pp. 39–59, 1994.

[3] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine Learning*, vol. 6, no. 1, pp. 37–66, 1991.

[4] D. Leake and M. Wilson, "How many cases do you need? Assessing and predicting case-base coverage," in *Proceedings of the Nineteenth International Conference on Case-Based Reasoning (ICCBR-11) Lecture Notes in Computer Science 6880*, London, England, September 2011, pp. 92–106.

[5] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society*, vol. 36, no. 2, pp. 111–147, 1974.

[6] B. Smyth and E. McKenna, "Building compact competent case-bases," in *Proceedings of the Third International Conference on Case-Based Reasoning (ICCBR-99)*. Springer, 1999, pp. 329–342.

[7] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[8] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis, "Modeling wine preferences by data mining from physicochemical properties," *Decision Support Systems*, vol. 47, no. 4, pp. 547–553, 2009.

Support Vector Machines and Mel-Frequency Cepstral Coefficients: an Application for Automatic Voice Recognition

F. G. Barbosa[1] and W. L. S. Serra[1]
[1]Department of Eletroelectronics, Federal Institute of Maranhao, Sao Luis, Maranhao, Brazil

Abstract— *The speech recognition problem can be modeled as a classification problem, where one wants to get the best degree of separability between classes representing the voice. In order to apply that concept to build an automated speech recognition system capable of identifying the speaker, many techniques using artificial intelligence and general classification have been developed, which lead to this paper. Here we propose a voice recognition method to recognize keywords in brazilian portuguese for biometric purpose using multiple Support Vector Machines, which builds a hyperplane that separates Mel Frequency Cepstral Coefficients, the MFCC's, for later classification of new data. With a small dataset the system was able to correctly identify the speaker in all cases, having great precision on the task. The machines are based on the Radial Basis Function kernel, the RBF, but were tested with severe different kernels, having also a good precision with the linear one.*

Keywords: Voice Recognition; Support Vector Machines; MFCCs; Brazilian Portuguese.

1. Introduction

The foundation of Support Vector Machines (SVM) was developed by Vapnik[1] and earned a lot of popularity due its promissing characteristics, with better empirical performance. The mathematical formulation uses the *Structural Risk Minimization* (SRM), that has shown itself superior to the *Empirical Risk Minimization* (ERM), used by conventional Neural Nets. SRM minimizes an upper limit over the expected risk, while ERM minimizes the error on the training data. This is the difference that leads SVM to have greater generalization capacity, which is the goal of statistical learning.

SVMs were developed to solve the classification problem, but, recently, have been applied to solve regression ones[2]. The classifiers generated by a *Support Vector Machine* achive good results in general, having that capacity of generalization measured by their eficiency on classifying data that does not belong to the training data set. The main idea of a SVM is building a hyperplane, that is a separating surface, as decision bounds in which the separation of the dichotomous examples is maximum. It is important to highlight from the *Statistical Learning Theory*, that a good classifier accounts all the data set but abstains from the particular cases, and as SVMs constitutes a learning technique, which has been getting attention from the science community because it obtains results comparable to, or even better than *Articial Neural Nets*, a lot of sucessful examples can be mentioned on many fields, like categorizing text[3], image analysis[4], [5] and bioinformatics[6], [7]. Due to its efficiency in working with highdimensional data, it is cited in the literature as a highly robust technique[8].

The Theory of Statistical Learning aims to establish mathematical conditions that allow the selection of a classifier with good performance for the data set available for training and testing. In other words, this theory seeks to find a good classifier with good generalization regarding the entire data set. But, this classifier abstains from particular cases, which defines the capability to correctly predict the class of new data from the same domain in which the learning occured. Machines Learning (ML) employs an inference principle called induction, in which general conclusions are obtained from a particular set of examples. A model of supervised learning based on Theory of Statistical Learning is given in Fig. 1.

The voice recognition, and mostly the automatic one, has been the main goal of many scientists and reseachers for almost five decades, and has inspired many wonders in science fiction. Despite all the glamour around this subjects, and even with many intelligent machines that are able to recognize words and understand its meaning, we're still far from achieving the desirable one, that could understand any speech, independently on the speaker or language spoken, in a noise filled environment[9]. In truth, the actual situation is another one: to recognize a simple word or phrase, one needs an absurd computacional effort, where many areas of knowledge are used on training and final recognition. In order to solve this problem and facilitate the speech recognition, the Support Vector Machines technique is used in large scale on pattern classification , and as for this paper, we propose a biometrical voice recogntion automated system, in which we train ten samples of the digits, from '0' to '9', in brazilian portuguese, for each speaker, and later try to identify which one he is and which word he spoke.

2. Fundamentals of Support Vector Machines

The *Support Vector Machines* classification technique stands out by its strong theoretical fundamentation, having

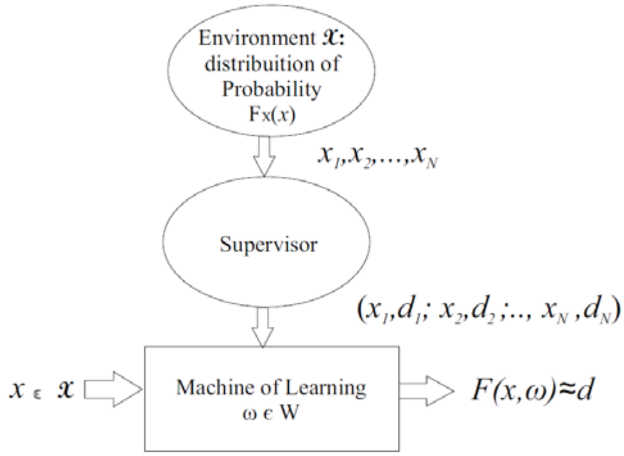


Fig. 1: Flowchart of a model of supervised learning.

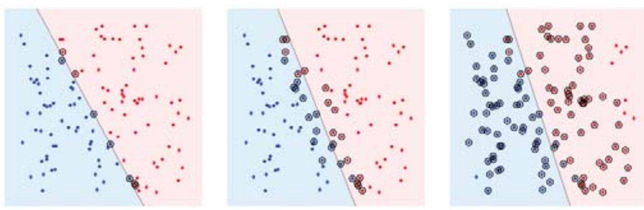


Fig. 2: Comparison of margins to find the optimum separating hyperplane.

on its core the Statitistical Learnig Theory (SLT) being this characteristic a differential above other techniques, as already said, Neural Nets. The ability to work with patterns of high dimensionality is another interesting circunstance of this technique, making it ideal to aplications where a noise data set is the one targetted.

SVM, as a supervised learning technique, can infere from a set of labeled examples, on which the class is known, a function capable of predicting new labels from unknown examples. The simpler derivation of the SVM algorithm is the linear function one, where to ilustrate the separation plane generated by it, we can drawn a line that represents the decision boundary that correctly classifies some data set, such as the one in Fig. 2.

A training dataset composed by two classes, where three decision functions realize the classification correctly between the blue squares and the red circles. However each function determines a different area and different quantity of support vectors for each class represented. Linear decision boundaries consist on a hyperplane (line in two dimensions, plane in 3 dimensions, ...), that separates two regions of the space in question. Such function $g(x)$ can be represented by a mathematical function of vector x e could assume the values $+1$ ou -1.

Support vector machines solve nonlinear problems by transforming the input feature vectors into a dimensionally higher hyperplane, where the linear separation becomes possible. Maximum discrimination is obtained with an optimal placement of the separation plane between the borders of the two classes [10]. If we assume a set H of points $x_i \in R^d$ with $i = 1, 2, 3, \ldots, n$. Each one of the x_i belongs to either of two classes labeled $y_i \in \{-1, 1\}$. Establishing the equation of a hyperplane that divides H is the desired goal, and for this purpose we have some preliminary definitions. By taking the set H, if linearly separable, there exists $w \in R^d$ and $b \in R$ to satisfy

$$y_i (w \cdot x_i + b) \geq 1 \tag{1}$$

where $i = 1, 2, 3, \ldots, n$.

The pair (w, b)defines a hyperplane

$$(w \cdot x_i + b) = 0 \tag{2}$$

This defines a separating hyperplane, leading to the problem of finding the optimal separating hyperplane, to which we try to minimize w as the following

$$min \frac{1}{2} \parallel w \parallel^2 \tag{3}$$

where $y_i (w \cdot x_i + b) \geq 1$.

Then converted to a dual problem by Lagrange multiplies

$$max \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (\mathbf{X_i} \cdot \mathbf{X_j}) \tag{4}$$

where $\sum_{i=1}^{N} \alpha_i y_i = 0, \alpha_i > 0$.

When H cannot be separated linearly, nonnegative slack factor $\xi = (\xi_1, \xi_2, \ldots, \xi_N)$ is introduced. There is

$$y_i (w \cdot x_i + b) \geq 1 - \xi_i \tag{5}$$

The optimal problem can be described as

$$max \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (\mathbf{X_i} \cdot \mathbf{X_j}) \tag{6}$$

where $\sum_{i=1}^{N} \alpha_i y_i = 0, i = 1, 2, \ldots, N,\ 0 \leq \alpha_i \leq C$.

This is the general form of SVM. If C tends to infinite, (6) becomes a linear separating problem, just like (4). Its a problem that can be solved by quadratic programming using sequential minimal optimization.

When the data is easily linearly separable, the previous equations are able to classify with minimum error, but when the data is highly nonlinear one needs to use the kernel method, in which the data is put in a higher dimensional plane, where it can be linearly separated. This is possible when we take the dot product of $X_i \cdot X_j$ and apply another function, validated by the Mercer's Conditions, that in some cases, like the *Radial Basis Function* (RBF), can place the data in a infinity dimensional space, where the data can easily be separated, for this reason it is the one used on this paper.

3. Voice Recognition for Biometrical Authentication

The voice exists for the human desire of verbalizing its thoughts, emotions and opinions, being part of our identity. It is one of the strongest extensions of our personality, and many times it's possible to recognize someone just by their voice.

From the beginning of its technological and intelectual development, the human beings intended to create machines that were able to produce and understand the human speech. Using voice to interact with automatic systems has a vast field of application. The combination with phone network allows remote access to databases and new services, like, for example, an e-mail check from anywhere on the globe and consultations of flight schedule without needing an operator.

Recently, several methods of Speech Recognition have been proposed using mel-frequency cepstral coefficients and Neural Networks Classifiers [11], [12], [13], Sparse Systems for Speech Recognition [14], Hybrid Robust Voice Activity Detection System [15], Wolof Speech Recognition with Limited vocabulary Based HMM and Toolkit [16], Real-Time Robust Speech Recognition using Compact Support Vector Machines.

On this context, the amazing advances in the last years, mostly in the microeletronics field, made possible to put into pratice this line of thought, effectively. At this point, one needs to adress the field of *Digital Signal Processing,* that is the core of many areas in science. From the engineering point of view, signals are functions or series used to carry information from a source to the recipient. The signals specific characteristics depend on the communication used for the transmission. They are processed on the transmission side to be produced and configured, and on the receptor they're decodified to extract the information contained, with maximum efficiency, if possible.

3.1 Digital Signal Processing

Method that consists in analysing real world signals (represented by a numerical sequence), extract its features through mathematical tools, in order to extract the essential information. There are many purposes on the matter such as biometric authentication, image processing and recognition and even preventing diseases. As for speech, subject of this paper, we follow basically three steps: sampling, followed by segmentation of words or phonemes[17] and short term analysis by Fourier transform or spectral analysis[18]. After this step, responsible for digital processing of the speech signal, we need to recognize and correctly classify a word, and for that there are some existing techniques, capable of extracting parameters based on a certain model and then applying a transformation to represent the signal in a more convenient form for recognition.

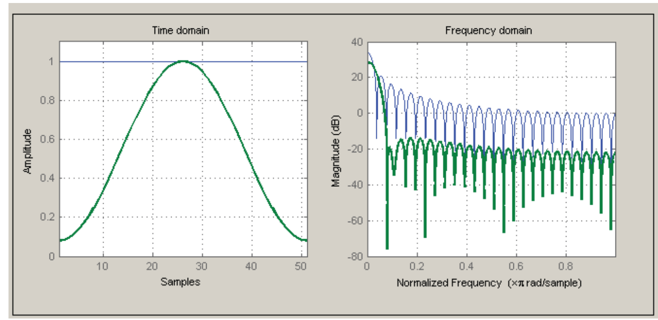


Fig. 3: Hamming Window and equivalent *SNR*

3.2 Pre-processing of the Speech Signal

The moment the segmentation of the speech is passed through the process of windowing, responsible for 'dividing' the signal with minimum power loss and noise, the speech signal is sampled and segmented into frames and is encoded in a set of mel-cepstral parameters. The number of parameters obtained is determined by the order of mel-cepstral coefficients. The obtained coefficients are then encoded by Discrete Cosine Transform (DCT) [18] in a two dimensional matrix that will represent the speech signal to be recognized. The process of windowing, hamming windowing for this case, in a given signal, aims to select a small portion of this signal, which will be analysed and named frame. A short-term Fourier analysis performed on these frames is called signal analysis frame by frame. The length of the frame T_f is defined as the length of time upon which a parameter set is valid. The term frame is used to determine the length of time between successive calculations of parameters. Normally, for speech processing, the time frame is between 10ms and 30ms[19]. There's also the superposition of the windowing, which determines where the window will start in order to reduce the power loss, initiating before the previous window reaches its end. Fig. 3 shows the plot of a hamming window in time and frequency domains.

3.3 SVMs and Biometry

Biometric authentication is any form of human biological measurement or metric that can be used to identify and authenticate an authorized user of a secure system. Biometric authentication- can include fingerprint, voice,S iris, facial, keystroke, and hand geometry[22]. Concerns on widespread use of biometric authentication systems are primarily centered around template security, revocability, and privacy[22]. The use of cryptographic primitives to bolster the authentication process can alleviate some of these concerns as shown by biometric cryptosystems[23]. Support Vector Machines or SVM is one of the most successful and powerful statistical learning classification techniques and it has been also implemented in the biometric field[24]. As for voice recognition, the technique has shown excellent results, hence not only it can generalize, but it can also restrict the parameters if correctly made, leading to a great biometrical authentication voice based system.

4. Methodology

As a recognition default we proposed the classification and identification of the voice of a speaker by a keyword, in a text-dependent system. The speech signal is sampled and encoded in mel-cepstral coefficients and coefficients of Discrete Cosine Transform (DCT)[18] in order to parameterize the signal with a reduced number of parameters. Then, it generates two dimensional matrices referring to the Discrete Cossine Transform coefficients. The elements of these matrices representing two-dimensional temporal patterns will be classified by Support Vector Machines (SVMs)[20]. The innovation of this work is in the reduced number of parameters which lies in the SVM classifier and in the reduction of computational load caused by this reduction of parameters. The classification is made based o the *Radial Basis Function.*

4.1 The DCT matrix

After being properly parameterized in mel-cepstral coefficients, the signal is encoded by DCT performed in a sequence of T observation vectors of mel-cepstral coefficients on the time axis. The coding by DCT is given by the equation following:

$$C_k\left(n,T\right)=\frac{1}{N}\sum_{t=1}^{T}MFCC_k\left(t\right)cos\frac{\left(2t+1\right)n\pi}{2T} \tag{7}$$

where k, $1\leq k\leq K$, refers to the k-th line (number of Mel frequency cepstral coefficients) of t-th segment of the matrix n, $1\leq n\leq N$ component refers to the n-th column (order of DCT), $MFCC_k(t)$ represents the mel-cepstral coefficients. Thus, one obtains the two-dimensional matrix that encode the long term variations of the spectral envelope of the speech signal [21]. This procedure is performed for each spoken word. Thus, there is a two-dimensional matrix $C_k(n,T)\equiv C_{k^n}$ for each input signal. The matrix elements are obtained as the following:

1) For a given model of spoken word W (digit), ten examples of this model are pronounced. Each example is properly divided into T frames distributed along the time axis. Thus, we have: P_i^j, where $i=0,1,2,...,9$ is the number of patterns to be recognized and $j=1,2,3,...,10$, is the number of samples to generate each pattern.

2) Each frame of a given example of model W generates a total of K mel-frequency cepstral coefficients, and then, significant characteristics are obtained within each frame over this time. The DCT of order N is then calculated for each mel-cepstral coefficient of the same order within the frame, that is, C_1 in the frame t_1, C_1 in the frame $t_2,...,C_1$ in the frame t_T, and so on, generating elements $\{C_{11},...,C_{1N}\}$, $\{C_{21},...,C_{2N}\}$, $\{C_{K1},...,C_{KN}\}$ in the matrix given in (7).

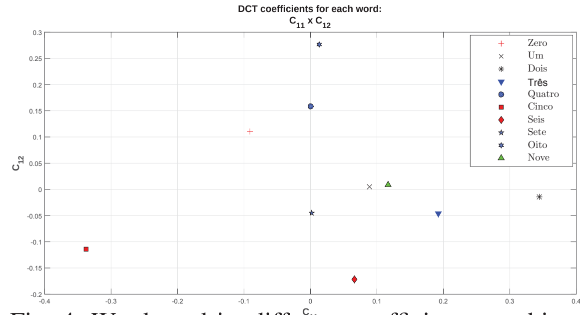


Fig. 4: Words and its different coefficients combinations

Thus, a two-dimensional temporal array DCT is generated for each j example of model W, represented by C_{KN}^{ij}.

4.2 Generating the Support Vector Machines

As SVM calls for a bidimensional space, two parameters will place the speech signal's characteristcs on a $2D$ representation of space, where the hyperplane will try and separate them in the best possible way. For these characteristcs we have the DCT n-order square matrix with its elements, each set composing a word, where the $n's$ used were $2,3$ and 4.

One of the differentials of this paper is the use of Brazilian Portuguese language, an area that lacks works of this kind and has limited database. The keywords used on the experiment are the digits from '0' to '9'.

The coefficients generated for each person and each word are compared one by one with each other, in a methodology that's called one versus all technique. For example, one speaker has a dataset composed by ten samples of each of the ten digits, and the coefficients of the ten samples are extracted and disposed on the plane for later separation. They are put on the plane in pairs of characteristcs (the coefficients) in six different combinations for a 2 by 2 matrix, as shown in the Fig. 4, the plot of the mean for each coefficient and each word. First we extract the coefficients of each word for two different speakers, then we compare one of the words spoken by one of the speakers with all the words spoken by the other speaker, and so on for all the other speakers. The pairs of characteristcs are every possible nonrepeated combination of the DCT-matrix elements, first the $C_{11}xC_{12}$ then $C_{11}xC_{21}$, and so on: $C_{11}xC_{22}$, $C_{12}xC_{21}$, $C_{12}xC_{22}$ and $C_{21}xC_{22}$. Each combination expresses a part of the biometrical authentication, and the algorithm classifies a voice based on the majority of the matches. In Brasolin[25], the use of SVM with wavelet digital voice recognition in Brazilian Portuguese, obtained an average of 97.76% using 26 MFCC's in the pre-processing of voice and SVM machine's with the following characteristics: MLP as Kernel functions, ten machines (one for each class) and "one vs. all" as method of multiple classes. Also, the author tried to generalise instead of restricting. In comparison to this work, the results of this remain more effective, because

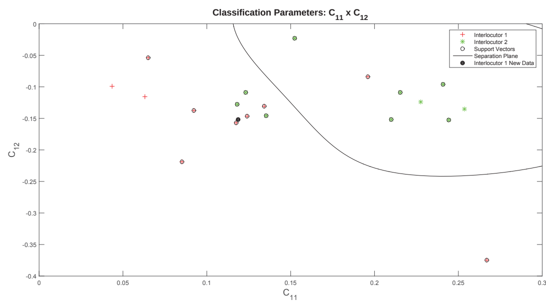


Fig. 5: One of the generated hyperplanes correctly classifying the new data

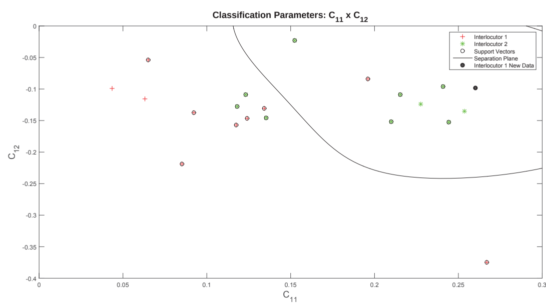


Fig. 6: One of the few misclassifications

the amount of MFCC's is smaller (only a 2 by 2 matrix) and, also, the input of parameters in the machines are lower. Consequently, the computational load is lower. But one cannot really compare because of the objetive intended of each one.

5. Training and testing

After performing the extraction of the parameters and putting them in the pairs, the Support Vector Machine algorithm is applied in order to generate the hyperplane and classify the new data. As shown in Fig. 5 and Fig. 6.

The black dots represent the new data input entering the system, and, for most of the cases, was correctly classified, showing in overall a precise voice recognition system, able to identify the speaker with higher than 90% probability, misclassifying few of the parameters, later compensated by the other combinations of coefficients as show in Table 1 and 2. The mentioned tables contain the percentage of the words correctly classified, for example, for the first speaker the system correctly classified all the keywords spoken, regarding $C_{11}xC_{12}$, as for the fourth speaker, the system matched correctly nine of the ten words, hence 90 percent.

Table 3 shows the result after the 6 combinations of pairs are made, to achieve more confiability on the identification. Were used on the training, as mentioned before, thirteen different voices from thirteen different speakers and ten different keywords, the digits from '0' to '9' (zero, um, dois, trÃłs, quatro, cinco, seis, sete, oito, nove, in brazilian portuguese), each one spoken ten times by the same speaker in order to generate good parameters for each digit. Tables 4 and 5 show the computational time needed for one of the runs of the algorithm, and may vary depending on other tasks executed at the same time. The tests were made based on the same procedure, where no other tasks or softwares, but the operational system fundamentals, were initialized, hence reducing delays due processing sharing.

When using the combinations of DCT coefficients, most of the times it is easy for the program to generate the hyperplane, thus little time of training. That happens because the points are well placed on the plane, creating clusters that

Table 1: Overall results for $C_{11}xC_{12}, C_{11}xC_{21}, C_{11}xC_{22}$

Speaker	C11x12	C11xC21	C11xC22
1	100	90	100
2	100	100	100
3	100	90	100
4	90	100	100
5	100	90	100
6	90	90	90
7	100	100	100
8	90	80	90
9	90	100	100
10	100	100	100
11	100	80	90
12	80	80	90
13	100	80	80

Table 2: Overall results for $C_{12}xC_{21}, C_{12}xC_{22}, C_{21}xC_{22}$

Speaker	C12x21	C12xC22	C21xC22
1	100	90	100
2	100	100	100
3	90	100	90
4	100	100	100
5	100	100	100
6	80	100	100
7	100	90	100
8	100	100	100
9	100	100	100
10	100	100	100
11	100	90	80
12	90	90	100
13	90	100	90

Table 3: Percentage results for all pairs combined

Speaker 1	95
Speaker 2	100
Speaker 3	95
Speaker 4	98
Speaker 5	98
Speaker 6	91
Speaker 7	98
Speaker 8	93
Speaker 9	98
Speaker 10	100
Speaker 11	90
Speaker 12	88
Speaker 13	90

Table 4: Computation time for training

Speaker	Time in seconds
1	5,60
2	5,28
3	5,62
4	5,56
5	6,18
6	5,59
7	6,06
8	5,57
9	5,63
10	5,62
11	5,58
12	6,12
13	5,61

Table 5: Overall prediction time

Speaker	Time in milliseconds
1	9,30
2	8,14
3	8,65
4	8,20
5	8,25
6	9,53
7	10,64
8	12,65
9	11,68
10	13,66
11	9,25
12	8,86
13	9,22

are visually easy to separate, where sometimes even a linear kernel can obtain excellent results. Most of the computing time it's on the extraction of the parameters from the voice signal, i. e., calculating the MFCCs and the DCT transform. The values from Table 4 are the conjoint time of extraction and hyperplane generation, and the values of Table 5 are times of prediction for one keyword.

6. Conclusion

Biometrical classification utilizing voice as a input parameter a SVMs to classification, has shown sucess in general for identifying the speaker. Also the restrictions set by the classifiers, restricts in such a way that prevents false positive to rule over the actual positive results. The dichotomical nature of the technique leads to a excellent response time of computational execution, although the time the algorithm took for training all the datasets and comparing then with new data was aproximatelly two hours, one must remeber the absurd quantity of Suppor Vector Machines, exactly 912600, hence the delay. However for one keyword and one speaker at time, the training can last a insignificant time when compared with other techniques, as presented on the tables afore mentioned, so as the classification, revealing the real time application possible, fast, precise and very reliable. The computer used for training and prediciton has 6 GB of ram and a Intel Core i5^{TM}. The data was sampled at a 22050Hz frequency, with 16 bits of resolution.

With an overall greater than 90% of sucess rate, the system was acomplished and the premise validated, and in order to improve the work, more training data can be given to the system. As for the use of *Kernel Functions*, the used one for the final results was the *Radial Basis Function*, but in order to reduce training and predicting time, one can use the *Linear Kernel* with little loss of precision and reliability.

Acknowledgment

The authors thank the Scientific Initiation Program of the Federal Institute of Maranhao for financial support through grant aid and also the eletrical engineering student Libanio Vieira who helped substantially on the paper.

References

[1] B. E. Boser, I. L. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In Proceedings of the 5th Annual Workshop on Computational Learning Theory, pages 144–152, Pittsburg, Pennsylvania, US, 1992.

[2] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. Knowledge Discovery and Data Mining, 2(2):1–43, 1998.

[3] T. Joachims. Learning to classify texts using support vector machines: methods, theory and algorithms. Kluwer Academic Publishers, 2002.

[4] K. I. Kim, K. Jung, S. H. Park, and H. J. Kim. Support vector machines for texture classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(11):1542–1550, 2002.

[5] M. Pontil and A. Verri. Support vector machines for 3-D object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(6):637–646, 1998.

[6] W. S. Noble. Support vector machine applications in computational biology. In B. SchÃűlkopf, K. Tsuda, and J.-P. Vert, editors, Kernel Methods in computational biology, pages 71–92. MIT Press, 2004.

[7] B. , I. Guyon, and J.Weston. Statistical learning and kernel methods in bioinformatics. In P. Frasconi and R. Shamir, editors, Artificial Intelligence and Heuristic Methods in Bioinformatics, pages 1–21. IOS Press, 2003.

[8] C. Ding and I. Dubchak, Multi-class protein fold recognition using support vector machines and neural networks. Bioinformatics, 2001.

[9] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines. Available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm/. Acessed in: 09/2003, 2004.

[10] Jian Zhou; Wang, Guoyin; Yong Yang; Peijun Chen, Speech Emotion Recognition Based on Rough Set and SVM, Cognitive Informatics, 2006. ICCI 2006. 5th IEEE International Conference on , vol.1, no., pp.53,61, 17-19 July 2006.

[11] D. Hanchate, M. Nalawade, M. Pawar, V. Pohale, and P. Maurya, "vocal digit recognition using artificial neural network"." 2nd International Conference on Coumputer Engineering and Technology, April 2010, pp. 88–91.

[12] R. Aggarwal and M. Dave, "application of genetically optimized neural networks for hindi speech recognition system"." World Congress on Information and Communication Technologies (WICT), December 2011, pp. 512–517.

[13] S. Azam, Z. Mansor, M. Mughal, and S. Moshin, "urdu spoken digits recognition using classifield mfcc and backpropagation neural network"." 4th International Conference on Computer Graphics, Imaging and Visualization (CGIV), August 2007, pp. 414–418.

[14] M. Mohammed, E. Bijov, C. Xavier, A. Yasif, and V. Supriya, "robust automatic speech recognition systems:hmm vesus sparse"." Third International Conference on Intelligent Systems modelling and Simulation, February 2012, pp. 339–342.

[15] C. Ganesh, H. Kumar, and P. Vanathi, "performance analysis of hybrid robust automatic speech recognition system"." IEEE International Conference on Signal Processing, Computing and Control (ISPCC), March 2012, pp. 1–4.

[16] J. Tamgo, E. Barnard, C. Lishou, and M. Richome, "wolof speech recognition model of digits and limited-vocabulary based on hmm and toolkit"." 14th International Conference on Computer Modelling and Simulation (UKSim), March 2012, pp. 389–395.

[17] P. Fantinato, Segmentacao de Voz baseada na Analise Fractal e na Transformada Wavelet. Prentice Hall, Outubro 2008.

[18] L. Rabiner and R. Schafer, Digital Processing of Speech Signals. Prentice Hall, 1978.

[19] J. Picone, "Signal modeling techniques in speech recognition." IEEE Transactions on Computer, April 1991, pp. 1215–1247.

[20] S. Haykin, Redes Neurais:Principio e pratica. Bookman, 2002.

[21] P. Fissore and E. Rivera, "Using word temporal structure in hmm speech recongnition." ICASSP 97, April 1997, pp. 975–978.

[22] M.A.,Kowtko, Biometric authentication for older adults, Systems, Applications and Technology Conference (LISAT), 2014 IEEE Long Island , vol., no., pp.1,6, 2-2 May 2014.

[23] Upmanyu, M.; Namboodiri, A.M.; Srinathan, K.; Jawahar, C.V., Blind Authentication: A Secure Crypto-Biometric Verification Protocol, Information Forensics and Security, IEEE Transactions on , vol.5, no.2, pp.255,268, June 2010

[24] Fahmy, M.S.; Atyia, A.F.; Elfouly, R.S., Biometric Fusion Using Enhanced SVM Classification, Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08 International Conference on , vol., no., pp.1043,1048, 15-17 Aug. 2008

[25] A. Brasolin, A. Neto, and P. Alsin, ""digit recognition using wavelet and svm in brazilian portuguese"." ICASSP 2008, April 2008, pp. 1–4.

Toward a Short Text Classification Framework Based on Background Knowledge Discovery

Isak Taksa
Computer Information Systems, Baruch College, CUNY, New York, NY, USA

Abstract - *The ubiquitous, diverse and growing impact of digital living creates a massive amount of short text - a search query, a twit or a caption. Short text frequently presents itself as an arbitrary combination of semantically unconnected words. Using machine learning to classify the corpora of such texts is a challenging task. A large body of research exists in this area, but in this paper we will focus on Background Knowledge (BK) and its role in machine learning for short-text and non-topical classification. More specifically, we present an effort to create a short text classification framework based on Background Knowledge. We propose novel Information Retrieval techniques to construct BK and demonstrate the advantages of Automatic Query Expansion (AQE) vs. basic search. We discuss other results of this research and its implications on the advancement of short text classification.*

Keywords: Machine learning, short text, classification, background knowledge, information gain

1 Introduction

Text classification in the framework of machine learning is an active area of research, encompassing a variety of learning algorithms [20], classification systems [19] and data representations [16]. This paper examines the classification of search queries, which is one example of text classification that is particularly complex and challenging. Typically, search queries are short, reveal very few features per single query and are therefore a weak source for traditional machine learning [6].

We examine the issues of non-hierarchical [17] classification and investigate a method that combines limited manual labeling, computational linguistics and information retrieval to classify a large collection of search queries. We discuss classification proficiency of the proposed method on a large search engine query log, and the effect of the variations of this method on the quality and efficiency of short-text classification.

We start with a search engine query log which is viewed as a set of textual data on which we perform classification [13]. Observed in this way each query in a log can be seen as a document that is to be classified according to some pre-defined set of labels, or *classes*. Viewing the initial log with the search queries as a document corpus $D = \{d_1, d_2, \ldots d_i, \ldots d_n\}$, we create a set of classes that indicate a personal demographic characteristic of the searcher, $C = \{c_1, c_2, \ldots c_j, \ldots c_m\}$. Using Web searches our approach retrieves a set of background knowledge to learn additional features that are indicative of the classes, *C*. This allows for the categorization of the queries. The approach consists of the following five steps:

I) Select (from print and online media) a short set of manually chosen terms $T_{init} = \{t_1, t_2, \ldots, t_j, \ldots, t_m\}$ consisting of terms t_j that are known a priori to be descriptive of a particular class c_j;

II) Use this initial set T_{init} to classify a small subset of (search queries) set *D* thereby creating an initial set of classified queries $Q_{init} = \{q_1, q_2, \ldots q_j \ldots q_l\}$;

III) Consists of sub-step A and sub-steps B

- Sub-step A – Automatic query expansion (AQE) step. Submit these queries q_j to a commercial search engine and use the snippets of the returned search results to build an initial set of expanded classified queries $EQ_{init} = \{eq_1, eq_2, \ldots eq_j \ldots eq_l\}$;
- Sub-step B – Submit these queries eq_l to a commercial search engine and use the returned search results (*n* for each query eq_l) to build a temporary corpus of background knowledge $EB_{temp} = \{eb_1, eb_2, \ldots eb_j \ldots eb_l\}$;

IV) Use an algorithm to select from *EB* more class related terms *T*;

V) Use this newly created set *T* to classify more documents (search queries) in corpus D thereby adding more classified queries to set *Q*.

While steps I and II are executed only once, steps III through V are repeated continuously until the classification process is terminated.

We focus on validating our approach to the classification of a set of short documents, namely search queries. This approach uses a combination of techniques: we first look at developing several methods to obtain relevant background knowledge for a set of web queries; then we build the background knowledge to acquire ranked terms for improved information retrieval; we then investigate the impact of the new terms' selection algorithms on the effectiveness of the classification process.

2 Background

The *Text Classification* problem has been studied extensively by machine learning researchers over the last decade. We can define the categorization as follows: Given a set of documents *D* and a set of *m* classes (or labels) *C*, define a function *F* that will assign a value from the set of *C* to each document in *D*. For example, *D* might consist of the set of all *Classified Advertisements*, and *C* could be the set of types of classified advertisements (automobile, home furnishing, help wanted, etc.) Although the text classification problem can be defined easily, in practice there is often not enough information to find the function *F*.

A text document can be expressed as a feature-value vector, where the features correspond to particular words (or phrases), and the value corresponds to the presence/absence of the word (or some weight that corresponds to the word). When documents are expressed in this fashion the machine learning community can apply well-known algorithms to this problem [15]. The straightforward approach deals with text classification problems as *supervised* learning problems. In this case the human expert simply has to label a set of examples with appropriate classes. This set of labeled examples is called the *training set*, which we will refer to as the set *T*. Once a training corpus of correctly labeled documents is available, there are a variety of techniques that can be used to create a set of rules or a model of the data that will allow future documents to be classified correctly. The techniques can be optimized and studied independently of the domains and specific problems that they will be used to address. A plethora of different learning algorithms including Bayesian classifiers [7], nearest neighbor [23], and support vector machines [1], have been applied to many different representations of textual documents, successfully allowing for the classification of documents in varied domains.

Topical text categorization problems, such as the example given above, that have a sufficient number of training examples, are now well-understood by machine learning researchers, however, non-standard problems have been the focus of more recent research. A common problem when using machine learning for text classification is dealing with an insufficient number of training examples to correctly classify instances with unknown classes. If there are too few examples, machine learning algorithms often cannot represent the classes properly, and therefore have a high error rate when attempting to classify new examples.

There are a number of approaches that may be taken to aid in the creation of more accurate classifiers. Researchers have noted that although it is often the case that there are very few labeled examples, there are often many unlabeled examples readily available [4]. An approach that has been taken by a number of researchers has been to choose, in some way, a *small* number of additional training examples that should be hand-labeled in order to add particular examples to the *labeled* training set that will improve learning. These hand labeled examples then become part of the training corpus. In this way fewer examples must be given to an expert to be labeled than if the examples were simply randomly sampled.

Other approaches have been taken in these hard-to classify domains. There have been studies on the incorporation of domain knowledge by the selection and creation of cross-referencing query [11], and domain documents [3], or reweighing of features using related information such as ontologies [9] or user feedback [5]. Domain knowledge has also been incorporated into text classifiers by modification of the classifiers to include prior results [24]. There has also been work done using query-expansion type techniques to incorporate additional knowledge into text classifiers [22] and query formulation techniques using terms found in previously retrieved documents [14].

Often, with short text classification problems, there are related textual documents that are not examples that can be classified. We term this set of related documents research *background knowledge* and use it to aid a short-text classifier. Background knowledge has been previously used [12] to improve classification of unknown instances. These sets of background text are not of the same length and form as the training and unlabeled examples, but can be used to find common co-occurrences of terms, as well as terms that are indicative of specific classes.

Short text classification is a challenging type of classification because very little information (i.e. words) is known for each example that is to be classified. Researchers have recognized that since short text examples tend to share few terms, it is particularly difficult to classify new instances and common comparisons between texts often yield no useful results. Simply comparing a training set to unknown examples using traditional methods such as cosine similarities can therefore be useless. An example of short text classification that is receiving interest lately is the query classification [21].

3 Methodology

Our approach in this research is different from the traditional machine learning approaches described above. Instead of actually incorporating the background knowledge set into the learning algorithm, we use background knowledge for the purpose of finding previously unknown class related terms. As described earlier, we begin with only a small set of manually selected class related terms (or phrases). These terms are used to label a small set of documents – search queries extracted from a large Excite query log collected in the morning and afternoon hours of one day, and contains close to 2.5 million queries. This small set of labeled documents is then used as search queries to retrieve a much larger set of longer, related documents. We analyze the larger set of related documents to learn additional class related terms for the classification task.

3.1 Bootstrapping from known class-related terms

To create the set of classes we used Levinson's *Life Structure Theory*. After studying a group of men and women Levinson introduced his theory [10] as consisting of equilibrium/disequilibrium periods during which a man builds/questions his life structure. At the center of his theory is the *life structure*, the underlying pattern of an individual's life at any particular time. For our classes we use Levinson's seasonal cycles – *C {Early adulthood, Adult world, Settle down), Middle adulthood), Culmination, Late adulthood}* or simpler *C= {EA, AW, SD, MA, CL, LA}* which corresponds to the following age brackets *A {17-22, 23-33, 34-45, 46-55, 56-65, 66+}*.

We then acquired terms that are indicative of each of these classes. In particular, we obtained the terms (words or phrases) from well-known printed publications (Seventeen, Parenting, Family Circle, American Association of Retired Persons Magazine, etc.) and popular social media sites (Facebook, Linkedin, TopMommyBlogs, etc.). For each of the classes in set *C*, we manually (and arbitrarily) selected a list of words and phrases that are indicative of each of these classes, e.g. T_{init} = { *EA(*Cliff notes*), AW(*Wedding*), SD(*Investments*), MA*(Eldercare), *CL*(Inheritance), *LA*(Pain) *}*

Using the list of terms T_{init} for each class, we culled a set of queries Q_{init} from the Excite log that contained these age-indicative terms. We began with a very small set of returned labeled queries; 60 in all, 10 per each of the six classes, e.g. Q_{init} = { *EA(*Cliff notes wuthering heights*), AW(Hotel catered wedding), SD(*Investment policy*), MA*(Eldercare cost), *CL*(Inheritance tax irs), *LA*(Chronic back pain) *}*. This set of queries is our set of classified training documents that we use to start the classification of many other queries that do not contain the original list of class related terms.

3.2 Automatic retrieval of background sets

We submitted the classified queries to the Google search engine to automatically create a background set of knowledge. For each of the classified queries we created a pool of documents, each of which was the text of a search result obtained by submitting the classified query to Google. We restricted search results to documents written in the English language. Google returned the top results of the search on the classified query that were downloaded and stored. We saved the textual sections of the pages that were downloaded, and each one became a document, classified according to the class of the query that generated it. We limited our search results to the top 10 results returned for each query since users are generally satisfied if the desired page is found within the top 10 results [8].

After downloading search results we had a set of ten text documents for each one of our queries. These were then used as a corpus for analysis. This method allows for the retrieval of documents that are class related, but are much longer than the original queries. The queries from the search log are an average length of ~3 words, whereas the new documents that were downloaded had an average length of several thousand words, and hence more could be learned from them.

3.3 Finding new class-related terms

Each page that was returned by Google was labeled with the class category of the query that produced it. This set of pages can be looked at as a new and different document *training* corpus with known labels. The training set T consists of the returned search pages, and the classes *C* are the classes that were used to label the original small set of hand-labeled queries. However, the properties of this training corpus are markedly different than the original query training set. Essentially, this newly created training corpus does not consist of short-text examples. As opposed to our original data set, where examples were queries only a few words long, this larger returned corpus contains entries that are web-page length. Hence there is much more generalization that we can draw from the words in this larger returned document corpus.

What is especially interesting is the new, larger, document corpus vocabulary. A serious disadvantage of short-text corpora is that they do not contain a rich enough vocabulary to facilitate learning, however, with a longer document corpus we can learn much about the domain from the set of words that are in it. In essence, our method of page retrieval allows us to swap a short-text corpus for one with longer entries from which we can learn.

Our approach studies the set of terms that compose the returned document corpus to find those particular terms that are related to our classification problem. We began by using the information gain (IG) criterion to rank *all* terms in the corpus; no stemming was used to facilitate query creation later. For a supervised text classification task, each term that is present in the training corpus can be seen as a feature that can be used individually for classification. For example, suppose that the term *investment* occurs in the training corpus. We can partition the training corpus into two disjoint subsets, one of which contains the word *investment*, and one of which does not. Given the training set of classified examples, *T*, we can partition it by the presence or absence of each term, *t* that exists in these examples. We can then determine how closely related this term is to the classification task.

To do this, we borrow a concept from Information Theory, called *information gain*, which has been used by machine learning researchers for the purposes of classification (Quinlan, 1986). Given a probability distribution $P = (p_1, p_2, \ldots, p_n)$ then the information conveyed by this distribution, also called the *entropy* of P, is:

$$entropy(P) = -(p_1 \times \log(p_1) + p_2 \times \log(p_2) + \ldots p_n \times \log(p_n)) \quad (1)$$

Essentially, this measure is a measure of the randomness of the distribution. High entropy signifies that the distribution is random, whereas low entropy signifies that there is some pattern in the data. In the field of information theory, the entropy is a measure of how many bits it takes to transmit a message with the probability distribution *P*. If we wish to discover the entropy of a training set *T*, then the probability distribution *P* is simply the set of probabilities that a training example fits into any of the classes of set *C*. From these training set probabilities we can compute *entropy (T).*

Each term *t* gives a partition of the training set *T*, $\{T_0, T_1\}$, where T_0 consists of those training examples that contain the term *t*, and T_1 consists of those training examples that do not contain the term *t*. For each of these subsets, we can compute individual entropies, and the summation of those entropies, weighted by the probability distribution gives us the information needed to identify the class of a training example after the partition is done.
The information gain (*IG*) for a term *t* tells us how much information is gained by partitioning the training set *T* on the term *t*. It is defined as the subtraction:

$$IG(t) = entropy(T) - (entropy(T_0) \times \frac{|T_0|}{|T|} + entropy(T_1) \times \frac{|T_1|}{|T|}) \quad (2)$$

Terms with high information gain create partitions of the original training set that overall have lower entropy, and therefore are reflective of the particular classification scheme.

The computation of the *IG* value for each of these terms allows us to learn important features in this background corpus. However, our challenge was to determine which of these features best reflected each class. To discover which terms give us information about particular classes, we sorted all terms in the corpus in descending order based upon the *IG* value. We labeled each of the terms with the class whose training examples most reflected this term, i.e. whose training examples actually most often contained that term.

We then chose the top terms for each of the classes. At this point we selected a list of fifty terms (per class) to classify queries that were not classified before. contain no terms that can be deemed class related, or may contain terms that fit two or more classes. It would be impossible to classify these types of queries.

The other important factor that affects the match of queries in the log and terms in the background set, is the fact that we are using today's Web collection and search engine to produce the background set, but the query log was collected several years ago. The Web collection grew exponentially; search engines are fine-tuned to return results that reflect contemporary culture and language [2].
It is important to note that some of the text documents did not contain the terms that were associated with their class. We are not concerned with this fact, however, because we are simply looking for good indicative terms that are related to particular classes.

4 Discussion

Four parameters describe every scenario: $\boldsymbol{S_{eval}}$. = {Number of queries to retrieve background knowledge (*NBK*), Query selection process (*QSP*), Number of top classification terms (*NCT*), and Terms selection process (*TSP*)}.

- NBK represents the number of queries that are selected from the queries classified in the prior iteration of our algorithm. These queries are submitted to Google to retrieve "background knowledge". We use a variable number of queries to examine whether increasing the size of the retrieved background knowledge would generate better quality classification terms.
- QSP specifies the process of selecting new search queries from amongst the newly classified queries. The selection criterion is the query length.
- NCT represents the number of new terms that will be used in the next classification iteration. After calculating the Information Gain (IG) for every term in background knowledge, we sort the list in descending order of IG. We use various size lists (between 30 and 100 terms).
- TSP specifies the selection of new classification terms from the sorted list produced by IG calculations. The selection criterion is the term frequency in the Google collection. In some scenarios we use "light" terms (low term frequency) while in others we use "heavy" terms (high term frequency).

4.1 Appraising classification results

- To bootstrap all the scenarios we used a manually selected set consisting of 60 terms (10 per each class). Even though this set is negligible in size, it allowed for classification of over 8% of the query log.
- Increasing the number of queries above five doesn't produce significant classification improvements.
- Longer queries produce better background knowledge for some of the classes.
- "Heavy" queries produce more noise (cross-class classification).
- "Light" queries produce IG lists with smaller entropy on top of the list.

Furthermore, as we reflect on the nature of the data, there are several objective factors that make this classification task a difficult one. First, according to a topical study of the same log, a large number of queries are intra-class in nature (e.g. 20.3% People and places, 7.5% Sex and pornography, 6.8% Non-English or unknown) [18], and therefore are not easily classifiable according to our original set of classes, *C*. In particular, many of the queries in the Excite log may

5 Conclusion

Starting with a small, manually selected set of terms, we develop and present an approach that classifies a set of web queries. This text classification task is difficult for three reasons: the dataset does not contain many labeled examples, the text examples are extremely short, and classification is non-topical by characteristics of the users. By iteratively applying our approach and improving performance of the ranking algorithm we are able to classify many queries in a large query log. However, analysis of several distinct background discovery scenarios did not produce a clear winner.

An essential area of further research is how to evaluate the efficiency of the classification methods (relatively easy) and the quality of the classification results (much more complicated due to its subjectivity). To accomplish the first task we plan to utilize the *k*-Means algorithm. There are several ways to explore the second task. We can use a limited number of the classified queries to retrieve web sites and manually compare these sites to well-known age-related sites. Alternatively, we can use a large number of the classified queries to retrieve web sites and algorithmically compare these sites to a set of age-related web sites to check for cross-classification of results.

6 References

[1] Aphinyanaphongs, Y., Ray, B., Statnikov, A. & Krebs, P. "Text classification for automatic detection of alcohol use-related tweets: A feasibility study"; In Information Reuse and Integration (IRI), IEEE 15th International Conference on Information Reuse and Integration, 93–97, Aug 2014.

[2] Chambers, J. K. & Schilling, N. (Eds.). "The handbook of language variation and change", Vol. 80, Wiley-Blackwell, 2013.

[3] Dinh, D. & Tamine, L. "Towards a context sensitive approach to searching information based on domain specific knowledge sources"; Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 12, 41–52, 2012.

[4] Fujino, A., Ueda, N. & Nagata, M. "Adaptive semi-supervised learning on labeled and unlabeled data with different distributions"; Knowledge and Information Systems, Springer, Vol. 37, 129–154, 2013

[5] Imran, M., Castillo, C., Lucas, J., Patrick, M. & Rogstadius, J. "Coordinating human and machine intelligence to classify microblog communications in crises", Proceedings of the 11 International ISCRAM Conference, 712–721, May 2014.

[6] Kateb, F. & Kalita, J. "Classifying Short Text in Social Media: Twitter as Case Study"; International Journal of Computer Applications, Vol. 111, Issue 9, 1–12, 2015.

[7] Kazemian, H. B. & Ahmed, S. "Comparisons of machine learning techniques for detecting malicious webpages", Expert Systems with Applications, Vol. 42, Issue 3, 1166–1177, 2015.

[8] Kim, J. Y., Cramer, M., Teevan, J. & Lagun, D. "Understanding How People Interact with Web Search Results that Change in Real-Time Using Implicit Feedback", In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management, CIKM '13, 2321-2326, ACM, 2013.

[9] Lajis, A, & Aziz, N. A. "Competency assessment of short free text answers", In Engineering Technology and Technopreneuship (ICE2T), 4th International Conference, 41–45, IEEE, Aug 2014.

[10] Levinson, D.J. "The Seasons of a Man's Life", New York: Ballantine Books, 1986.

[11] Makris, C, & Simos, M. A. "Novel Techniques for Text Annotation with Wikipedia Entities", In Artificial Intelligence Applications and Innovations, 508–518, Springer Berlin Heidelberg, 2014.

[12] Meng, W., Lanfen, L., Jing, W., Penghua, Y., Jiaolong, L. & Fei, X. "Improving Short Text Classification Using Public Search Engines", In Integrated Uncertainty in Knowledge Modeling and Decision Making, 157–166. Springer Berlin Heidelberg, 2013.

[13] Ortiz-Cordova, A. & Jansen, B. J. "Classifying web search queries to identify high revenue generating customers", Journal of the American Society for Information Science and Technology Vol. 63, Issue 7, 1426–1441, 2012.

[14] Qvarfordt, P., Golovchinsky, G., Dunnigan, T. & Agapie, E. 'Looking ahead: query preview in exploratory search". In Proceedings of the 36th International ACM SIGIR conference on Research and development in information retrieval, 243–252, 2013.

[15] Reddy, E. K. "Nomenclature and Benchmarking Models of Text Classification Models: Contemporary Affirmation of

the Recent Literature", Global Journal of Computer Science and Technology, Vol. 14, Issue 9, 2015.

[16] Rossi, R. A., McDowell, L. K., Aha, D. W. & Neville, J. "Transforming Graph Data for Statistical Relational Learning", Journal of Artificial Intelligence Research, Vol. 45, 363–441, 2012.

[17] Singhal, A. & Srivastava, J. "Semantic Tagging for Documents Using "Short Text" information", International Journal of Computer Science & Information Technology, Vol. Issue 4, 2014.

[18] Spink, A. & Jansen, B. J. "Web search, public searching of the web", New York: Kluwer, 2004.

[19] Song, G., Ye, Y., Du, X., Huang, X. & Bie, S. "Short text classification: A survey", Journal of Multimedia, Vol. 9, Issue 5, 635–643, 2014.

[20] Vo, D-T. & Ock, C-Y. "Learning to classify short text from scientific documents using topic models with various types of knowledge, Expert Systems with Applications, Vol. 42, Issue 3, 1684–1698, 2015.

[21] Wang, B. K., Huang, Y. F., Yang, W. X. & Li, X. "Short text classification based on strong feature thesaurus", Journal of Zhejiang University SCIENCE C, Vol. 13, Issue 9, 649–659, 2012.

[22] Ward, E., Ikeda1, K., Erdmann, M., Nakazawa, M., Hattori, G. & Ono, C. "Automatic Query Expansion and Classification for Television Related Tweet Collection", Proceedings of Information Processing Society of Japan (IPSJ) SIG Technical Reports, Vol. 10, 1–8, 2012.

[23] Wiebe, N., Kapoor, A. & Svore, K. "Quantum nearest-neighbor algorithms for machine learning", arXiv preprint arXiv:1401.2142, 2014

[24] Zheng, D., Zhang, C., Fei, G. & Zhao, T. "Research on text categorization based on a weakly-supervised transfer learning method", In Computational Linguistics and Intelligent Text Processing, 144–156. Springer Berlin Heidelberg, 2012.

Initial Pattern Library Algorithm Based on Mean/Variance Classification for 3D SOM

Hongsong Li, Fulin Cheng
School of Information and Communication, Guilin University of Electronic Technology,
Guilin，Guangxi, China

Abstract - *Initial pattern library algorithm is an important part of three-dimensional self-organizing feature maps (3D SOM) algorithm. To overcome disadvantages of existing initial pattern library algorithm such as random algorithm and splitting algorithm, a new initial pattern library algorithm based on mean/variance classification for 3D SOM was proposed and applied to the image pattern recognition. Experimental results for image coding show that initial pattern library algorithm based on mean/variance classification has advantages of less invalid pattern vectors and high performance of pattern library. It is an effective way to improve the performance of 3D SOM.*

Keywords: image pattern recognition; image coding; three-dimensional self-organizing maps

1 Introduction

Kohonen's self-organizing maps (SOM) [1] is a very effective clustering method, and has been widely used in data mining and pattern recognition [4]-[8]. SOM algorithm is used to design an optimal pattern library by training a lot of samples.

Image coding based on pattern recognition is a new image coding method; the pattern library design is the key to it. Li et al. [9] propose an image coding scheme based on pattern recognition, the scheme use SOM algorithm to train pattern library, the initial pattern library generates by random algorithm, and experiment results show that the proposed coding scheme is better than JPEG2000.

In pattern library design, the initial pattern library plays an important part. Random algorithm and splitting algorithm are two common methods. Random algorithm is simpler, but has some disadvantages such as slower convergence speed, many invalid pattern vectors, etc. Splitting algorithm is more complex and needs a great deal of calculation. Li et al. [10] propose a sorting algorithm based on minimum distance between pattern libraries. In this paper, initial pattern library algorithm based on mean/variance classification for 3D SOM is proposed. The basic idea is that the similar pattern vectors in the initial pattern library are put together. Experimental results show that the proposed algorithm has advantages of less invalid pattern vectors and high matching degree with source.

This paper was supported by The Natural Science Foundation of China (No.61261035) and Innovation Project of GUET Graduate Education (No.GDYCSZ201451).

2 3D SOM algorithm

2.1 The principle of 3D SOM

SOM network consists of double-layer network structure, including the input layer and the mapping layer. The input layer receives input patterns, the mapping layer exports result, each input neuron connects with every mapping layer neurons through the connection weights, and mapping layer neurons connect to each other. Traditional SOM network generally adopts one-dimensional input layer and two-dimensional mapping layer; it can effectively deal with one-dimensional and two-dimensional signal. In recent years, the three-dimensional signal processing such as three-dimensional image and video increasingly attract attention, however, the traditional SOM algorithm can’t be used in it directly. Three-dimensional self-organizing feature maps (3D SOM) algorithm solves this problem, it can map two-dimensional input to three-dimensional output, achieve the three dimensional signal nonlinear mapping. The network structure of 3D SOM algorithm is shown in Fig.1. The mapping layer neurons are arranged in a three-dimensional structure, the number of rows, columns and layers can take different values. Obviously, different three-dimensional network structure usually result in different performance. The shape of three-dimensional neighborhood can have different choice, we usually choose spherical neighborhood, square neighborhood or orthogonal cross neighborhood, algorithm performance varies when selecting different three-dimensional shape of neighborhood. 3D SOM network is the same as ordinary competition network, for each input pattern, there is corresponding winning node in mapping layer, winning nodes represent the most similar pattern, the winning nodes and all nodes in its three-dimensional neighborhood adjust their own weight according to certain rules. When the input pattern changes, there will be a different pattern win through competition. In this way, the network adjusts the network weights through a large number of training samples by means of the self-organizing way. Finally the weight vector space is in accordance with the probability distribution of input patterns, namely the weight vector space can reflect the statistical characteristics of input pattern [5].

The neighborhood of 3D SOM algorithm is a three-dimensional structure; the number of nodes within the same neighborhood radius is more than two-dimensional plane structure, so the probability of nodes to be adjusted is increasing. Therefore, 3D SOM algorithm is used to design the pattern library of image pattern recognition can reduce the proportion of invalid mode vector, so as to improve the performance of the pattern library. SOM algorithm derived from the concept of biological neural networks, and the topology of biological neural network is a more complex three-dimensional structure, so use SOM neural network based on three-dimensional neighborhood to simulate the three-dimensional structure and function of biological neural network is more reasonable, and has achieved good effect in practical application.

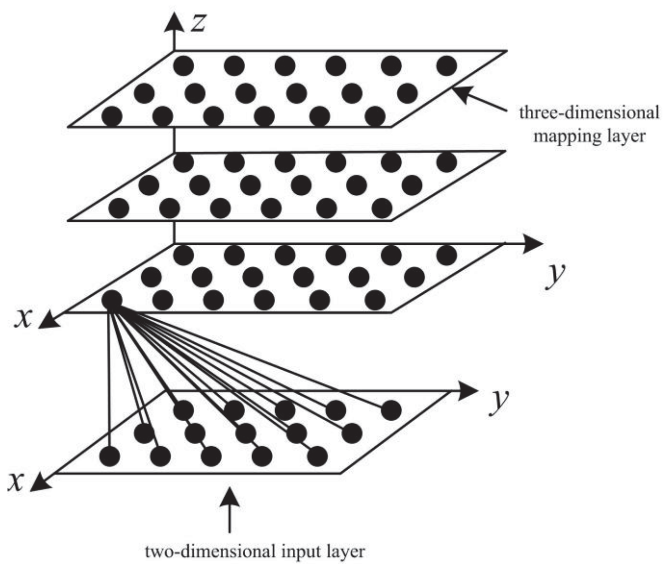


Fig.1 The network structure of 3D SOM algorithms

2.2 Steps of 3D SOM algorithm

Step-1: Given a neural size of network (N, M), and a training vector set $\{\mathbf{X}(t), t = 0,1,\cdots, L-1\}$, while N is the size of pattern library, M is the size of each pattern vector, and L is the number of training vectors. Initialize the pattern library $\{\mathbf{W}_j(0), j = 0,1,\cdots, N-1\}$ by initial pattern library algorithm based on mean/variance classification. Then the pattern vector are arranged in a three-dimensional structure $a \times b \times c$, where a, b and c represent the number of rows, columns and layers.

Step-2: Initialize the neighborhood $NE_j(0), j = 0,1,\cdots, N-1$.

Step-3: Input a new training vector $\mathbf{X} = (x_1, x_2, \cdots, x_M)^{\mathrm{T}}$.

Step-4: Compute the distortion d_j between the input vector and each pattern vector in the pattern library with some distortion measure, and select j^* as the winning pattern vector.

Step-5: Modify the winning pattern vector j^* and its neighboring pattern vectors by

$$\mathbf{W}_j(t+1) = \begin{cases} \mathbf{W}_j(t) + \alpha(t)[\mathbf{X}(t) - \mathbf{W}_j(t)] & j \in j*, NE_{j^*}(t) \\ \mathbf{W}_j(t) & else \end{cases}$$

Where, $NE_{j^*}(t)$ is Euclidean distance neighborhood around the winning pattern vector which is decreased with t, $NE_{j^*}(t) = A_0 + A_1 \mathrm{e}^{-t/T_1}$, A_0 and A_1 are constants determining the range of the neighborhood, and T_1 is a constant determining the decreasing rate.

The learning rate $\alpha(t)$ determines the modification amount of pattern vectors. Theoretically, if $\alpha(t)$ is small enough, the average error function of the system will reach the minimum after a long time training. In practice, learning rate is usually determined by $\alpha(t) = A_2 \mathrm{e}^{-t/T_2}$. Where, A_2 is a constant determining the maximum of learning rate, and T_2 is a constant determining the decreasing rate.

Step-6: Go to Step-3.

3 The original initial pattern library algorithm

3.1 Random Algorithm

Random algorithm can be classified to random data setting and random sampling. In random data setting algorithm, the initial pattern vectors are set to random data. Because the initial pattern library is independent with the training set, and some invalid pattern vector often exist in the pattern library after training, thus random data setting algorithm is not usually used.

In random sampling algorithm, the initial pattern vectors are elected randomly from the training set. For example, if the training set is denoted by $\{\mathbf{X}(t), t = 0,1,\cdots, L-1\}$, the initial pattern library can be obtained by selecting vectors $\mathbf{X}(0), \mathbf{X}(p), \mathbf{X}(2p), \cdots \mathbf{X}((N-1)p))$, $p = L/N$. This algorithm has two advantages: it doesn't need any calculation; there will be few invalid pattern vectors in the pattern library. However, sometimes many similar vectors are selected, and the pattern library performance will degrade.

3.2 Splitting Algorithm

The splitting algorithm proposed by Linde, Buzo and Gray can be implemented as follows:

Step-1: Compute the centroid (represented by $\mathbf{W}(0)$) of all training vectors, split $\mathbf{W}(0)$ to two close vector $\mathbf{W}_1(0) = \mathbf{W}(0) + \mathrm{e}$ and $\mathbf{W}_2(0) = \mathbf{W}(0) + \mathrm{e}$, where e is a fix perturbation vector.

Step-2: Use $\{\mathbf{W}_1(0), \ \mathbf{W}_2(0)\}$ as the initial pattern library, design the pattern library with two pattern vectors by

LBG algorithm. The resulting pattern library is represented by $\{\mathbf{W}_1(1),\ \mathbf{W}_2(1)\}$.

Step-3: Split $\{\mathbf{W}_1(1),\ \mathbf{W}_2(1)\}$ to four pattern vectors as Step-1.

Step-4: Design the pattern library with four pattern vectors as step- 2, and the resulting pattern vectors are split to eight pattern vectors. Repeat this procedure until N initial pattern vectors are created.

The defect of splitting algorithm is too complex, and it can not adapt statistic characteristic of information sources.

3.3 The Average Separation Algorithm

In average separation algorithm, the training vector set is divided into segments, each segment length is $p=(L/N)$ where L is training vector number and N is the pattern library size. Take average in each section will get pattern vector. The initial pattern library can be obtained by

$$\mathbf{W}_j(0)=\frac{\mathbf{X}(jp+1)+\mathbf{X}(jp+2)+\cdots+\mathbf{X}(jp+p)}{p} \qquad j=0,1,\cdots,N-1\cdot$$

In practice, the average separation algorithm is an improved method of random algorithm, the performance has improved, but its enhancement is definite.

4 Mean/variance classification algorithm

In 3D SOM algorithm, neighborhood pattern vectors will influence each other. The basic idea of mean/variance classification algorithm is to put the similar pattern vectors together in the initial pattern library which can reduce the bad influence between neighborhood pattern vectors. Training vector of similar average is likely to constitute a similar pattern. When the average is similar, training vector of similar variance is more likely to constitute a similar pattern. So, we can obtained pattern library based on mean/variance classification algorithm. Specific steps are as follows:

Step-1: Compute the mean value of each vector in the training set $\{\mathbf{X}(t), t=0,1,\cdots,L-1\}$.

Step-2: Sort the training vectors by mean value from small to large order, and divide the adjusted training vectors into four parts: $\{\mathbf{X}_1(t)\}$、$\{\mathbf{X}_2(t)\}$、$\{\mathbf{X}_3(t)\}$、$\{\mathbf{X}_4(t)\}$.

Step-3: Sort the training vectors in each part by variance.

Step-4: Choose $N/4$ training vectors in each part at the same intervals, put them together to form initial pattern library which contain N pattern vectors.

5 Experimental results

We use standard testing grayscale images ('Lena') to test mean/variance classification algorithm. In our experiments, distortion measure is calculated by square error criterion $d_j(t)=\|\mathbf{X}(t)-\mathbf{W}_j(t)\|^2$, and the size of image block is $M=8\times 8$. The reconstructed image quality is measured by PSNR, where $R_{\mathrm{PSNR}}=10\lg\frac{255^2}{E_{\mathrm{MSE}}}\mathrm{dB}$, and MSE is the mean square error between the original image and the reconstructed image. Ratio of image compression is calculated by $C_{\mathrm{R}}=\frac{M\times B_{\mathrm{O}}}{B_{\mathrm{C}}}$, while M is the dimension of the pattern vector, B_{O} is bits of the original image pixels, B_{C} is the bit of pattern vector class indexes. All experimental data is obtained after a large number of experiments and constantly adjusting the experimental parameters.

In the experiment, we use random sampling algorithm, average separation algorithm and mean/variance classification algorithm to design initial pattern library of 3D SOM algorithm, then compare their performance by the reconstructed image quality. The pattern library size is $N=40960$. When the pattern library size varies, the PSNR of the reconstructed image of the three kinds of initial pattern library algorithm is shown in Table I. It is clear that when the pattern library size is small, the PSNR of the reconstructed images is similar, but when the pattern library size is big, the PSNR of the reconstructed image based on mean/variance classification algorithm is obviously higher. When the pattern library size is 1024 (compression ratio $C_{\mathrm{R}}=\frac{64\times 8}{10}=51.2$), compare with random sampling algorithm and average separation algorithm, the PSNR of the reconstructed image based on mean/variance classification algorithm increase 0.26dB and 0.21dB. When the pattern library size is 2048 (compression ratio $C_{\mathrm{R}}=\frac{64\times 8}{11}=46.5$), the PSNR of the reconstructed image increases 0.51dB and 0.55dB.

Table 1
The PSNR of the reconstructed image

pattern library size	random sampling algorithm (dB)	average separation algorithm (dB)	mean/variance classification algorithm (dB)
2048	36.068 5	36.034 8	36.583 7
1024	33.152 6	33.195 8	33.411 9
512	30.859 9	30.925 9	31.008 7
256	28.965 8	28.876 3	29.071 2
128	27.506 3	27.440 5	27.576 7

Fig.2 illustrates the comparisons of the reconstructed image between initial pattern library based on random sampling algorithm, average separation algorithm and mean/variance classification algorithm. Fig.2 (a) show the original image, Fig.2 (b) show the reconstructed image when initial pattern library algorithm is random sampling algorithm, Fig.2 (c) show the reconstructed image when initial pattern library algorithm is average separation algorithm, Fig.2 (d) show the reconstructed image when initial pattern library algorithm is mean/variance classification algorithm. Obviously, the reconstructed images using mean/variance classification algorithm have better subjective equality.

Fig.2 Original image and reconstructed images; (a) Original image (‘Lena’); (b) reconstructed image by random sampling algorithm; (c) reconstructed image by average separation algorithm; (d) reconstructed image by mean/variance classification algorithm.

6 Conclusion

In this paper, a new initial pattern library algorithm based on mean/variance classification for 3D SOM algorithm is proposed. Experimental results show that initial pattern library algorithm based on mean/variance classification has advantages of less invalid pattern vectors and high matching degree with source. The next work is applying 3D SOM based on the proposed initial pattern library algorithm to three-dimensional video coding and doing further improvement to get better performance.

7 References

[1] T. Kohonen, "An introduction to neural computing," Journal of Neural Networks, vol. 2, no. 7, pp. 3-13, Jan 1988.

[2] Ortiz A, Gorriz J M, Ramirez J, et al, “Improving MR brain image segmentation using self-organising maps and entropy-gradient clustering,” Information Sciences, vol. 262, pp. 117-136, Mar 2014.

[3] Yuanfeng L, Falin W, “Three-dimensional probabilistic neural network using for MR image segmentation,” Electronic Measurement & Instruments (ICEMI), 2011 10th International Conference on. IEEE, vol. 3, pp. 127-131, Aug 2011.

[4] Parisi G I, Wermter S, “Hierarchical SOM-based detection of novel behavior for 3D human tracking,” Neural Networks (IJCNN), The 2013 International Joint Conference on. IEEE, pp. 1-8, Aug 2013.

[5] T. Kohonen, “Essentials of the self-organizing map.” Neural Networks, vol. 37, pp. 52-65, Jan 2013.

[6] Wijayasekara D, Linda O, Manic M, “CAVE-SOM: Immersive visual data mining using 3D Self-Organizing Maps,” Neural Networks (IJCNN), The 2011 International Joint Conference on. IEEE, pp. 2471-2478, Aug 2011.

[7] Schmidt A,Hanson C, Kathilankal J,et al, “Classification and assessment of turbulent fluxes above ecosystems in North-America with self-organizing feature map networks,” Agricultural and Forest Meteorology, vol. 151, no.4, pp. 508-520, Apr 2011.

[8] Wirth H, Löffler M, von Bergen M, et al., “Expression cartography of human tissues using self organizing maps,” BMC bioinformatics, vol. 12, no. 1, pp. 306, Jul 2011.

[9] Li H, Li Da, “A new image coding scheme based upon image pattern recognition,” Intelligent Computing and Intelligent Systems (ICIS), 2010 IEEE International Conference on. IEEE, vol. 1, pp. 172-174, Oct 2010.

[10] Li Hongsong, Xu Baohua, “A New Initial Codebook Algorithm for Learning Vector Quantization,” Communications, 2007. ICC’07, IEEE International Conference on. IEEE, pp. 2610-2613, Jun 2007.

Classification Using Jumping Emerging Patterns and Cosine Similarity

Mauri Ferrandin[1], **Adão Boava**[1], **Alex Sandro Roschildt Pinto**[1]
{mauri.ferrandin, adao.boava, a.r.pinto}@ufsc.br
[1]Federal University of Santa Catarina - UFSC
Blumenau, Santa Catarina, Brazil

Abstract— *Classification is a common task in Machine Learning and Data Mining. Jumping Emerging Patterns have been applied for classification in different contexts with good results and the advantage of to be easily understandable for users. In this work we propose the use of cosine similarity measure to select the patterns which will be used to predict the classes in the classification process. Two versions of the algorithm were proposed and tested with four different parameter values in 21 datasets. The results were compared with three frequently used classification algorithms from the literature and proposed algorithms showed a promising results achieving in some the datasets best results than C4.5 algorithm.*

Keywords: machine learning, classification, jumping emerging pattern.

1. Introduction

Classification is one of the most important problems in Machine Learning and Data Mining. The classification consists in associating one or more classes from a set of predefined classes to a not classified example (instance) from a database. The features (attributes) of each example will determine the classes it will be associated to.

Jumping Emerging Patterns (JEPs) have been largely applied for classification with the advantage of to be easy understandable by the user. The lack of comprehensibility is an important drawback that may impede the use of some classification models in some domains (e.g. medical diagnosis, credit approval).

Most researches related to classification through the use of JEPs were proposed using a covering strategy to select the the of JEPs to classify an instance. Broadly speaking, in this work we propose the use of cosine similarity measure for the selection of the JEPs in the classification algorithm allowing not only the JEPs which are a subset of the instance influence in the class prediction task. The proposed algorithms were tested on 21 datasets and their results were compared against three other frequently used classifiers from the literature.

The remainder of this paper is organized as follows: Section 2 presents background on JEPs foundations and the related works in the subjects of this paper. Section 3 discusses the new proposed algorithms for classification. Section 4 presents the experimental setup and reports the computational results obtained with the algorithms proposed in this paper. Conclusions and some perspectives about future works are stated in Section 5.

2. Emerging Patterns

A pattern is an expression defined in a language which describes a collection of objects. In order to represent patterns, combinations of feature values are used, like (Color = green, Sex = male, Age = 23) or as logical properties, like [Color = green] $\wedge$ [Sex = male] $\wedge$ [Age $>$ 23].

Is said that the pattern P covers the object x, or the object x supports the pattern P, if the object fulfils the property expressed by the pattern. A useful characteristic of a pattern P is the amount of objects from a collection X that supports P, which is called the pattern support and it is denoted by $sup(P, X)$ [1].

In a supervised classification problem, we say that a pattern is discriminative if it includes properties which help to differentiate if a given instance belongs to one class or other. In general, many comprehensible classifiers use discriminative patterns and support information for classification.

In [2] the authors introduced the p-emerging pattern for two class problems, which is an emerging pattern (EP) with Growth Rate $GR \geq p$. The GR measures how frequent a pattern is in its own class C_P with respect to its frequency in the other class C.

$$GR(P) = \begin{cases} 0, & \text{if } sup(P,C) = 0 \wedge sup(P,C_p) = 0 \\ \infty, & \text{if } sup(P,C) = 0 \wedge sup(P,C_p) > 0 \\ \frac{sup(P,C_p)}{sup(P,C)}, & \text{otherwise} \end{cases} \tag{1}$$

An important class of emerging patterns are those that cover objects in a single class, which are named JEPs. JEPs are emerging patterns with $GR = \infty$ and they have been widely used in emerging pattern classifiers because they have a strong predictive power. Jumping emerging patterns represent properties that are only present in a single class, so they should be distinctive [1].

A JEP can also be maximal. A JEP is a maximal emerging pattern whose supersets are not emerging patterns.

The process of finding emerging patterns from a dataset is a challenging procedure because the number of candidates grows exponentially with respect to the number of attributes. The algorithms to extract the set of JEPs from a dataset are not focus of this work and more about the subject can be found in [1] and [2].

In the survey of [1], the authors say that there no exists general theory that can help us to select the appropriate patterns and classifier for a given problem. Actually, the only solution for users is to test all the available classifiers over some data samples, selecting the one with the highest accuracy.

Emerging patterns theory have been largely used for flat classifications and applied to a variety of domains with different techniques and configuration. More about the subject can be found in [1] and [2].

2.1 Classification using JEPs

In a supervised learning, classification is to predict the classes for a given query instance q. A pattern based classifier assigns to the instance q the class with the highest score. The score is computed based on the patterns contained in q and usually the scoring function is designed taking into account the characteristics of emerging pattern used. According to [1] emerging pattern classifiers can be grouped into categories, according to the type of scoring they use. The author divided them into groups:

- Aggregation support classifiers: the scoring function is defined based on the support of the JEPS. The most important classifier in this group is the CAEP [2].
- Information based classifiers: is a variant of aggregation support in which the scoring function is defined using a minimum encoding inference approach instead of the aggregation of support. The most representative classifier is named iCAEP [3];
- Bayesian classifiers: is a hybrid classifier using JEPs and Naive Bayes. It uses essential Jumping Emerging Patterns to relax the strong feature independence assumption. One sample is the Bayesian Classification base on Emerging Patterns (BCEP) [4]. BCEP uses the patterns extracted in the training stage to derive a product approximation for each class probability. To obtain such approximation, the matching patterns are combined using the chain rule of probability;
- Graph classifiers: uses a graph representation of the JEPs to consider the hierarchical relations between them. [5] is the main classifier in this group;
- Combined classifiers: uses the combination of two or more groups and other techniques. The most remarkable works in this group are: [6] combined a pattern based classifier in cascade with a k-NN classifier; A bagging and boosting method presented in [7]; and [8] in which the authors introduced an algorithm for building cascades of emerging pattern based classifiers.

The two most commonly used schemes are using minimal patterns with aggregation of support classifiers and maximal patterns with Bayesian classifiers [1].

3. Proposed Algorithm

Considering the use JEP based techniques in the domain of classification explained in the Section 2, two new classification models named JEP Classifier with Cosine Similarity (JEPCS) and JEP Classifier with Cosine Similarity Aggregating Support (JEPCSas) are proposed. The JEPCS is shown in Algorithm 1 and the JEPCSas is detailed in the end of this Section.

Algorithm 1 JEPCS

Require: The sets of instances for: training TR, testing TE; A set of classes C, a minimum support threshold parameter S and a parameter k;

1: **foreach** ($c_i \in C$) **do**
2: Create a subset trc_i containing the instances of TR labelled with c_i;
3: Compute a set $jtrc_i$ of JEPs with $sup(JEP, c_i) >= S$ in the subset trc_i;
4: Add $jtrc_i$ to a set JTR;
5: **end for**
6: **foreach** ($jtrc_i \in JTR$) **do**
7: Clean the JEPs in the set $jtrc_i$ removing the redundant JEPs among the classes;
8: **end for**
9: **foreach** ($te_i \in TE$) **do**
10: Represent te_i attributes as vector tev_i;
11: Find the a set $JTRS \supseteq JTR$ containing k JEPs more similar to tev_i using cosine similarity measure $cos(te_i, JEP)$;
12: Apply the score function $score(te_i, C)$ for each class C using JEPs from the set $JTRS$ to obtain the class CS with highest score;
13: Predict the select class CS to the instance te_i;
14: **end for**
15: Compute the results of classification process;

The proposed algorithm starts splitting the training instances to generate a subset of instances for each class in the dataset (line 2). In the sequence, the JEPs are computed for each subset created (line 3). The core implementation used to find the maximal JEPs was the FP-Growth that is presented in [9].

The next step represented in the line 7 aims to eliminate JEPs associated with more than one class. If a JEP is associated with more than one class, only the class in which the JEP has bigger intra-class support will be kept.

After the creation of classification model, all classes are represented by a set of JEPs that discriminate them from each other, the next phase is use the model to classify unseen instances. The main step of this phase is represented in the line 11 where occurs the selection of the JEP most similar to the instance being classified. The similarity between an instance te_i and a JEP is obtained through the cosine similarity as represented in Equation 2. If more than one JEP were select with the highest similarity, the one with biggest intra-class support is chosen as more important.

$$cos(te_i, JEP) = \frac{te_i \cdot JEP}{||te_i|| \; ||JEP||} \tag{2}$$

The set of k JEPS more similar to the instance te_i with their respective classes are used as input to a score function to determine which class the instance te_i will be associated to.

The score function consists in to obtain a score for each candidate class. The class with biggest score is chosen. Given a test instance te_i and a set $JTRS(C)$ with the select JEPs that discriminate the class C, the score function for the algorithm JEPCS is represented in the Equation 3.

$$score(te_i, C) = |JTRS(C)| \tag{3}$$

A different score function is used for JEPCSas aiming to take into account the support of the selected JEPs by aggregating their sum on each class. The Equation 4 show the score function for the JEPCSas algorithm.

$$score(te_i, C) = \sum_{e \in JTRS(C)} sup(e) \tag{4}$$

Finally the classifier hits and misses are computed (line 15) through the measures of classification accuracy.

4. Experimental Evaluation

In this Section the experimental evaluation performed with the proposed algorithm is presented. Also, the directly and statistical comparisons of results against other algorithms from the literature is demonstrated.

4.1 Datasets

In order to analyse the proposed classifiers, 21 datasets were chosen randomly from the UCI Machine Learning Repository [10]. The main objective was to test the proposed classifier with a wide range of datasets with different features and number of classes. Considering the chosen datasets 2/3 of them have two classes (binary classification problem) and 1/3 of them have more than two classes (multiclass classification problem). Table 1 summarizes the main information about the datasets, number of instances, attributes and classes.

Table 1: Datasets main characteristics.

Dataset	Instances	Attributes	Classes
anneal	798	38	6
australian	690	14	2
breast-cancer	286	9	2
breast-w	699	9	2
diabetes	768	8	2
heart	270	13	2
hepatitis	55	19	2
ionosphere	351	34	2
labor	57	16	2
liver-disorders	345	6	2
mushroom	8124	22	2
nursery	12960	8	5
shuttle	43500	9	9
sick	3772	30	2
sonar	209	60	2
soybean	683	35	19
tic-tac-toe	958	9	2
vehicle	946	18	4
vote	435	16	2
wine	179	13	3
zoo	101	18	7

Before performing the experiments, the following preprocessing steps were applied to the datasets using the Weka environment [11] : (i) All attributes with missing values were replaced using with the modes and means from the training data (ReplaceMissingValues filter); (ii) All continuous attributes were converted to nominal attributes using a discretization algorithm (Discretize filter) with 10 bins; (iii) All discretized attributes were converted to binary attributes with value 0 or 1 indicating the absence or presence of the attribute in the instance (NominalToBynary filter). Last three steps were done in a unsupervised way (i.e. without taking into account the the class information of the instances).

4.2 Experiments Obtained Results

The experiments with the two proposed algorithms were conducted as represented below:

- JEPCS: four experiments with minimum support $S = 1\%$ in the JEPS selection and different values for parameter $k = \{1, 3, 5, 15\}$ hereafter named as sk1, sk3, sk5 and sk15 repectively;
- JEPCSas: four experiments with minimum support $S = 1\%$ in the JEPS selection and different values for parameter $k = \{1, 3, 5, 15\}$ hereafter named as sak1, sak3, sak5 and sak15 repectively;

In order to compare the results of the JPECS and JEPCSas with classifiers from the literature, the algorithms k-NN with k = 3 (here after named 3-NN) [12], Naive Bayes (here after named NB) [13] and C4.5 [14] were chosen as baseline.

The All the experiments were performed using 10-fold cross validation method.

Table 2 shows the obtained results with the proposed approaches comparing their results with the algorithms from the literature. The measure use was the classification accuracy and the best result for each dataset is presented in bold.

Table 2: Obtained results compared with 3-NN, NB and C4.5 algorithms.

Dataset	JEPCS				JEPCSas				3-NN	NB	C45
	sk1	sk3	sk5	sk15	sak1	sak3	sak5	sak15			
anneal	0.953	0.975	0.977	0.954	0.953	0.975	0.977	0.958	0.980	0.947	**0.987**
australian	0.755	0.783	0.791	0.803	0.755	0.783	0.791	0.803	0.823	0.832	**0.836**
breast-cancer	0.950	0.964	0.966	0.963	0.950	0.964	0.966	0.963	0.956	**0.974**	0.946
breast-w	0.699	0.699	0.699	0.717	0.699	0.699	0.699	**0.720**	0.713	0.706	0.703
diabetes	0.673	0.681	0.699	0.692	0.673	0.681	0.699	0.694	0.697	**0.757**	0.727
heart	0.756	0.785	0.800	0.804	0.756	0.785	0.800	0.804	0.789	**0.841**	0.767
hepatitis	0.820	0.833	0.815	0.795	0.820	0.833	0.815	0.795	0.839	**0.858**	0.781
ionosphere	0.912	0.909	0.906	0.906	**0.937**	0.919	0.907	0.904	0.906	0.906	0.872
labor	0.840	0.840	0.803	0.720	0.840	0.840	0.803	0.720	0.807	**0.930**	0.825
liver-disorders	0.618	0.471	0.500	0.559	0.618	0.471	0.500	0.588	**0.649**	0.620	0.626
mushroom	**1.000**	0.996	0.995	0.992	**1.000**	0.996	0.995	0.992	**1.000**	0.929	**1.000**
nursery	0.431	0.435	0.478	0.705	0.431	0.438	0.483	0.703	0.983	0.903	**0.994**
shuttle	0.918	0.880	0.823	0.866	0.918	0.881	0.853	0.874	0.941	0.851	**0.942**
sick	0.957	0.955	0.958	0.963	0.957	0.955	0.958	0.963	0.978	0.964	**0.979**
sonar	0.783	0.783	0.745	0.769	0.783	0.783	0.767	0.809	**0.827**	0.750	0.697
soybean	0.911	0.914	0.911	0.889	0.911	0.915	0.915	0.899	0.892	**0.936**	0.931
tic-tac-toe	0.845	0.861	0.871	0.889	0.845	0.855	0.866	0.888	**0.990**	0.689	0.948
vehicle	0.695	0.688	0.681	0.680	0.696	0.689	0.682	**0.717**	0.708	0.602	0.694
vote	0.706	0.703	0.696	0.651	0.706	0.703	0.712	0.651	0.931	0.913	**0.963**
wine	0.894	0.905	0.911	0.883	0.894	0.905	0.916	0.894	0.916	**0.961**	0.910
zoo	0.950	0.930	0.910	0.781	0.950	0.950	0.920	0.811	0.941	**0.960**	0.901

Comparing results presented in Table 2 the eight different configurations of JPECS and JEPCSas obtained a best or equal results only in four datasets. Considering the direct comparison of the accuracy Table 3 presents the ranking of the algorithms. In this ranking the results with different k parameter values were merged taking only the best result for JEPCS and JEPCSas. This sum of rankings shows a good performance of JEPCSas when taken in to account not just the first position in the rank as showed. Stratifying the results considering the number of classes is possible to observe a little advantage of 3-NN on binary datasets, a little advantage of JEPSw on multiclass datasets and equal result of JEPCSas and 3-NN both tied in the first position considering overall ranking.

Statistical tests based on Friedman comparing the accuracy values achieved by the classifiers with ($\alpha = 0.05$) indicated significant differences between the classifiers. The matrix with post-hoc analysis represented in Table 4. The statistical analysis shows:

- on datasets with $|class| = 2$: a clear advantage of the 3-NN classifier, a less accentuate advantage of the NB algorithm and no statistical difference between C4.5 and the proposed classifiers;
- on datasets with $|class| > 2$: a little advantage to 3-NN and C4.5 and a very little advantage in favor of NB classifier. Also, in this group of datasets the test shows a statistical advantage of the JEPCSas with $k = \{3, 5\}$ when compared to JEPCS with $k = \{15\}$. Moreover, the test showed no statistical difference between JEPCSas with $k = \{3, 5\}$ when compared to

Table 3: Best JEPCS and JEPCSas results compared with 3-NN, NB and C4.5 algorithms considering the ranking of the five compared approaches. The number of * indicates the ranking position of each classifier.

Dataset	JEPCS	JEPCSas	3-NN	NB	C4.5
anneal	***	***	**	*****	*
australian	****	****	***	**	*
breast-cancer	**	**	****	*	*****
breast-w	**	*	***	****	*****
diabetes	**	**	****	*	*****
heart	**	**	****	*	*****
hepatitis	***	***	**	*	*****
ionosphere	**	*	***	***	*****
labor	***	***	*****	*	****
liver-disorders	*****	****	*	***	**
mushroom	*	*	*	*****	*
nursery	*****	****	**	***	*
shuttle	***	***	**	*****	*
sick	****	****	**	***	*
sonar	****	**	*	****	*****
soybean	****	***	*****	*	**
tic-tac-toe	***	****	*	*****	**
vehicle	***	*	**	*****	****
vote	*****	****	**	***	*
wine	****	**	**	*	*****
zoo	**	**	****	*	*****
|class| = 2	42	37	**36**	37	47
|class| > 2	24	**18**	19	21	19
All	66	**55**	**55**	58	66

Table 4: Results of Friedman test considering all results (blank means no significance, and the crescent scale .,*,**,*** indicate significance in favor of the classifier represented in the column)

Dataset Group	Algorithm	JEPCS				JEPCSas			
		sk1	sk3	sk5	sk15	sak1	sak3	sak5	sak15
|class| = 2	3-NN	*	*	**	**	.	*	*	*
	NB	.	*	*	*		.	.	*
	C4.5								
|class| > 2	3-NN	.		*	**				.
	NB								*
	C4.5	.	.	*	**	.			*
	sak3				*				
	sak5				*				
All	3-NN	**	**	***	***	**	**	*	**
	NB	*	*	*	**	*	.		.
	C4.5	*	*	*	**	.			.

the three baseline approaches in this group of datasets;

- on all datasets: a clear advantage of 3-NN classifier against all versions of JEPCS and JEPCSas followed by a less accentuate advantage of the NB classifier with no advantage when compared against the JEPCSas with $k = \{5\}$, and no statistical advantage of C4.5 when compared to JEPCSas with $k = \{1, 3, 5\}$.

5. Conclusion

This paper has presented a new algorithm for classification problems. The algorithm is based in jumping emerging patterns concepts combined with the cosine similarity measure and were applied in 21 largely used datasets with different number of target classes. The results were compared with three algorithms - 3-NN, NB and C4.5 - largely adopted in the classification literature.

Classifiers based in JEPs have as the main advantage of to be easily understandable making them useful to applications on classification systems in which this feature is required. This kind of classifiers are also useful for real-world classification problems because they deal with noise extremely well as demonstrated in [7]. On the other hand, this kind of classifiers can suffer with the exponential behaviour of the datasets in the task of mining all JEPs due to the great number of possible candidates. This drawback can turn very hard the adoption of JEPs based approach in some domains.

The analysis of results showed the proposed algorithm was very competitive and achieved remarkable results taking into account that classification is a complex task and the approaches chosen as baseline for comparison are very studied and stable approaches. Even though the proposed did not achieved best accuracy results in most of the datasets, it showed to be statistical equivalent to the three baseline algorithms in one group of datasets (datasets with $|class| > 2$ compared to JEPCSas with $k = \{3, 5\}$). Moreover, is noteworthy that the statistical test showed no statistical difference when comparing all configuration of proposed algorithms against C4.5 algorithm in datasets with $|class| = 2$.

As future research we highlight a deeper analysis of use the combination of JEPs and cosine similarity function testing it's behavior with different parameters of support and extending the application to a largest number of datasets. Also, avail the use of techniques that allow to avoid the discretization process and allow the use of continuous attributes with the cosine similarity function is a clue to improve the results. As last, the proposed approach needs to be compared to other JEP based approaches presented in the Section 2 of this paper.

References

[1] M. García-Borroto, J. Martinez Trinidad, and J. A. Carrasco-Ochoa, "A survey of emerging patterns for supervised classification," *Artificial Intelligence Review*, vol. 42, no. 4, pp. 705–721, 2014. [Online]. Available: http://dx.doi.org/10.1007/s10462-012-9355-x

[2] G. Dong, X. Zhang, L. Wong, and J. Li, "CAEP: Classification by Aggregating Emerging Patterns," in *Proceedings of the 2nd international conference on discovery science (DS '99)*, 1999, p. 42. [Online]. Available: http://dx.doi.org/10.1007/3-540-46846-3_4

[3] X. Zhang, G. Dong, and K. Ramamohanarao, "Information-based classification by aggregating emerging patterns." in *IDEAL*, ser. Lecture Notes in Computer Science, K.-S. Leung, L.-W. Chan, and H. Meng, Eds., vol. 1983. Springer, 2000, pp. 48–53.

[4] Z. Wang, H. Fan, and K. Ramamohanarao, "Exploiting maximal emerging patterns for classification." in *Australian Conference on Artificial Intelligence*, ser. Lecture Notes in Computer Science, G. I. Webb and X. Yu, Eds., vol. 3339. Springer, 2004, pp. 1062–1068.

[5] M. García-Borroto, J. F. M. Trinidad, and J. A. Carrasco-Ochoa, "Fuzzy emerging patterns for classifying hard domains." *Knowl. Inf. Syst.*, vol. 28, no. 2, pp. 473–489, 2011.

[6] J. Li, K. Ramamohanarao, and G. Dong, "Combining the strength of pattern frequency and distance for classification." in *PAKDD*, ser. Lecture Notes in Computer Science, D. W.-L. Cheung, G. J. Williams, and Q. Li, Eds., vol. 2035. Springer, 2001, pp. 455–466.

[7] H. Fan and K. Ramamohanarao, "Fast discovery and the generalization of strong jumping emerging patterns for building compact and accurate classifiers." *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 6, pp. 721–737, 2006.

[8] M. García-Borroto, J. F. M. Trinidad, and J. A. Carrasco-Ochoa, "Cascading an emerging pattern based classifier." in *MCPR*, ser.

Lecture Notes in Computer Science, J. F. M. Trinidad, J. A. Carrasco-Ochoa, and J. Kittler, Eds., vol. 6256. Springer, 2010, pp. 240–249.

[9] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *SIGMOD Rec.*, vol. 29, no. 2, pp. 1–12, May 2000. [Online]. Available: http://doi.acm.org/10.1145/335191.335372

[10] C. Blake and C. Merz, "UCI repository of machine learning data sets," http://www.ics.uci.edu/∼mlearn/MLRepository.html, 1999.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009. [Online]. Available: http://doi.acm.org/10.1145/1656274.1656278

[12] D. Aha, D. Kibler, and M. Albert, "Instance-based learning algorithms," *Machine Learning*, vol. 6, no. 1, pp. 37–66, 1991. [Online]. Available: http://dx.doi.org/10.1007/BF00153759

[13] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Eleventh Conference on Uncertainty in Artificial Intelligence*. San Mateo: Morgan Kaufmann, 1995, pp. 338–345.

[14] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

AN EMPIRICAL EVALUATION OF ADABOOST IN NEAT AND rtNEAT

Robert Schukei[1], and Dr. Blayne E. Mayfield[2]
[1]Computer Science Department, Baker University, PO BOX 65, Baldwin City, KS, 66006
[2]Computer Science Department, Oklahoma State University, 219 MSCS, Stillwater, OK, 74078

Abstract - *While the level of spam (unsolicited, unwanted emails) has dropped over the past couple of years, it still accounts for more than 60% of email traffic.[1] Not only is this an inconvenience to email users, but in areas of the world with limited Internet bandwidth, spam can choke much of that capacity. The research described in this paper attempts to decrease the runtime of spam filter training by employing the machine learning techniques of Adaptive Boosting (AdaBoost) in conjunction with NEAT (NeuroEvolution of Augmenting Technologies) or rtNEAT (real-time NEAT).*

Keywords: NEAT, rtNEAT, AdaBoost, Neuroevolution, Spam Filtering

1 Introduction

While the level of spam (unsolicited, unwanted emails) has dropped over the past couple of years, it still accounts for more than 60% of email traffic.[1] Not only is this an inconvenience to email users, but in areas of the world with limited Internet bandwidth, spam can choke much of that capacity. Further, spam often is used as a vehicle for attempting to introduce malware onto computers that receive it.

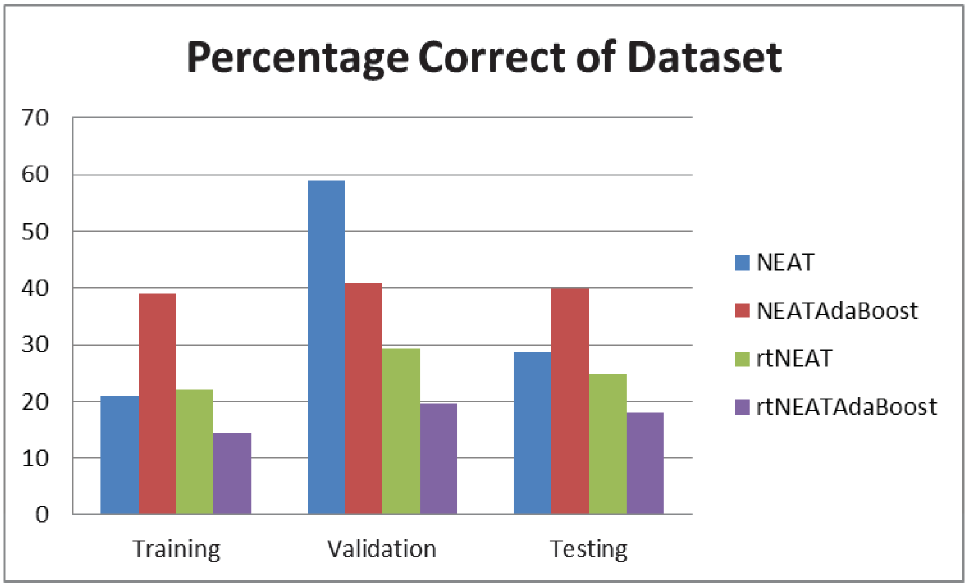
Most Internet Service Providers (ISPs), as well as email server and client software, deploy spam filters in the battle to prevent spam from getting to the end user. These spam filters require time-consuming training and retraining to do their jobs. The research described in this paper attempts to decrease the runtime of spam filter training by employing the machine learning techniques of Adaptive Boosting (AdaBoost) in conjunction with NEAT (NeuroEvolution of Augmenting Technologies) or rtNEAT (real-time NEAT). While it is understood that spam filters utilizing these techniques might not be as effective as current filters, the experiment will indicate which technique might be the best candidate for further investigation. The training will be using the UCI SPAM dataset.[2]

2 Background

NEAT is a neuroevolution technique that constructs and trains Artificial Neural Networks (ANNs) over a data set. It was developed by the Neural Networks Research Group at the University of Texas, and they state that, “neuroevolution is a method for optimizing neural network weights and topologies using evolutionary computation.”[3] NEAT is an algorithm that evolves a neural network instead of having human specialists spend their time trying to work out a network topology that can solve a specific type of problem. In each generation of its run, NEAT examines and can change the weights and topologies of a network to learn better how the inputs and outputs are related to each other.[4, 5, 6] Unlike many evolutionary algorithms, NEAT uses speciation (evolution of species) to guarantee that favorable attributes in the population will persist for some number of generations, thus giving the species that exhibit those attributes a greater chance of passing them along to organisms in future generations.[4, 5, 6]

One variation of NEAT that has proven to be more effective in some cases is rtNEAT. Unlike NEAT, rtNEAT does not evolve the entire population in a single generation.[7] The result is that only small changes occur in each generation, decreasing the generation-to-generation runtime. Thus, rtNEAT is a better choice than NEAT in applications where evolution must take place in real-time. The algorithm tracks the age of each organisms and, when an organism reaches some specific age, its fitness is tested. If that fitness is among the worst in the population, then the organism dies; two, more fit organisms from the fittest species then mate to produce a replacement organism.

While NEAT and rtNeat is about evolving artificial neural networks an ensemble technique that has been used to improve performance of ANN is Adaptive Boosting. AdaBoost is a common ensemble methodology for machine learning; i.e., it is a secondary technique (sometimes referred to as a meta-algorithm) that can be used in conjunction with machine learning algorithms to improve their performance.[8, 9] AdaBoost has proven to be valid with decision trees, neural networks, and support vector machines.[10, 11, 8, 9] When used with decision trees, AdaBoost allows one or more weak hypotheses to be joined together on a decision to form a strong hypothesis. As long as the weak hypotheses are better than random, then AdaBoost had been proven to be an effective ensemble technique that can provide more accurate results. The version of AdaBoost used in this research is based on the AdaBoost.m2 algorithm, which trains each new hypothesis on the hard-to classify input.[8, 9] This allows easy-to-classify input to be handled quickly, thus permitting more time to be spent on the hard-to-classify input. One question posed in this research is whether AdaBoost can be used decrease the training time of NEAT or rtNEAT using the UCI SPAM data



3 The Experiment

The implementations of NEAT, and rtNEAT downloaded from the Neural Networks Research Group at the University of Texas was used to perform the experiments.[12, 13] The implementation of the AdaBoost algorithm was programmed based of the pseudocode provided in Boosting Neural Networks using method R of that paper.[8] Each of the four techniques (NEAT and rtNEAT, both with and without AdaBoost) was run numerous times, and data was collected. Each run was permitted to train for 100,000 organism evolutions. For NEAT with a population of 100, this equates to 1000 generations, since every organism evolves in every generation. In rtNEAT, this means 100,000 generations will pass, since rtNEAT evolves only one organism per generation. Using NEAT with AdaBoost, where we assume 10 weak hypotheses in AdaBoost, the NEAT algorithm has 100 generations for each hypothesis. While for rtNEAT with AdaBoost, again expecting 10 hypotheses, rtNEAT will have 10,000 generations. It is hoped that this will allow NEAT and rtNEAT enough time to figure out some of the basic connections for each of the hard hypotheses. For all experiments, training is done using parameters from two of the default parameter files provided as part of the NEAT and rtNEAT downloads from their respective websites. These files provide adequate starting parameters, with the understanding that they do not necessarily represent perfect choices for the parameter values. However, the assumption is that any inadequacies in the parameter values will affect NEAT, rtNEAT, and implementations with AdaBoost in the same ways, so the relative runtimes of the different techniques still can be used to determine which one (or ones) perform best.

As stated earlier, the spam dataset from the University of California, Irvine Center for Machine Learning Repository will be used. This dataset provides 56 different attributes calculated for different sample emails. These attributes include such features as maximum run-length of uppercase letters, average run-length of uppercase letters, percentage of words that match the word “order”, and percentage of words that match the word “address”.[2] (Many current spam filters look for words or sentences that are exclusively uppercase.) The dataset has over 6,000 samples that are broken into 3 different sets for this research: one set for training, another for validation, and the third for testing of the final, trained networks. In addition, the data in all three sets has been normalized and standardized. The normalization has been shown to help standard neural networks converge faster.[14]

4 Early Results

Early results of the experiment are inconclusive at this time. However, as can be seen in Figure 1, the results show that NEAT looks promising in the validation part of the dataset, but NEAT with AdaBoost seems to be better in the training and testing datasets. It also shows that NEAT with AdaBoost seems to be more consistent between the three different parts of the dataset. What this figure does not show is that the NEAT algorithm completed more iterations in the same amount of time than the other three algorithms. Until such time as more tests are completed with a greater number of generations, it will be difficult to determine whether adding AdaBoost to NEAT or rtNEAT results in faster convergence.

5 Conclusions and Future Works

The preliminary results do not show whether or not the use of AdaBoost with NEAT or rtNEAT is beneficial. As such, more experiments will be needed. Some of these might be to test different ways the NEAT generations can be used with the AdaBoost algorithm, including what might occur if the number of generations of NEAT is limited. It also might be interesting to see how other ensemble algorithms would work with NEAT instead of the AdaBoost algorithm. As these experiments have focused in on the AdaBoost.m2 algorithm, there are several other versions of AdaBoost that may work better. All of these are interesting topics that could be researched in future experiments.

6 References

[1] M. Vergelis, T. Shcherbakova, and N. Demidova. (2015, May) Kaspersky security bulletin. Spam in 2014. [Online]. Available: https://securelist.com/analysis/kaspersky-security

[2] M. Lichman, “UCI machine learning repository,” 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[3] (2015, May) Neural network research group. [Online]. Available: http://nn.cs.utexas.edu/

[4] K. O. Stanley and R. Miikkulainen, “Efficient reinforcement learning through evolving neural network topologies,” in Proceedings of the Genetic and Evolutionary Computation Conference, ser. GECCO ’02. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, pp. 569–577. [Online]. Available: http://dl.acm.org/citation.cfm?id=646205.758739

[5] K. O. Stanley, “Evolving neural networks,” in Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation, ser. GECCO ’12. New York, NY, USA: ACM, 2012, pp. 805–826. [Online]. Available: http://doi.acm.org/10.1145/2330784.2330917

[6] K. O. Stanley and R. Miikkulainen, “Evolving neural networks through augmenting topologies,” Evol. Comput., vol. 10, no. 2, pp. 99–127, June 2002. [Online]. Available: http://dx.doi.org/10.1162/106365602320169811

[7] K. O. Stanley, B. D. Bryant, and R. Miikkulainen, “Real-time neuroevolution in the nero video game,” Trans. Evol. Comp, vol. 9, no. 6, pp. 653–668, Dec. 2005. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2005.856210

[8] H. Schwenk and Y. Bengio, “Boosting neural networks,” Neural Comp., vol. 12, no. 8, pp. 1869–1887, Aug. 2000.

[9] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” J. Comput. Syst. Sci., vol. 55, no. 1, pp. 119–139, Aug. 1997. [Online]. Available: http://dx.doi.org/10.1006/jcss.1997.1504

[10]S. J. Russell and P. Norvig, Artificial intelligence: a modern approach (3rd edition). Prentice Hall, 2009.

[11]X. Li, L. Wang, and E. Sung, “Adaboost with svm-based component classifiers,” Eng. Appl. Artif. Intell., vol. 21, no. 5, pp. 785–795, Aug. 2008. [Online]. Available: http://dx.doi.org/10.1016/j.engappai.2007.07.001

[12](2015, May) Nnrg software - neat c++. [Online]. Available: http://nn.cs.utexas.edu/?neat-c

[13](2015, May) Nnrg software - rtneat c++. [Online]. Available: http://nn.cs.utexas.edu/?rtNEAT

[14] Y. Lecun, L. Bottou, G. B. Orr, and K.-R. MÃijller, “Efficient backprop,” 1998.

SESSION

NEURAL NETWORKS + OTHER ARTIFICIAL INTELLIGENCE ALGORITHMS AND APPLICATIONS

Chair(s)

TBA

Spiking Neuron Model for Wavelet Encoding of Temporal Signals

Zhenzhong Wang, Lilin Guo, and Malek Adjouadi
Center for Advanced Technology and Education, Florida International University, Miami, FL, USA

Abstract – *Wavelet decomposition is a widely used method to preprocess temporal signals before they could be analyzed by Artificial Spiking Neural Networks (ASNN). This study proposes a biological plausible way to encode the temporal signals into spike trains with wavelet amplitude spectrum represented by the delay phases during each encoding period. The encoding method is presented in the form of a spiking neuron model for easy implementation in ASNN. The proposed neuron model is tested on encoding of human voice records for speech recognition purpose, and compared with results from continuous wavelet transform. The nonlinearity properties and choices of biological plausible wavelet kernels for the proposed encoding method is discussed for the generality of its application.*

Keywords: Phase encoding, Spiking neural network, Wavelet decomposition, Leaky Integrate-and-Fire neuron

1 Introduction

The most significant difference between Artificial Spiking Neural Networks (ASNN) and traditional neural networks is that information in ASNN is represented by spike trains which are a series of pulses with timings of interests. There are mainly two kinds of interpretations developed in signal processing applications about how information is related to spike trains: (1) the rate encoding, which assumes that the information is encoded by the counts of spikes in a short time window; and (2) the spike time encoding which considers information carried at the exact time of each pulse in the spike train. Although the mechanisms for data representation and analysis using biologically-inspired neural networks is still under development, empirical evidence has shown that spike time encoding might be more reliable in explaining experiments on the biology of nervous systems[1], [2].

Both rate encoding and spike time encoding essential in ASNN applications. The easiest way to rate encode an analog signal is to feed it to a Poisson neuron, which fires output spikes at probability proportional to its membrane potential, thus making its firing rate within a short time window proportional to the amplitude of the input signal. Such an encoding method has been adopted by Sprekeler *et al*. [3]. and Keer *et al*. [4] in order to analyze the recurrent ASNN behaviors. Although Poisson neuron model is simple and suitable for theoretical analysis, it was rarely implemented in real-world applications due to its inaccuracy in mapping analog signals to spike trains. De Garis *et al*. [5] introduced another rate encoding method which deconvolves the input signal into its individual spike responses, so that the post-synaptic potential of the encoded spike train could be quite similar to the original signal. Schrauwen and Van Campenhout [6] improved algorithm proposed by De Garis *et al*. by optimizing the deconvolution threshold yielding the so-called Bens Spiker Algorithm (BSA). BSA has been used widely as a rate encoding method for ASNN applications [7]–[9]. The major problem of this type of rate encoding is that an averaging time window is required for each sampling of the input signal, which as a consequence limits the temporal resolution of the encoded signals.

Synchronized spike time encoding, dubbed as Phase Encoding (PE), was also widely used in ASNN application. A simple implementation of PE could be realized by linearly mapping the input signal to the delay of spikes within each synchronizing period [10]. This implementation of PE requires the input signal either to be static or vary at frequencies much lower than the synchronizing frequency. Temporal receptive fields could also be utilized for PE to improve the encoding resolution [11], [12]. To be more biologically plausible, Rumbell *et al*. [13] introduced a synchronizing method which considered spiking neurons as PE units instead of performing linear mapping between analog values and spike delays. Receptive fields in this study were applied to the amplitude dimension instead of the temporal dimension, which yielded good performance for static input data. However, PE method which could accurately encode temporal signals is still under development.

In this paper, we propose a preprocessing unit for the Leaky Integrate-and-Fire (LIF) spiking neurons. The assumption is that a neuron model combining the preprocessing unit with a LIF neuron could be used to encode analog signals with wide frequency range. We will demonstrate that our preprocessing unit could decompose the input signal into wavelet spectrum, and further encode the spectrum amplitude into the delay amount between output spikes and the clock signals. Empirical results of PE encoding of speech records are provided, with linearity, temporal

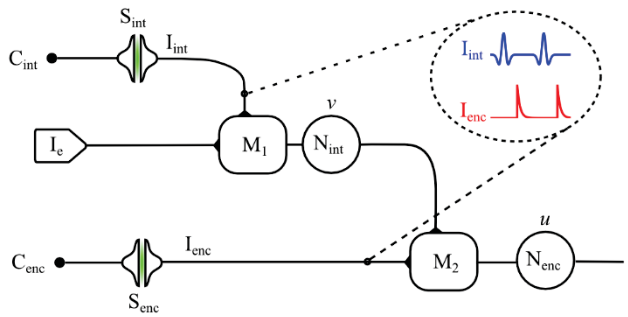


Fig. 1: Structure of the Two-Stage Modulate-and-Integrate Module

resolution issues and possible extension of the encoding method discussed.

2 Encoding Neuron Model

In this section, we will demonstrate that an array of specially designed LIF neurons could perform wavelet decomposition of temporal signals. This special design of a LIF neuron differs from traditional LIF neurons by incorporating a two-stage spike triggered modulate-and-integrate module to pre-process the input signal. Such design was inspired by the multiplication relationship found among afferent synaptic currents in biological neurons [14]. Delay synchronized spikes sent to the two synapses integrated in the special designed LIF neuron could trigger the wavelet transform of the input signal at certain time scales, and encode the spectrum amplitudes into delays between the output fire times and the control spike arriving times. Simulations in this research were conducted using NEural Simulation Tool [15] (NEST) with custom made neuron models.

2.1 Wavelet Encoding Spiking Neuron Model

Although linear summation of synaptic currents and external current has been widely accepted as a simplified relationship among the afferent stimulations in large scale ASNN, the interaction between post-synaptic currents was found to be more complicated in biological nervous system. Koch and Segev [14] found that biological neurons might approximate sum of products among different groups of synaptic currents. Inspired by this finding, we designed a two-stage modulate-and-integrate module, where the multiplication is performed instead of summation between the input signal and synaptic currents. The first stage of the module incorporates the integration of the multiplication of external current and a wavelet shape synaptic current, while the second stage modulate the output from first stage with an exponential decay synaptic current. We will prove that using our preprocessing module together with a LIF neuron, input signal could be decomposed into wavelet spectrum and such spectrum amplitude could be encoded into synchronized spike trains.

In reference to Fig. 1, C_{int} and C_{enc} are delay synchronized clock spikes satisfying:

$$t_i^{\text{enc}} - t_i^{\text{int}} = T_{\text{e}} \tag{1}$$

where T_{e} is the delay phase, t_i^{int} and t_i^{enc} are time of spikes in C_{int} and C_{enc} respectively, with $i = 1, 2, ..., n$ being the index of each spike. The interval of spikes in both C_{int} and C_{enc} is T_{clk}. C_{int} and C_{enc} are converted into post-synaptic current I_{enc} and I_{int} by synapse S_{int} and S_{enc} respectively. Input signal I_{e} is multiplied with I_{int}, and integrated by neuron N_{int} into its state variable v. N_{enc} is a normal LIF neuron, stimulated by the absolute amplitude of v modulated with I_{enc}.

The overall dynamics of this encoding unit could be specified by the following equations:

$$\tau \frac{du(t)}{dt} = -u(t) + \frac{\tau}{C_{\text{m}}} |v(t)| I_{\text{enc}}(t) \tag{2}$$

$$a \frac{dv(t)}{dt} = I_{\text{e}}(t) I_{\text{int}}(t) \tag{3}$$

where u is the state variable of N_{enc}, I_{enc} and I_{int} are summations of the post-synaptic currents of spikes in C_{enc}, and C_{int} respectively, and are defined as follows:

$$I_{\text{enc}}(t) = \sum_i \exp\left(-\frac{t - t_i^{\text{enc}}}{\tau}\right) \Theta\left(t - t_i^{\text{enc}}\right) \tag{4}$$

$$I_{\text{int}}(t) = \sum_i \sqrt{a}\,\Psi\left(t - t_i^{\text{int}} - d, \sigma\right) \Theta\left(t - t_i^{\text{int}}\right) \tag{5}$$

where Ψ is a wavelet mother function used as the PSC for S_{int}, with a representing the scale of the wavelet, $\sigma = a\,f_s$ indicating the time scale of the wavelet related to the sampling frequency f_s, d serving as an offset parameter, and Θ being a Heaviside step function. We selected a shifted Mexican-hat wavelet mother function for Ψ as a demonstration here:

$$\Psi(t, \sigma) = \frac{2}{\sqrt{3}\pi^{1/4}} \left(1 - \frac{t^2}{\sigma^2}\right) \exp\left[-\frac{t^2}{2\sigma^2}\right] \tag{6}$$

Assuming that the length of integration period T_{i} satisfies $T_{\text{i}} < T_{\text{clk}}$, we could define $d = T_{\text{i}} / 2$ in (5), so that the wavelet function is centered within each integration window. Note that both I_{enc} and I_{int} are constructed in a unitless manner for the model simplification.

Suppose that each spike in C_{int} could reset the state variable v of neuron N_{int} to zero, and that σ is significantly smaller than T_{clk}, then (2) could be solved for $t_i^{\text{int}} \le t < t_{i+1}^{\text{int}}$ as:

$$v(t) = \frac{1}{\sqrt{a}} \int_{t_i^{\text{int}}}^{t} I_{\text{e}}(\zeta) \Psi(\zeta - t_i^{\text{int}} - T_{\text{i}}/2, \sigma) d\zeta \tag{7}$$

Suppose further that σ is significantly smaller than T_i, and consider that $\Psi(t,\sigma) \to 0$ when $t > T_i$ or if $t < 0$, then (7) could be approximated by:

$$\begin{aligned} v(t) &= \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} I_e(\zeta) \Psi\left(\zeta - t_i^{\mathrm{int}} - T_i / 2, \sigma\right) d\zeta \\ &= X_w\left(t_i^{\mathrm{int}} + T_i / 2, \sigma\right) \end{aligned} \tag{8}$$

for $t_i^{\mathrm{int}} + T_i \le t < t_{i+1}^{\mathrm{int}}$, where X_w is the wavelet transform of input I_e at translation $t_i^{\mathrm{int}} + T_i / 2$ and time scale σ.

Assuming that:

$$T_i < T_e < T_{\mathrm{clk}} \tag{9}$$

and suppose each input spike from C_{enc} could reset the state variable from u to u_c for neuron N_{enc}, (2) could be solved for $t_i^{\mathrm{enc}} \le t < t_{i+1}^{\mathrm{enc}}$ as:

$$u(\Delta t) = u_c \exp\left(-\Delta t / \tau\right) + V(\Delta t) \tag{10}$$

$$V(\Delta t) = \frac{\tau \Delta t}{C_m} \exp\left(-\Delta t / \tau\right) \left| X_w\left(t_i^{\mathrm{int}} + T_i / 2, \sigma\right) \right| \tag{11}$$

where Δt is the elapsed time since last input spike from C_{enc} arrives at the neuron. Note that the absolute value operation applied to v makes $V(\Delta t)$ a function of the absolute spectrum of the wavelet transform X_w. The absolute spectrum is preferable to power spectrum of the wavelet transform, in the sense that it ensures that the units in equation (10) are balanced without need for extra constants.

We considered two different combinations of reset potential u_c and output firing threshold u_{th} for N_{enc}:

i) Negative threshold: $u_c < u_{th} < 0$.
ii) Positive threshold: $u_c = 0$ and $u_{th} > 0$.

In the first combination, as long as

$$T_{\mathrm{clk}} - T_e > \tau \ln\left(u_{th} / u_c\right) \tag{12}$$

V(Δt) is a non-negative function. The membrane potential will exceed the threshold and an output spike will be generated during each time segment $\left[t_i^{\mathrm{enc}}, t_{i+1}^{\mathrm{enc}}\right)$. The fire delay T in the i-th segment could be solved from:

$$\left| X_w \right| = \frac{C_m}{\tau T}\left[u_{th} \exp\left(T / \tau\right) - u_c \right] \tag{13}$$

where T is guaranteed to be a monotonic decreasing function of $\left| X_w \right|$.

For the second combination, consider that $V(\Delta t)$ is a bell shape function which reaches its maximum when $\Delta t = \tau$, the membrane potential could exceed the threshold only if the wavelet spectrum amplitude satisfies:

$$\left| X_w \right| \ge X_{th} = \frac{u_{th} C_m}{e \tau^2} \tag{14}$$

in which case the firing delay T could be solved from:

$$\left| X_w \right| = \frac{u_{th} C_m}{\tau T \exp\left(-T / \tau\right)} \tag{15}$$

Note that T is always less than τ in (15), which ensures that T is a monotonic decreasing function of $\left| X_w \right|$ when the amplitude spectrum $\left| X_w \right|$ is larger than the threshold X_{th}. If the wavelet spectrum amplitude is smaller than X_{th}, the LIF neuron N_{enc} will not fire during $\left[t_i^{\mathrm{enc}}, t_{i+1}^{\mathrm{enc}}\right)$.

In both combinations discussed above, the wavelet spectrum of input signal I_e is encoded into delay phase T which is the difference between the time of each output fire and the arrival time of the most recent input spike in C_{enc}. Thus, larger wavelet spectrum amplitude corresponds to faster firing after each clock spike.

2.2 Encoding Implementation

Synapses and neurons as described in (2) through (6) are implemented in NEST with a single customized neuron model referred to as the Wavelet Sensor Neuron (WSN). In order to balance the accuracy and efficiency while solving ODEs for WSN, exponential integration method has been adopted to solve the state variable u, and Simpson's rule was applied to the integration for state variable v:

$$u_{n+1} = P_{32} s_n \left| v_n \right| + P_{33} u_n \tag{16}$$

$$v_{n+1} = \frac{h}{6}\left(I_m(t_n) + 4 I_m\left(t_n + \frac{h}{2}\right) + I_m\left(t_n + h\right) \right) + v_n \tag{17}$$

$$s_{n+1} = P_{33} s_n \tag{18}$$

$$I_m(t) = P_2 \left(1 - \frac{\delta t^2}{\sigma^2}\right) \exp\left(-\frac{\delta t^2}{2\sigma^2}\right) I_e(t) \tag{19}$$

$$\delta t = t - T_i / 2 - t^{\mathrm{int}} \tag{20}$$

where subscript n indicates the n-th simulation step, h is the simulation step size, t^{int} is the arrival time of the most recent spike in C_{int}, and P_2, P_{32}, and P_{33} are constant parameters defined by the following relations:

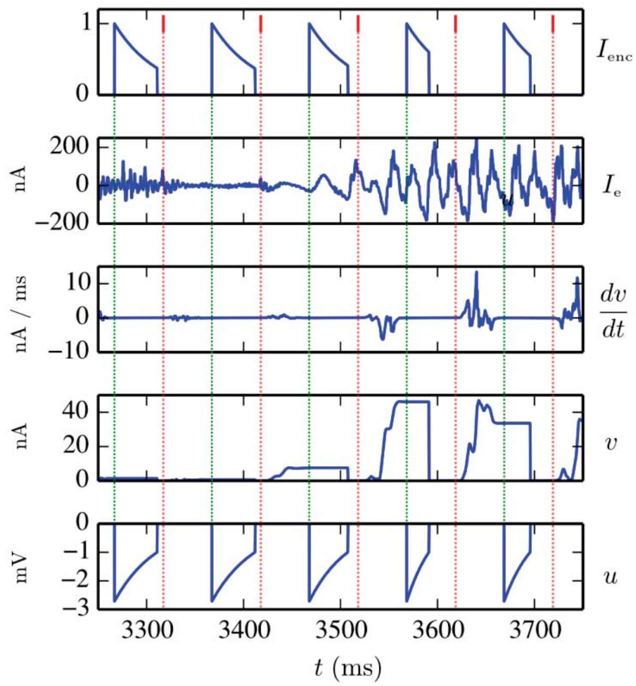


Fig. 2 Time course of variables in one WSN with $\sigma = 5.64$ ms. Red vertical dash lines indicates the arrival times of spikes in C_{int}; green dash lines indicates the arrival times of spikes in C_{enc}.

$$
\begin{aligned}
P_2 &= \frac{2}{\sqrt{3a}\pi^{1/4}} \\
P_{32} &= \frac{\tau}{C_m}\left[1-\exp\left(-h/\tau\right)\right] \\
P_{33} &= \exp\left(-h/\tau\right)
\end{aligned}
\tag{21}
$$

The WSN model incorporates two types of spike receptors to distinguish whether a spike is send to S_{int} or S_{enc}, in the same manner as any other neuron model implemented in NEST which could receive spike input from more than one type of synapses. Input spikes with receptor type I are recognized as spikes sent to S_{int}, which could reset v_n to zero and set t^{int} to the current time; while input spikes with receptor type II are recognized as spikes sent to S_{enc}, which in turn could reset u to u_c and s to zero.

A normal LIF neuron N_{clk} with an exponential decay synapse is implemented in this network as the clock generator. This LIF neuron is connected to itself with axon delay T_{clk} and synaptic efficacy large enough to generate a new output spike from itself. A short strong pulse injected to N_{clk} could initialize the first firing of N_{clk}, and generate oscillatory clock spikes at constant interval approximate to T_{clk}. These clock spikes are sent to type I receptors of WSN neurons with a short delay D_0, and type II receptors with a longer delay T_e.

We built an encoding network to convert the human voice records obtained from Census Database of Carnegie Mellon University [16] (AN4) into spike trains related to the wavelet spectrum. An array of 100 WSNs with $\tau = 45$ ms and σ varies between 0.2 ms and 10.0 ms were implemented in the encoding network. The spike trains could encode frequency components ranging from 100 Hz to 50 kHz in the input signal, which is wider than the human voice frequency limitations. Time constants $T_{clk} = 100$ ms, $D_0 = 1.0$ ms, $T_i = 45$ ms, and $T_e = 50$ ms was selected to meet all the requirements posed by (9). A negative threshold $V_{th} = -1.0$ mV was used in this implementation. The reset membrane voltage was set to $u_c = -2.72$ mV so that the longest spike delay is $T_{max} = 45$ ms, according to the solution of (10) with $V(T) = 0$ mV and $u(T_{max}) = u_{th}$. Since $T_e + T_{max} < T_{clk}$, there is always one output spike from each WSN within one clock cycle.

3 Results and Discussion

The record file "an253-fash-b.raw" from the training set of AN4 database was used as the input to the WSN encoding network. The state variables of each WSN neurons were recorded for the testing purpose. A portion of the recorded variables of one WSN with $\sigma = 5.64$ ms was captured and plotted in Fig. 2

Vertical red dash lines in Fig. 2 represent the arrival times of the clock spikes for the type I synapse receptor of this neuron. Input I_e was modulated with the wavelet kernel for 45 ms after each clock signal. When I_e contains components matching the 5.64 ms time scale of the wavelet function, the WSN generates a larger modulated current, yielding as a consequence a larger state variable v. The clock spikes arrive at the type II synapse receptor of this WSN after 50 ms delay (indicated by the green vertical lines in Fig. 2, which trigger the encoding periods. At the beginning of each encoding period, the integration of v has already finished, thus v holds its value for the whole encoding period. The LIF neuron

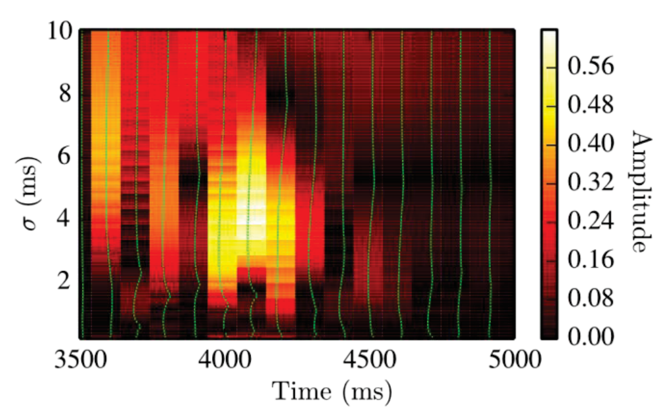


Fig. 3 Comparison of WSN encoding with Continuous Wavelet Transform at corresponding translations. Green lines bars output spikes from the WSN array

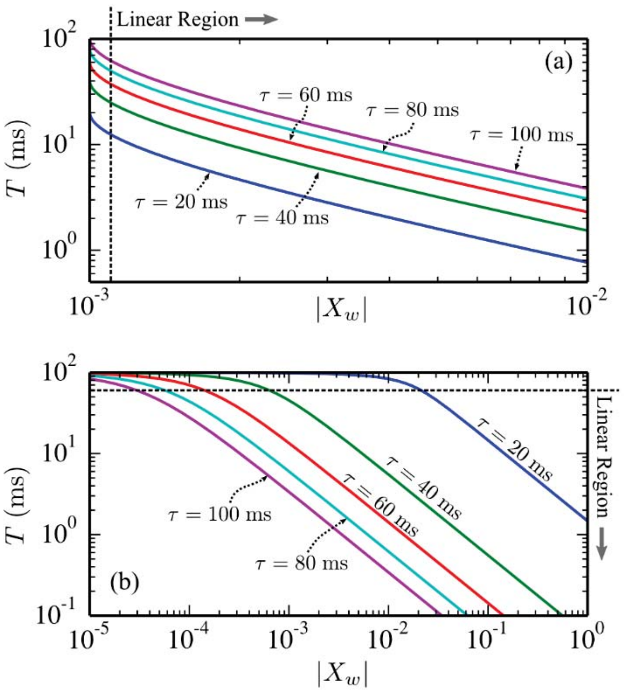


Fig. 4 Logarithm relationship of the input intensity and output spike delay: (a) the relationship of positive threshold WSNs; (b) the relationship of negative threshold WSNs.

incorporated by the WSN would encode the constant v into an output firing delay. It could be found from the records of u that, the WSN fires faster when the input signal I_e contains components matching $\sigma = 5.64$ ms (i.e., periods from 3520 ms to 3720 ms), yet fires slower at almost the end of each encoding period when I_e contains only higher frequency components (i.e., periods from 3120 ms to 3320 ms).

The voice record used in this experiment was the sound of female pronouncing the word "GO". The output spikes of all 100 WSNs were raster-plotted for the time range from 3500 ms to 5000 ms using short vertical green bars as shown in Fig. 3. Continuous wavelet transform using Mexican-hat wavelet was also applied to the same voice record. The wavelet transform at translations $t_i^{\text{clk}} + 22.5$ ms were color coded and superimposed on Fig. 3, where t_i^{clk} are the firing times of N_{clk}. It could be found from Fig. 3 that, the change of the fundamental frequency when pronouncing the word "GO" was clearly captured by the Mexican-hat wavelet transform, and the delay phases of WSN output fires were a good representation of the wavelet spectrum amplitudes during each clock cycle. Such phase encoded spike trains are applicable to any supervised spiking neural learning. Thus, the clustered or classified features of the frequency changes could be used to recognize the word pronounced. The phase delays of the WSN array in this example could substitute for the spectrogram in estimating key characteristics in speech recognition [17], and could support the building of speech perception system using ASNN.

3.1 Encoding Non-linearity

The logarithm relationship between stimulation intensity and the delay phase of encoded spikes in sensory neurons was identified by many neurologists [18]. In many spiking neural network applications which implements PE as the sensing method, a log function was applied to the input signals to mimic the logarithm relationship [9], [12]. The WSN encoding method is highly nonlinear according to (13) and (15), yet the logarithmic relationship between stimulation intensity and the delay phase of spikes is a natural feature of the WSN encoding.

As shown in Fig. 4, the linearity between $\log(|X_w|)$ and $\log(T)$ could be found in certain regions for the five selected WSN neurons with time constants τ being 20 ms, 40 ms, 60 ms, 80 ms and 100 ms, respectively. In Fig. 4(a), positive firing threshold was adopted for these neurons, and the wavelet spectrum amplitude threshold was set to $X_{\text{th}} = 10^{-3}$ for all five neurons. The firing threshold u_{th} for these neurons could be calculated by (14). We could find that WSN could encode $\log(|X_w|)$ to $\log(T)$ in a linear way when $|X_w|$ is in the linear region shown in Fig. 4(a). Different time constants τ introduce different offsets to the linear relationship along the y-axis: larger τ values corresponds to better encoding resolution for small $|X_w|$.

As a comparison, negative firing threshold were used for the WSN neurons in Fig. 4(b), with u_{th} all set to −0.2 mV. u_c for these neurons was adjusted according to:

$$u_c = u_{\text{th}} \exp(\tau_{\max} / \tau) \tag{22}$$

such that the maximum output fire delay was always $T_{\max} = 100$ ms. Linearity could also be found in the linear region indicated in Fig. 4(b), when T is a bit smaller than $T_{\max}$. Different time constants τ introduce different offsets to the linear relationship along the x-axis.

It should be noted that, using the same τ settings, negative firing thresholds provide better logarithm linearity than positive firing thresholds for the encoding of signals with a larger range of $|X_w|$. Since the parameter τ in the WSN neuron is limited by the encoding window length, negative firing thresholds could be a better choice when the encoding linearity is of interest, as demonstrated in this paper when encoding was performed on the example of the human voice record. However, the $|X_w|$ cut-off feature provided by the positive firing thresholds could be useful when only large values of $|X_w|$ are of interest. The threshold configuration as well as the time constant τ should thus be carefully selected for a given application, so that the features of interest in the input signal could be best encoded into the delay of output fires.

3.2 Mother Wavelet Functions

Although a shifted Mexican-hat wavelet mother function was used for the post-synaptic current shape function in S_{int}, it is not required for the WSN neuron to work properly. Any types of wavelet mother functions could be used as the current shape function in WSN, and the input signal will be decomposed according to the mother wavelet functions selected. If a discrete wavelet is demanded, the kernel function of the discrete wavelet at different time scale with proper shifting should be used as Ψ in (5).

More interestingly, since the integration of the wavelet kernel performs only in a limited time duration, the only requirement for $\Psi(t)$ is that:

$$\lim_{t \to \pm\infty} \Psi(t) = 0 \tag{23}$$

and $\Psi(t)$ is not required to be absolutely integrable and square integrable from $-\infty$ to $+\infty$. Some functions, such as the alpha function:

$$\Psi_{\alpha}(t, \sigma) = \frac{t}{\sigma} \exp\left(-\frac{t}{\sigma}\right) \tag{24}$$

could also be used to decompose the input signals.

3.3 Temporal Resolution

Since the wavelet of a WSN is convolved with the input signal only once during each clock cycle, the encoding temporal resolution of one WSN is limited to the clock interval T_{clk}. Considering that the total of integration time T_{i} and that encoding time T_{e} should be smaller than T_{clk}, and the time constant τ should also be smaller than T_{i}, although a decreased T_{clk} could enhance the encoding temporal resolution, it might also harm the encoding range of the wavelet spectrum amplitude. In order to enhance the temporal resolution of a WSN array without interfering with the encoding range, we could still implement multiple WSNs for each time scale selection, but with different D_0 values. Accordingly, the wavelet transform would be performed at different translations within each clock cycle, and thus could significantly enhance the temporal resolution of the encoding without shrinking the length of each clock cycle.

4 Conclusion and Future Work

Encoding of analog signals into spike trains is one of the most important steps for information processing in biological nervous systems. The encoding method we proposed in this paper incorporates the concepts of synaptic current modulation with phase encoding representation. We proved that the proposed WSN model combining a preprocessing unit and a LIF neuron could perform the wavelet decomposition of the input signal, and convert the wavelet spectrum amplitude at certain translation and time scales into the output fire delay of the WSN neuron.

Encoding networks using WSN neurons were implemented in this study to encode an example of a human voice record, with results that are quite similar to continuous wavelet decomposition. The linearity property and limitations of mother wavelet functions of this WSN encoding method were discussed as a guidance for choosing proper parameters for the WSN network to fit a specific application. We also provide a simple method to overcome the temporal resolution limitation posed by the clock signal, so that the wavelet decomposition could be performed with higher temporal accuracy if needed.

Beyond the above contributions, this work also provides an intuitive insight of how stimulations gathered by sensor neurons might be represented and processed by a biological nervous system: the modulation behavior found between dendrites together with the integration feature of a biological neuron could perform decomposition of stimulation signals similar to wavelet transforms, and encode only those features of interest in the stimulation into the spike delay phases.

There are other possibilities for using the proposed encoding method such as: (1) apply graph theory [19] to find the connectivity between encoded spike trains, or (2) build spiking self-organizing-map and supervised learning systems to further process the encoded spike trains, and classify the patterns represented by the encoded spike trains into meaningful symbols. Although the WSN and the encoding network was implemented in the NEST environment, which is based on a digital computing platform, the concepts of WSN is fully compatible with analog computing. We are interested in developing analog circuits to implement WSN encoding network, so that Ultra Large Scale Integration methods could be used to build a highly parallel neuromorphic system.

5 Acknowledgment

The authors are extremely grateful for the support provided by the National Science Foundation under grants CNS-0959985, CNS-1042341, HRD-0833093, IIP-1338922, and CNS-1429345. The generous support of the Ware Foundation is also greatly appreciated.

6 References

[1] S. Panzeri, R. S. Petersen, S. R. Schultz, M. Lebedev, and M. E. Diamond, "The role of spike timing in the coding of stimulus location in rat somatosensory cortex," *Neuron*, vol. 29, no. 3, pp. 769–777, Mar. 2001.

[2] R. S. Johansson and I. Birznieks, "First spikes in ensembles of human tactile afferents code complex spatial fingertip events.," *Nat. Neurosci.*, vol. 7, no. 2, pp. 170–177, Feb. 2004.

[3] H. Sprekeler, C. Michaelis, and L. Wiskott, "Slowness: an objective for spike-timing-dependent plasticity?," *PLoS Comput. Biol.*, vol. 3, no. 6, pp. 1136–1148, Jun. 2007.

[4] R. R. Keer, A. N. Burkitt, D. A. Thomas, M. Gilson, and D. B. Grayden, "Delay Selection by Spike-Timing-Dependent Plasticity in Recurrent Networks of Spiking Neurons Receiving Oscillatory Inputs," *PLoS Comput. Biol.*, vol. 9, no. 2, p. e1002897, Feb. 2013.

[5] H. De Garis, M. Korkin, F. Gers, E. Nawa, and M. Hough, "Building an artificial brain using an FPGA based CAM-Brain Machine," *Appl. Math. Comput.*, vol. 111, no. 2–3, pp. 163–192, May 2000.

[6] B. Schrauwen and J. Van Campenhout, "BSA, a fast and accurate spike train encoding scheme," in *Proceedings of the International Joint Conference on Neural Networks*, 2003, vol. 4, pp. 2825–2830.

[7] N. Nuntalid, K. Dhoble, and N. Kasabov, "EEG Classification with BSA Spike Encoding Algorithm and Evolving Probabilistic Spiking Neural Network," in *International Conference on Neural Information Processing*, 2011, vol. 7062, pp. 451–460.

[8] Y. Chen, J. Hu, N. K. Kasabov, Z. Hou, and L. Cheng, "NeuCubeRehab: A Pilot Study for EEG Classification in Rehabilitation Practice Based on Spiking Neural Networks," in *International Conference on Neural Information Processing*, 2013, pp. 70–77.

[9] N. Kasabov, K. Dhoble, N. Nuntalid, and G. Indiveri, "Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition," *Neural Networks*, vol. 41, no. Special Issue, pp. 188–201, May 2013.

[10] S. Ghosh-Dastidar and H. Adeli, "Improved spiking neural networks for EEG classification and epilepsy and seizure detection," *Integr. Comput. Aided. Eng.*, vol. 14, no. 3, pp. 187–212, Aug. 2007.

[11] S. G. Wysoski, L. Benuskova, and N. K. Kasabov, "Evolving spiking neural networks for audiovisual information processing," *Neural Networks*, vol. 23, no. 7, pp. 819–835, Sep. 2010.

[12] J. A. Wall, L. J. McDaid, L. P. Maguire, and T. M. McGinnity, "Spiking neural network model of sound localization using the interaural intensity difference," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 23, no. 4, pp. 574–586, Apr. 2012.

[13] T. Rumbell, S. L. Denham, and T. Wennekers, "A spiking self-organizing map combining STDP, oscillations, and continuous learning," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 25, no. 5, pp. 894–907, May 2014.

[14] C. Koch and I. Segev, "The role of single neurons in information processing.," *Nat. Neurosci.*, vol. 3, pp. 1171–1177, Nov. 2000.

[15] M.-O. Gewaltig and M. Diesmann, "NEST (NEural Simulation Tool)," *Scholarpedia*, vol. 2, no. 4, p. 1430, Apr. 2007.

[16] J. Kominek and A. W. Black, "The CMU Arctic Speech Databases," in *ISCA Speech Synthesis Workshop*, 2004, pp. 223–224.

[17] P. Gómez-Vilda, J. M. Ferrández-Vicente, and V. Rodellar-Biarge, "Simulating the phonological auditory cortex from vowel representation spaces to categories," *Neurocomputing*, vol. 114, pp. 63–75, Aug. 2013.

[18] T. J. Sejnowski, "Time for a New Neural Code?," *Nature*, vol. 376, no. 6, pp. 21–22, Jul. 1995.

[19] S. Sargolzaei, M. Cabrerizo, M. Goryawala, A. S. Eddin, and M. Adjouadi, "Functional Connectivity Network based on Graph Analysis of Scalp EEG for Epileptic Classification," in *Signal Processing in Medicine and Biology Symposium*, 2013, pp. 1 – 4.

Implementation of Hardware Model for Spiking Neural Network

Jungmin Choi[1], Minwook Ahn[2], and Jong Tae Kim[1]
[1]Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon, Korea
[2]DMC R&D Center, Samsung Electronics, Suwon, Korea

Abstract – *The izhikevich neuron model is well known for mimicking almost all dynamics of the biological neurons like Hodgkin-Huxley neuron models with much less hardware resources. Despite its versatility and biological plausibility, izhikevich neuron model is still not suited for a large scale neural network simulation due to its complexity compared to the simpler neuron models like integrate-and-fire model. In this paper, we implement a Spiking Neural Network (SNN) of the silicon neurons based on the izhikevich neuron model in order to show that it is feasible to simulate a large scale SNN. As a demonstration, we construct our system to simulate a sparse network of 1000 spiking neurons on Xilinx FPGA. During the simulation period (1000ms), the network exhibits a rhythmic activity in delta frequency range around 4Hz. This means that the proposed network can simulate a large scale SNN based on izhikevich neuron model for human cortical system.*

Keywords: Spiking neural network, SNN, Izhikevich model, Biophysical model

1 Introduction

Neural network, mimicking the function of human brain, is widely used for several key applications such as vision processing, speech recognition, and classification. In most studies, the neural network is designed on a large scale to improve the performance of those algorithms. In order to implement the large scale neural networks, there are some approaches. One of the most useful methods is using the neuron model emulation on GPU. This is successful approaches due to the powerful computation resources provided by GPU. However, most CE devices are adapting the approaches via a network-based server, because of its huge computational power. There are some attempts [1] [2] to implement a neural network by hardware in order to overcome such a drawback.

One of the most commonly used algorithms to simulate a large scale neural network and hardware acceleration is the Spiking Neural Network (SNN) models [2]. These models simulate neuronal behavior more similarly than the traditional neural networks. In many researches, these models have designed on FPGAs to examine the feasibility of hardware implementations about the models [3] [4] [5] [6].

In this paper, we implemented a SNN of the silicon neurons based on the izhikevich neuron model, which is known for resembling almost all biological neuron dynamics [7]. Our proposed architecture can simulate randomly-connected SNN network of thousands of spiking neurons with millons of synapses in real-time. The simulated time resolution is configured by Phase Locked Loop (PLL) logic and the parameters of each neuron are stored in on-chip memory.

We explore the feasibility of using FPGA for large scale simulations of SNN based on the izhikevich model. Our experiment shows that our silicon neural network has biological plausibility of spiking behavior and resembles some dynamics of human brains through hardware simulation.

Section 2 of this paper provides background of the SNN and characteristics of the individual spiking neuron models. Section 3 explains the proposed hardware architecture of SNN and their operations. Section 4 discusses the result of this work and analyzes the meaning of data while section 5 concludes this paper.

2 Background

SNN, the third generation of neural network, is characterized the high level of biological realism by using individual spikes [8]. Fig. 1 shows an overview of a spiking neural network. A set of neurons on the left side generates voltage spike or fire. Each spike of neurons goes to the synapse which is connected to the other neurons. Each synapse has a weight and the magnitude of the input spike is scaled with that weight. The scaled spikes are summed to provide the overall input current for the next neuron. After that, the neuron decides its next state using the input current and its current state. According to the changed state, the neuron determines whether or not the generation of voltage spike.

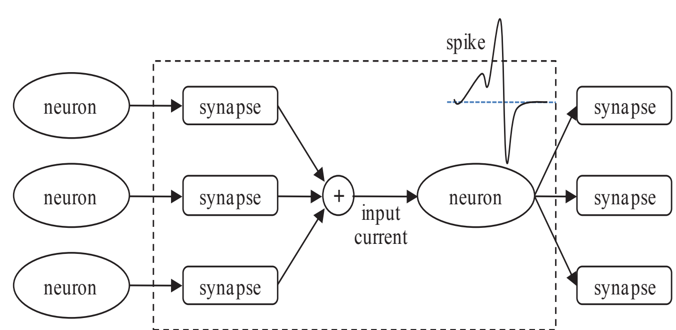


Fig. 1. Overview of a spiking neural network

The SNN include the concept of time into the operating models with spatial information, like real neurons do. The concept is that neurons in SNN do not fire every cycle, but it fires only when a membrane potential exceeds a certain value. When a neuron fires, it provides a signal to other neurons connected by synapses.

There are lots of neuron model in SNN. Several studies compare 11 spiking neuron models base on their biological plausibility and computational complexity [7]. Of these models, current studies of SNN models are generally using the Hodgkin-Huxley, leaky integrate-and-fire and izhikevich models [9].

2.1 Hodgkin-Huxley model

The Hodgkin-Huxley model [10] is one of the most important and biologically accurate models in SNN. It consist of four differential equations (eq. 1-4) and lots of parameters which describe membrane potential, activation of Na and K currents, and inactivation of Na current. The model can exhibit all kinds of spiking patterns and neural dynamics if the parameters are tuned.

The Hodgkin-Huxley model is the most biologically plausible, but one of the main problems is that it is extremely expensive for large scale implementations. Thus, one can use the model only to implement a small number of neurons.

$$\frac{dv}{dt} = (\frac{1}{C})\{I - g_k n^4 (v - E_k) - g_{Na} m^3 h (v - E_{Na}) - g_L (v - E_L)\} \quad (1)$$

$$\frac{dn}{dt} = (n_\infty(v) - n) / \tau_n(v) \quad (2)$$

$$\frac{dm}{dt} = (m_\infty(v) - m) / \tau_m(v) \quad (3)$$

$$\frac{dh}{dt} = (h_\infty(v) - h) / \tau_h(v) \quad (4)$$

2.2 Leaky Integrate-and-Fire (I&F) model

One of the most widely used and simplest models in SNN is the leaky I&F neuron model. It has only one differential equation (eq. 5) where *a, b, c* are parameters of the model, I is the neuron current, and v is the membrane potential of the neuron. When the v reaches its threshold value, the neuron generates spike. Then v is reset to value c according to the expression (6).

Leaky I&F model is incapable of producing rich spiking patterns or neural dynamics, because it has very simple equation and variables. Though the model is computationally effective and ease with that can be simulated or analyzed, it is unrealistic to implement a biologically plausible SNN.

$$\frac{dv}{dt} = I + a - bv \quad (5)$$

$$\text{if} \quad v \geq v_{thresh} \quad \text{then} \quad v \leftarrow c \quad (6)$$

2.3 Izhikevich model

The izhikevich model is described by differential equations of the form (Eq. 7-9)

$$\frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I \quad (7)$$

$$\frac{du}{dt} = a(bv - u) \quad (8)$$

$$\text{if} \quad v \geq 30mV \quad \text{then} \quad v \leftarrow c,\ u \leftarrow u + d \quad (9)$$

where *a, b, c, d* are parameters of the model. The variable I is the neuron current, v is the membrane potential of the neuron and u is the membrane recovery factor which affects membrane reset. After the membrane potential reaches its threshold (*+30mV*), v and u variables are reset according to the expression (9).

In our system, we applied the Izhikevich model as a compromise for the biophysical similarity and computational power. The model is close to the Hodgkin-Huxley model in biological plausibility, but it is similar to the leaky I&F model in computational complexity. Table 1 shows that the excellent performance of the izhikevich model than the other models.

Spiking neuron model	Number of firing patterns	Resources
Hodgkin-Huxley	19	8320 (Spartan-3 xc3sd1800a 4-input LUT)
Hindmarsh-Rose	18	831 (Virtex-2 4-input LUT)
Izhikevich	21	195 (Virtex-5 xc5vlx330t LE)
Leaky Integrate-and-Fire	3	366 (Spartan xc3s1500 4-input LUT)
Quadratic Integrate-and-Fire	6	207 (Virtex-5 xc5vl330t 6-input LUT)

Table 1. The performance comparison of each spiking neuron models

3 Hardware Implementation

3.1 Different types of spiking neurons

The cortical neurons in human brain can be classified into several types according to the firing and bursting pattern [11]. All excitatory cortical neurons are divides into the following three classes : RS(Regular spiking) neuron is the most typical neurons in the cortex, IB(Intrinsically Bursting) neuron fires a stereotypical burst of spikes followed by repeated single spikes, and CH(Chattering) neuron can fire stereotypical bursts of nearly spaced spikes.

And all inhibitory cortical neurons are divided into the following two classes : FS(Fast Spiking) neuron can fire periodic trains of action potentials with extremely high frequency without any adaptation, and LTS(Low-Threshold Spiking) neuron can also fire high frequency trains of action potentials, but with a marked spike frequency adaptation.

In order to simulate accurately neuron dynamics in our system, we used these all kind of models by adjusting the parameters of izhikevich neuron model.

3.2 Structure

We construct our system to simulate a sparse network of 1000 spiking neurons with 1 million synaptic connections in real time (1*ms* resolution) using verilog on Xilinx ISE design suite. For efficient use of hardware resources, we design the actual logic so that only a single neuron is operated per 1 clock cycle. And we could obtain same results by the clock rate 1000 times faster. This system is shown in Fig. 2

Each neuron is randomly connected to 1000 other neurons with an excitatory-inhibitory ratio at 4:1. And we set each parameter (*a, b, c, d*) for modeling the rich spiking neuron structures (RS, IB, CH, FS, LTS) [11] described above are uniformly distributed per each time resolution. A RS model corresponds to c = -65mV (deep voltage reset) and d = 8 (large after-spike jump of u). An IB model corresponds to c = -55mV (high voltage reset) and d = 4 (large after-spike jump of u). A CH model corresponds to c = -50mV (very high voltage reset) and d = 2 (moderate after-spike jump of u). A FS model corresponds to a = 0.1 (fast recovery). And a LTS model corresponds to b = 0.25. There are synaptic connection weights which can scale the magnitude of input spike from previous neurons. And each neuron receives a noisy thalamic input, besides the synaptic input. These parameters and weights would be floating-point in software, but this is infeasible in hardware. Thus we use fixed-point for variables, constants and all arithmetic operations.

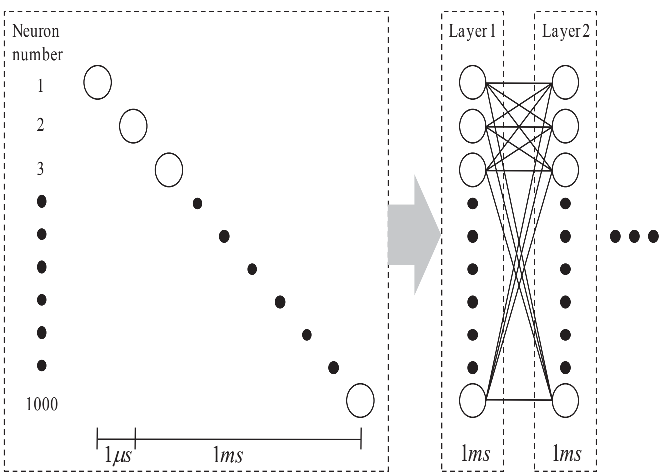


Fig. 2 The hardware structure of proposed SNN

3.3 Operation

To explore the feasibility of the proposed architecture and spiking neuron model explained in the previous section, we implemented the SNN structure on a Xilinx FPGA (part XC7Z020). Fig. 3 shows the overall design of this system. The system repeats the arithmetic operation to simulate the SNN model in time steps. According to izhikevich differential equations and synapse weights, each time step requires calculation of neuron states and summation of scaled input spike.

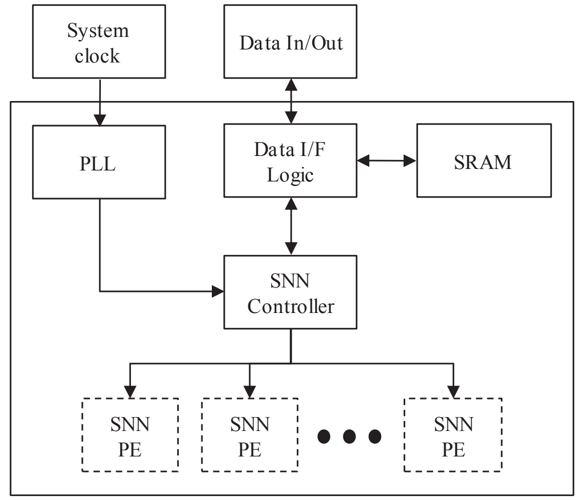


Fig. 3 Overall SNN design on FPGA

Generally, FPGA has several megabytes on-chip memory to store and load frequently accessed data in specific area. In our system, the internal RAM stores the parameters and weights which will be assigned to each neuron. And by request of SNN controller, it loads the parameters for Processing Elements (PEs).

A PLL can adjust the time resolution of SNN controller by dividing a system clock. It is useful for reconfiguring the SNN operating clock or time resolution and for improving the flexibility of system.

The main modules on FPGA consist of the following three components.

1) A SNN controller. This module receives a clock frequency from the PLL and constitutes the operations of the SNN PEs. It provides the equation parameters or synaptic weights to the SNN PEs in each clock-cycle from internal RAM while it delivers the final output data of SNN PEs to the data interface logic. This module also controls the output indices of the fired neurons and postsynaptic current which is accumulated from the previous synapses.

2) A SNN PE. This module is a set of izhikevich neuron models. It implements computations given by equation (7-9) and generates a local spike based on this set of neurons. They are connected together by the SNN controller which manages related parameters and schedules the operation of PEs.

3) A data interface logic. This module can take the neuron parameters and configurable values of network from external memory or test bench. Then it distributes these data to the internal RAM or the SNN controller. Also, it provides parameters of neuron to SNN controller from internal RAM when SNN controller requests these data.

4 Results

We developed a randomly connected SNN architecture on FPGA and the resource utilization of the FPGA implementations is 54% (LUTs).

To examine the performance of system, we use the Xilinx ISE hardware simulation which is shown in Fig. 4. As we intended, the local neuron generates a spike in $1us$ intervals. Also, the figure indicates that the system changes parameters of izhikevich equation to model excitatory or inhibitory cortical neurons.

In order to estimate the spiking or firing rate of neuron models, we extracted the simulated wave. Then, we analyzed the data by using MATLAB, as shown in Fig. 5. In the simulation period (1000ms), we could find that the network exhibits rhythmic activity in delta frequency range around 4Hz. The delta wave is one of the four fundamental types of brain waves, called deep sleep waves [12].

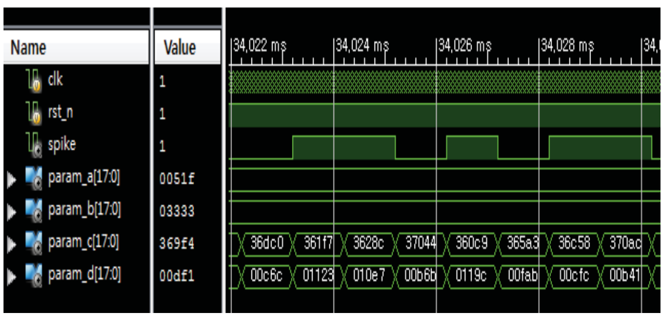


Fig. 4. Result of hardware simulation

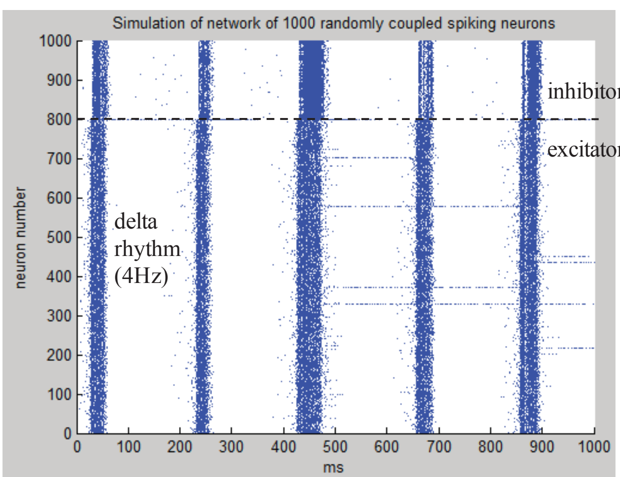


Fig. 5. Simulation of a SNN of 1000 randomly coupled spiking neurons

5 Conclusions

In this paper, we implement the izhikevich neuron model based on FPGA (part XC7Z020) to simulate a large-scale spiking neural network of 1000 spiking neurons with 1 million synaptic connections in real time.

Our SNN structure reproduces the behavior of biological neurons and we could find that our system has biological plausibility. Also, our results indicate that the proposed neural network is suitable for large scale izhikevich models based on cortical neurons while using computational resources efficiently.

6 References

[1] David B. Thomas and Wayne Luk. "FPGA accelerated simulation of biologically plausible spiking neural networks" *IEEE Symposium on Field Programmable Custom Computing Machines*, 2009

[2] Kenneth L. Rice, Mohammad A. Bhuiyan, Tarek M. Taha, Christopher N. Vutsinas and Melissa C. Smith. "FPGA Implementation of Izhikevich Spiking Neural Networks for Character Recognition" *International Conference on Reconfigurable Computing and FPGAs*, 2009

[3] H. Shayani, P.J. Bentley, and A. M. Tyrrell, "Hardware Implementation of a Bio-plausible Neuron Model for Evolution and Growth of Spiking Neural Networks on FPGA", *Conference on Adaptive Hardware and Systems 2008 (AHS '08), pp. 236-243, Jun.* 2008.

[4] A. Cassidy, S. Denham, P. Kanold, and A. G. Andreou. "FPGA Based Silicon Spiking Neural Array," *IEEE International Workshop on Biomedical Circuits and Systems (BIOCAS'2007), Montreal, pp. 75-78, Nov*. 2007.

[5] M. La Rosa, E.Caruso, L. Fortuna, M. Frasca, L. Occhipinti, and F. Rivoli, "Neuronal dynamics on FPGA: Izhikevich's model,' *Proceedings of the SPIE*, *vol. 5839, pp. 87-94,* 2005.

[6] M. Mokhtar, D. M. Halliday, and A. M. Tyrrell, "Hippocampus-Inspired Spiking Neural Network on FPGA," *Proceedings of the 8th International Conference on Evolvable Systems: From Biology to Hardware*, *pp. 362-371,* 2008.

[7] Eugene M. Izhikevich. "Which model to use for cortical spiking neurons?" *IEEE Transaction on neural network*, vol. 15, no. 5, Sep, 2004

[8] Jilles Vreeken. "Spiking neural networks, an introduction." *Institute for Information and Computing Sciences, Utrecht University*

[9] Mohammad A. Bhuiyan, Rommel Jalasutram, and Tarek M. Taha, "Character recognition with two spiking neural network models on multi-core architecture" *International Conference on Reconfigurable Computing and FPGAs*, Dec, 2009

[10] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and application to contribution and excitation in nerve" *J. Physiol., vol. 117, pp.500-544,* 1954.

[11] Eugene M. Izhikevich. "Simple model of spiking neurons" *IEEE Transaction on neural network*, vol. 14, no.6, Nov, 2003

[12] Eugene M. Izhikevich . "Polychronization: Computation with Spikes" *Massachusetts Institude of Technology*, 2005

Application of Self-Organizing Feature Maps to Water Resources Projects

B. Hsieh[1], and J. Ratcliff[1]
[1]US Army Engineers, Research and Development Center, Vicksburg, Mississippi, USA

Abstract – *The self-organizing feature maps (SOFM) is a kind of ANNs (Artificial Neural Networks) method which is capable of clustering, classification, estimation, prediction, and data mining in a wide-spread range of disciplines. Two types of complex water resources related applications, namely, watershed hydrology and coastal storm surge, are demonstrated here. The former application is to find the best match watershed from a large knowledge base of over on thousand quantifying watersheds and to determine the reliability of "transplant" watershed information during the clustering processes while the later application uses SOFM adequately characterize the storm surge response, and provide a means for reliably estimating surge response for storms not simulated with a selected physics-based surge model. The computational procedures and results are presented.*

Keywords: SOFM, Neural Networks, watershed similarity, storm surge response, water resources projects.

1 Introduction

Unsupervised training means the networks learn from their own classification of the training data, without external help. It is assumed that class membership is broadly defined by the input patterns that share common features, and that the network will be able to identify those common features across the range of input patterns. Self-organizing Feature Maps (SOFM) is a special kind of neural network that can be used for clustering tasks. Only one map node (winner) at a time is activated corresponding to each input. The location of the responses in the array tends to become ordered in the learning process as if some meaningful nonlinear coordinate system for the different input features were being created over the network. This illustrates an important and attractive feature of SOFM applications, in that a multi-dimensional input ensemble is mapped into a (one or) two-dimensional space, preserving the topological structure as much as possible. The SOFM is a set of artificial neurons, which are ordered in N^n space. A two dimensional array (n=2) is the most common map and is used to map an input signal in R^m (m >n) space onto the two-dimensional space. Basically, an SOFM typically consists of two layers (Figure 1). One is an input layer into which input feature vectors will be fed and other layer is a two-dimensional competitive layer, which orders the neuron's responses spatially. Neurons can be arranged on a rectangular map so that they can be implemented using a simple 2-D data array. A hexagonally arranged neuron map is, however, often used because it has the advantage of the Euclidean distance (equi-distance) between adjacent neurons (Kohonen [6]). The detailed theoretical development for the algorithms for both supervised and unsupervised ANNs can be found in the most textbooks.

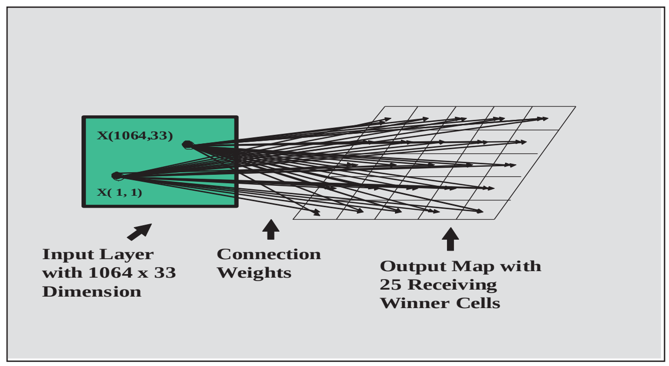


Figure 1. SOFM with 5 x 5 matrix lattice

Visualization techniques to depict the data structure of the feature space in the form of clustering of neurons in the 2-D SOFM have been developed (Ultsch [10]). This visualization typically uses the grayscale to illustrate the distance between connection weights. The light shading typically represents a small distance and the dark shading represents a large distance.

This type of visualization is useful as long as relatively clear cluster boundaries exist or the granularity of the distance differences is large. When the cluster boundaries get fuzzy or the granularity of the distances become too small to represent with the grayscale, it becomes increasingly difficult to identify fuzzy cluster landscapes. Moreover, since all distance values are normalized, only relative (qualitative) analysis is allowed. Subsequently, this "grayscale distance map" cannot be used to compare different SOFM mapping results.

When the SOFM is used to discover some structure of the given samples in the feature space, it is often useful to visualize the finding in the form of clustering formations. Visualization techniques to depict the data structure of the feature space in the form of clustering of neurons in the 2-D SOFM have been developed (Ultsch [10]). This visualization typically uses the grayscale to illustrate the distance between connection weights. The light shading typically represents a small distance and the dark shading represents a large distance.

NeuroDimensions [7] developed a visual version for the Kohonen topological feature maps to check the performance of SOFM. Three basic windows used for evaluating the clustering are quantization metric, united distance, and frequency.

Quantization Metric: It produces the average quantization error, which measures the goodness of fit of a clustering algorithm. It is the average distance between each input and the winning process element (PE). If the quantization error is large, then the winning PE is not a good representation of the input. If it is small, then the input is very close to the winning PE. The quantization error is best for comparing the clustering capabilities between multiple trainings of the same SOFM on the same point.

Unified Distance: This is the distance between PE clustering centers. The weights from the input to each PE cluster centers of the SOFM. Inside a cluster of inputs, SOFM PEs will be close to each other.

Kalteh, Hjorth, and Berndtsson [5] reviewed of the SOFM approach in water resources: analysis, modeling and application. They reviewed several applications in water resources processes and systems including riverflow and rainfall-runoff, precipitation, surface water quality, climate change, environment, and ecology subjects. Several recent studies (Chen, Chen, and Lin [1], Srinivas, etc. [8], and Gomez-Carracedo, etc. [2]) also successfully addressed for water resources and environmental related projects using SOFM.

2 Watershed similarity analysis

The ability to predict watershed hydrologic conditions and the associated potential for flooding to occur plays a significant role in planning and operational activities. To make highly accurate hydrologic predictions, either physically-based or system-based, the system parameters and prediction variables are sometimes unavailable or even totally missing. This certainly curtails the capability of prediction, particularly for operations where very little time is available to conduct the analysis. Very often, the information for a particular watershed may be entirely unavailable; this situation could be resolved by the similarity concept.

The purpose of this approach is to find the best match watershed from a large knowledge base and to determine the reliability of "transplant" watershed information such as hydrologic and climatic parameters (Hsieh and Jourdan [3]). The degree of similarity is based on inter and intra relationships among many geologic, soil, hydrologic and climatic factors. Various methods have been employed to analyze the similarity between two objects.

2.1 Geospatial knowledge base development

GIS data often includes satellite and other remotely sensed imagery. An example of the analysis of imagery involves either supervised or unsupervised classification. Unsupervised classification of imagery involves the analysis of color or black and white pixels of the image for the purposes of classifying image objects and entities, where, tone, texture and hue are used. Supervised classification of imagery involves referencing the pixels to actual field or site conditions and color balancing of the image for similar classification purposes. ANNs are increasingly being used for the purpose of determining spatial patterns. In the area of landscape ecology, the landscape pattern is an important factor enabling classification. Indeed, more recent developments in the area of remote sensing analysis involve ANNs for the analysis of images for the purposes of classifying objects.

Geospatial data of geographic locations and characteristic natural and constructed features were gathered for the database development. GIS databases were utilized for this endeavor; specifically the EPA's Better Assessment Science Integration Point & Non Point Sources (BASINS) system provided the 300-meter USGS Digital Elevation Model (DEM), Land Use/Land Cover, Soils (STATSGO), and watershed gauge locations within the conterminous United States of America (Figure 2). These gauge locations were selected with the criteria of 100 percent complete dataset for medium- to moderately large- sized basins, 6 to 7900 km^2.

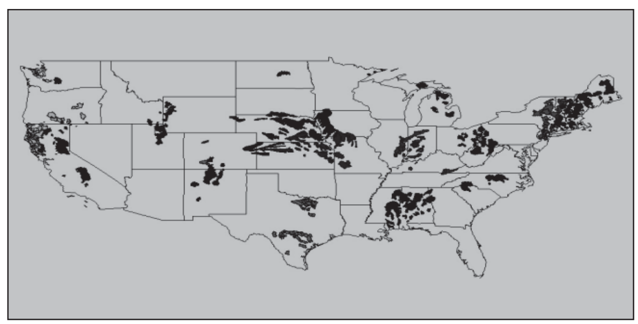


Figure 2. Watersheds within the conterminous United States

Watershed development was conducted with the Environmental Systems Research Institute's ArcGis/ArcView and the Department of Defense's Watershed Modeling System (WMS). From the GIS databases, data was extracted, projected and shaped into Arc/Info griddled ASCII data as input into the WMS interface where basin delineation and parameter estimations were conducted. Watershed parameters such as Drainage Area, Basin Slope, Basin Length, Basin Perimeter, etc. were among the variables derived for the ANN's analyses. Watersheds selected were within a 10 percent margin of error when the areas were compared with recorded drainage areas from BASINS.

From these selected watersheds, mean daily flow data for their respective periods of record were compiled for the ANN's' verification process. In addition, thirty-year mean monthly and annual precipitation, as well as temperature data, were derived from *PRISM (**P**arameter-elevation **R**egressions on **I**ndependent **S**lopes **M**odel)* and presented as GIS coverages. Subsequent GIS analyses produced mean monthly and annual, precipitation and temperature data, for all selected basins. The final knowledge base has the dimension of a 1064 watersheds x 70 variables matrix with final relevant parameters listed as follows.

Geometric Parameters:
Basin Area, Basin Slope, Basin Average Elevation, Basin Shape Factor, Basin Sinuosity Factor, Average Overland Flow Distance, Maximum Flow Distance, Maximum Flow Slope, Maximum Stream Slope, Centroid to Nearest Point of MaxFlowDist

Land Use/Land Cover Parameters:
Residential/Industrial, Agricultural Land, Rangeland, Forest Land, Open Water, Wetlands, Exposed Rock, Tundra, Glaciers

Soil Type Parameters:
Sands and Gravel, Silts, Sandy Loam, Clays

Hydrologic Parameters:
Seasonal and Annual Precipitation, Seasonal and Annual Temperature

A data-driven computational procedure including knowledge base and two components of ANNs (clustering and classification) and prediction (verification) was developed. Takatsuka [9] applied SOFM and interactive 3-D visualization to geospatial data. Ultsch, Korus, and Kleine [10] developed the integration of neural networks and knowledge-based systems in medicine.

2.2 Demonstration example

From the knowledge base, all the geometric parameters, the land use/land cover, the soil types and the seasonal and annual mean values of both precipitation and temperature were used to test this calculation procedure. In order to test the reliability of the system development, three sizes of watershed are selected to examine the performance. The detailed search process is only presented in the first example.

The goal of this test is to use a known watershed (gage number 4288000) to search for the best similar watershed. This part of study is divided into two portions. While the clustering analysis is used to identify the similarity between the watersheds, the classification analysis is used to verify the clustering performance. To check the reliability of the prediction, time series hydrographs are used to compare the resulting search pattern. In this procedure, the hydrograph of gage 4288000 is hidden purposely in order to check the performance of the system once the best similar watershed is found.

During the clustering computation, a 5 x 5 matrix of SOFM is initially selected. Through repeated iterations (usually 200) of the examination of frequency, unified distance, and quantization of the unsupervised synapse, an optimal clustering set to distribute the winner for each watershed is obtained (Figure 1). The numbers in this 5 X 5 matrix show the most similar watershed within the same group (there are 25 groups in this case).

For classification, the problem was trained with (Multi-layered Feed Forward neural Networks (MLP) ANNs and the outcome provided the confidence level of the clustering analysis, which resulted in a successful classification rate of about 91 percent meeting the target. This result indicates that watershed 4288000 belongs to the group with 103 (group 7) most similar watersheds from the original 1063 possible candidates. This clustering-classification process is repeated until the final target watershed is found. This process is called search iteration. Figure 3-4 show the classification verification during the first and second search iterations respectively. It is noted that the size of clustering for this iteration has been reduced to a 3 X 3 matrix.

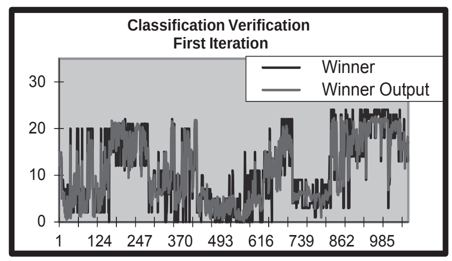


Figure 3. Classification verification for the first system iteration with 1064 watersheds (group – X-axis; number of assigned watershed – Y-axis)

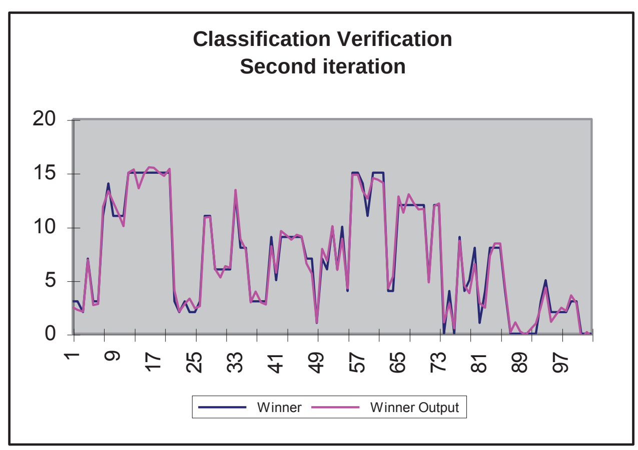


Figure 4. Classification verification for the second system iteration with 103 watersheds (group – X-axis; number of assigned watershed – Y-axis)

The final candidate for this search is the watershed number 01144000. This implies that the flow patterns from station 01144000 will be most similar to those of station 04288000. Flow hydrograph comparisons between these two stations during the period 1999-2001 are shown in Figure 5. Although the flow pattern, particularly, the phase matches very well, the performance of the amplitude representations is dissatisfactory.

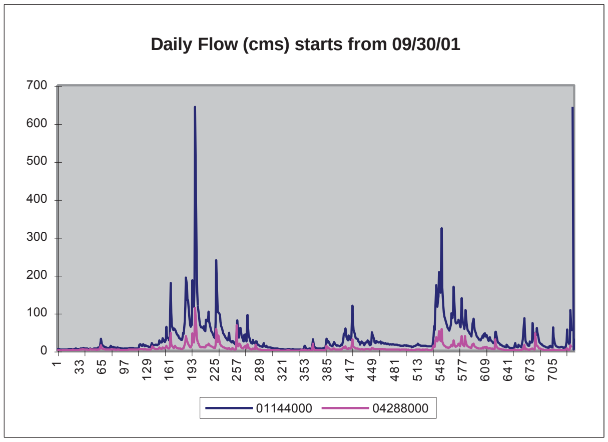


Figure 5. Most similar flow (cms) (01144000 - pink) vs. observed flow (04288000 - blue) flow – X-axis; days – Y- axis)

When examining the involved parameters between these two watersheds, the area and maximum flow distance showed a significant difference. The estimated hydrograph was adjusted by taking the area ratio of station 04288000 and station 01144000 (Figure 6).

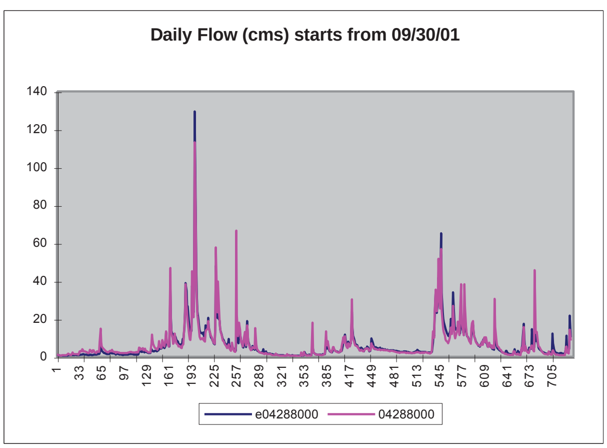


Figure 6. Flow (cms) estimation (e04288000 - pink) vs. observed flow (04288000 - blue) after basin area ratio adjustment for 33 inputs approach (flow – X-axis; days – Y- axis)

The major element in making this integration system a success is to tune the clustering group as well as rechecking the performance of the classification process. But the identification of the reliability for application also requires data on how well the "transplant" performs. Therefore, a series of combinations including the features of input parameters is adopted. Table 1 summarizes the performance due to the selection of input parameters. This indicates that the important group parameters are hydrologic, geometry, soil type, and land use. The performance difference between geometry and hydrologic groups is quite small. The magnitude of hydrographs could not be adjusted by ratios obtained using hydrologic, soil type, and land use groups since they do not contain the basin area factor after the best candidate is found.

Table 1. Sensitivity test due to input parameters

Parameters	Candidate Watershed	Correlation Coefficient	Mean Error
All Groups	0114000	0.92	0.17
Geometry	4282000	0.82	-7.80
Hydrologic	4288000	0.83	2.99
Land Use	2472000	0.10	- 28.34
Soil Type	1170100	0.67	- 4.25

2.3 Summary of watershed similarity analysis

An integration of database and ANNs learning was used to identify a very complex nonlinear watershed similarity analysis for military hydrology applications. While the unsupervised ANNs, such as SOFM, were used to perform the clustering of watershed characteristics, the supervised ANNs were used to identify the best match candidate watershed for classification analysis. The search procedure requires several iterations of the clustering-classification loop. The current knowledge base consists of 67 geometric, hydrological, land use, and soil type factors for 1064 selected watersheds. After removing the internal dependency and examining the annual and season representation, 33 factors were selected for final analysis.

Three demonstration examples, including random selection, average size, and median size watersheds were used as the target to search for the best match corresponding candidate. The first example obtained a good correlation coefficient (0.92) for hydrograph prediction (2 years daily flow). It is found that the basin area ratio provides a reasonable factor for making the adjustment for hydrograph prediction. The preliminary sensitivity tests indicate that the hydrologic factors are the best factors in producing a fitness for transplant. In general, monthly hydrograph comparisons have better agreement than the daily hydrographs for both average and median size watershed examples. The most significant reliability is obtained when many watershed patterns are included in the knowledge base. Development of an automated search procedure for a unique solution is the direction proposed for further research.

3 Storm surge runs analysis

Numerical simulation is the most accurate and efficient modern technique to calculate the surge response during a storm/hurricane event. Recently, a very successful interagency team effort has been made to model the storm surge response for hurricane events along the northern coast of the Gulf of Mexico. The dependencies between surge and waves are treated through coupled models, and a probabilistic approach has been adopted for calculating inundation levels and their associated probabilities. However, numerous model runs are required to cover a wide range of possible hurricane scenarios to meet the management and project design needs. This newly established modeling framework requires a significant amount of resources including personnel and computer resources, as well as contract labor and other factors which raise project costs and completion time requirements. Good planning with an alternative technology path that can reduce costs for projects is highly desirable. Simulation using ANN techniques was examined as a possible tool for reducing the resources required to make storm surge estimates for design purposes. Often in design, a large set of storm surge simulations must be made for each of a series of different project alternatives (such as different levee alignments). Increasing numbers of alternatives dramatically escalate the computational requirements for a detailed modeling approach.

3.1 LACRP storm surge runs example

A computational procedure (Hsieh and Ratcliff [4] to perform this effort is shown in Figure 7. The first step is to identify the significant storm parameters and use the resulting surge responses to build the ANNs model. The performance of this ANNs model is critical to assure the right input parameters are selected. The second step of the approach is clustering analysis, using SOFM to separate the similar storm patterns from the knowledge base (input parameters only), and to form a number of subgroups. The third step is to split each subgroup into two components: training and testing storm sets. The ensemble training component from all subgroups along with corresponding surges is the knowledge base which is assumed to represent the required ADCIRC runs, while the ensemble testing component from all subgroups along with corresponding surges are considered to be the unnecessary ADCIRC runs. More ADCIRC runs in the final ensemble testing group means a higher percentage of runs saving that can be obtained under the good performance of testing group from ANNs modeling. Although there is no particular rule to follow how to separate the training and testing group, but at least two numerical model runs from each subgroup need to selected to be the training group if this subgroup contains more than or equal to two numerical model runs. For a large number of numerical model runs, the decision is based on either second level of clustering or the variation for the most sensitive storm parameters.

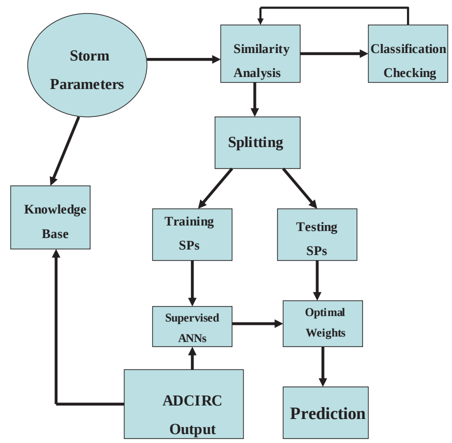


Figure 7. Computational procedure to determine necessary surge model scenario run.

They are 5 storm input parameters (CpLand (central pressure at landfall), VelAvg (forward translational speed of the storm center, which was assumed to be constant along the specified track), Rm (radius to the maximum winds), MaxWind (maximum wind speed at landfall), Distance (distance from the storm center at landfall to the location of interest), and Angle (angle between the storm center at landfall and the location of interest). The corresponding surge produced by the storm at the location of interest is the output. Since CpLand and Distance are negatively correlated to the surge response, a negative sign for both inputs is taken. The Angle for each storm is further decomposed into cosine (Angle-x) and sine (Angle-y) components. The first 4 input parameters are considered as global storm parameters while the remaining input parameters (Distance, Angle-x, and Angle-y) are treated as local, or positional, parameters.

This demonstration shows results for a computational point (141, circled in Figure 8) from among many considered in the LACPR study (Figure 8) to illustrate the ANN application concept. Usually, a correlation coefficient analysis is conducted between all the inputs and corresponding output in order to check the sensitivity of the system. Figure 9 shows the most significant input parameter – angle for point 141 with X and Y components over 152 storm events. Figure 10 illustrates the results of ANNs modeling for point 141. MLP is the training algorithm and total iterations are 5000. With high statistical significance (high correlation coefficient and low mean absolute error, for example), the ANNs modeling proves to be a satisfactory tool to quantify the relationship between inputs and output. The blue lines shows the computed peak surge using the ADCIRC model for all storms while the red line shows the computed peak surge values using ANNs modeling. This is the first step to identify the applicability for selected ANNs technique.

A 5x5 SOFM clustering analysis was then applied to group the dimensional 152x7 information (7 input parameter factors for each of 152 storms). The choice of the size for SOFM process is based on how detail you would like to deal with the clustering from the system. While large matrix may break 152 storms into too many pieces, small matrix may require the second level of clustering.

Figure 8. Surge response points from LACPR ADCIRC model near New Orleans area.

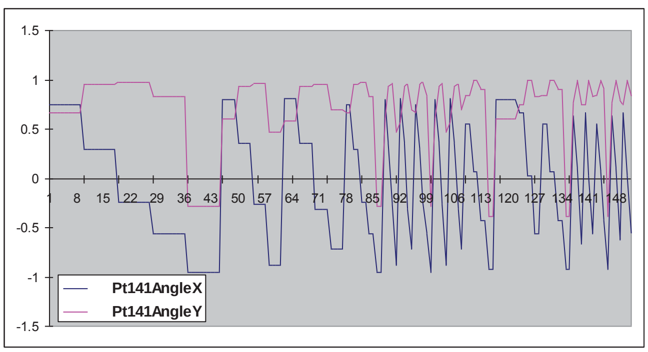


Figure 9. X and Y components for approach angle from point 141 (x-axis represents storm numbers and y-axis represents sine/cosine of the angle, between 1 and -1).

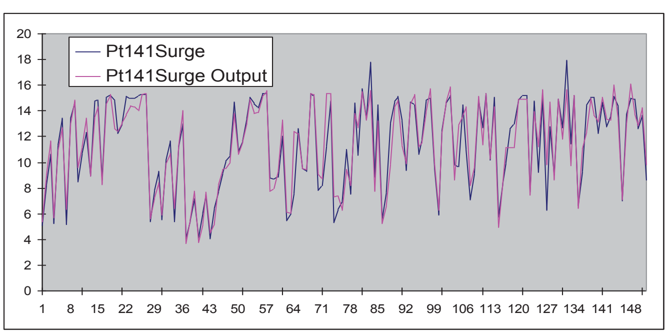


Figure 10. Comparison of results from ANNs- MLP training for point 141 and calculated surges from ADCIRC simulations of 152 storms based on 7 storm input parameters (pink represents ANNs simulation and blue shows ADCIRC results; x-axis represents storm number and y-axis represents surge response (ft)).

The ratio of total storm numbers to total subgroup numbers estimates the proper matrix size is 5x5 or 4x4. The final destination for each represented storm after iteration process for position adjustment is called "Get Winner". An optimal pattern distribution matrix for these 152 storms all reach "WINNER" is shown in Figure 11.

17	0	11	4	13
0	5	4	3	7
7	3	3	5	13
5	3	1	3	2
9	6	9	7	12

Storm Patterns for Training

9	0	6	2	7
0	3	2	2	4
4	2	2	3	7
3	2	1	2	2
5	3	5	4	5

Storm Patterns for Testing

8	0	5	2	6
0	2	2	1	3
3	1	1	2	6
2	1	0	1	0
4	3	4	3	7

Figure 11. A 5x5 SOFM clustering analysis and its training and testing components fro152 LACPR ADCIRC runs based on 7 storm parameters (point 141).

The number for each grid cell of the matrix shows that similar patterns are found from 7 input parameters. It is noted that the "0" value for a particular grid cell in the matrix indicates that there is no storm falls that specific pattern. It usually happens when too large size of matrix is assigned or too little variation of pattern does exist. The splitting process is then applied to separate the storms within a grid cell into a training component and a testing component after an ensemble process is conducted by collecting the minimum required storm events into the training group and putting the remaining events into the testing group. . Since the Angle-y was found to be the most significant parameter, it was used as a criterion, including extreme values and part of represented values from this parameter, to determine into which component each storm should go. The lower part of Figure 11 presents the final assignment of storms into the training and testing components. The ANNs training, using 85 selected storms with surge as output was applied to examine performance for the testingcomponent. Figure 12 illustrates performance of the testing component, the 67 selected storms (a correlation coefficient 0.912 was achieved). From this analysis, it is possible to avoid 44 percent (67 out of 152) of ADCIRC simulations, if the storm simulations have to be repeated.

Results also suggest that it might be possible to intelligently reduce the number of storms considered in simulations to look at various alternatives. To compare the surge frequencies computed based upon these three series, the response surges for return periods up to 2000 years are computed. The maximum deviation is about 0.18 m (0.6 ft) between original ADCIRC runs and this combination approach for point 141 (Figure 13).

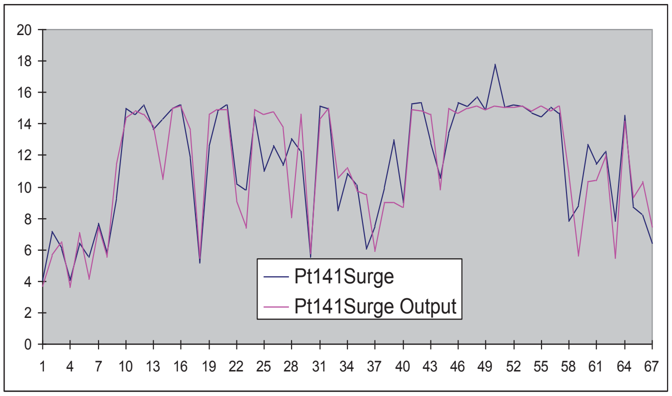


Figure 12. Performance of testing components of 152 storms for point 141 (pink represents ANNs simulation and blue shows ADCIRC results; x-axis represents storm numbers for testing component and y-axis represents surge response (ft))

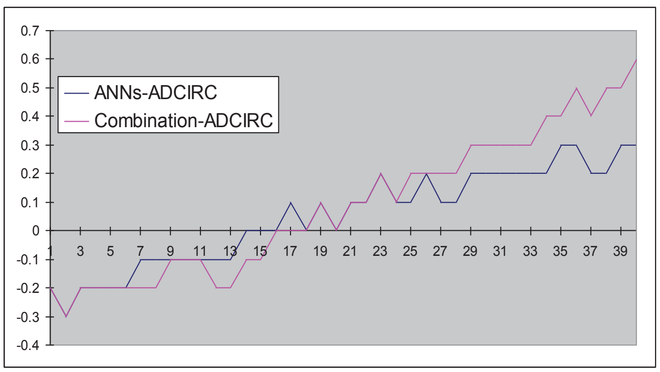


Figure 13. Deviation (in feet) of ANNs-minus-ADCIRC and combination-minus-ADCIRC (x-axis is the number of 50-yr increments in return period) for surge responses (point 141).

3.2 Summary of storm surge runs analysis

This study uses unsupervised ANNs (SOFM) to cluster storm patterns. This is based on four global storm parameters and three locals, or positional, parameters. The angle between the location of interest and the location of storm landfall was found to be the most sensitive input parameter, due in large part to the influence of the Mississippi River delta and levee system in dictating local surge conditions in southeastern Louisiana. The splitting process is able to separate all storm patterns into training and testing components. The number of storms in the testing component equals the number of numerical runs that can be potentially be reduced by simulating surge through the ANNs model using the training components along with their corresponding surge from actual numerical simulations of storm surge using the ADCIRC model. A demonstration project (LACPR), results for a single point in each case, show successful application of the developed computational procedures. Point 141 from LACPR project demonstrates reducing model runs about 40 percent of storm model runs. Results showed that the more storm patterns that are involved in the training component, the higher percentage in the reduction of numerical runs, which makes intuitive sense.

4 Acknowledgement

The Military Hydrology Research Program in the Coastal and Hydraulics Laboratory at the US Army Engineer Research and Development Center funded the watershed similarity study and USACE New Orleans District funded the storm surge runs analysis effort. Permission was granted by the Chief of Engineers to publish this information.

5 References

[1] Chen, L., Chen, C., and Lin, D., "Application of Integrated back-propagation network and SOFM for groundwater level forecasting", J. Water Resources Planning Management", 137(4), 352-365. 2011.

[2] Gomez-Carracedo, M., Ballabio, D., Andrade, J., Fernandez-Varela, R., and Consonni, V., "Application of SOFM to address environmental studies", Chapter 20, Soft Computing Methods for Practical Environmental Solutions, 331- 352, 2010.

[3] Hsieh, B., and Jourdan M..,"Similarity analysis for military application using unsupervised-supervised ANNs", Proceedings of 25th Army Science Conference, Orlando, Florida, 405-410, 2006.

[4] Hsieh, B. B., and Ratcliff, J., "Determining Necessary Storm Surge Model Scenario Runs Using Unsupervised-Supervised ANNs, Proceedings of The 13th World Multi-Conference on Systemics, Cybernetics and Informatics, Orlando, Florid, July 10-13, 334-339. 2009.

[5] Kalteh, A., Hjorth, P., Berndtsson, R., "Review of the self-organizing map approach in water resources: Analysis, modeling and application", Environmental Modeling & Software 23, 835-845, 2008.

[6] Kohonen T., "Self Organizing Maps", Springer Verlag, New York, 1995.

[7] Neurosolutions v6.0, "Developers level for window", NeuroDimensions, Inc., Gainsville, Florida, 2011.

[8] Srinivas, V., Tripathi, S., Rao, A., and Govindaraju, R., "Regional flood frequency by combining SOFM and fuzzy clustering", Journal of Hydrology, 348, 148-166, 2008.

[9] Takatsuka, M., "An application of the self-organizing map and interactive 3-D visualization to geospatial data" 1-9, 2001.

[10] Ultsch, A., Korus, D., and Kleine, T. O., "Integration of neural networks and knowledge-based system in medicine", 5th Conference on Artificial Intelligence in Medicine Europe AIME'95, Italy, 425-426, 1995.

Main Principles of the General Theory of Neural Network with Internal Feedback

D. Pescianschi
Progress Inc., West Bloomfield, MI, US, pro@p-progress.com

Abstract - *In this paper, a new model of formal neuron, analog mechanisms of neuron training, and a new model of biological feedback are proposed. The statement is supported by the neurobiological data published by other authors and through our experiments in silico. Key qualitative and quantitative differences of the proposed neural network model from the concept accepted today are discussed. A new concept reflects the mechanisms of memory formation. The model bridges the gap between the micro-level of the molecular processes in a neuron and the macro-level of information processing and storage in brain. Thus, an opportunity appears of modeling the processes occurring in brain, as well as developing the artificial neural networks, which are trained in a real-time mode, and are not limited in their structure and complexity of connections. The proposed model is easily implemented, both as virtual emulation and by means of digital and analog artificial neural networks.*

Keywords: Neural Network, Training Algorithm, Synapse, Memory, Feedback, Analog Training

1 Introduction

Despite a wide variety of neural network models and their training algorithms developed to date [1], the cornerstone of all models (with small variations) is the concept of a formal neuron (FN) introduced by McCulloch and Pitts (Fig. 1) [2].

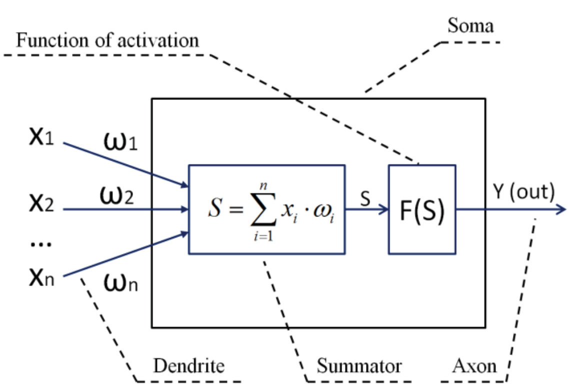


Fig. 1 Formal neuron

$x_1, x_2, \ldots x_n$ – Input signals of a neuron
$\omega_1, \omega_2, \ldots \omega_n$ – Weights of a neuron for input signals
S – Sum of weighted signals
Y – Output signal

FN is the minimum information level for the modern theory of the artificial neural networks (ANNs). This theory does not take into account the processes taking place at lower level, such as molecular one.

It is supposed that the signals in FN are propagated only in a forward direction, namely, from the dendrites through the soma and axon toward the downstream synapses. Besides, in a number of models, for instance a Hopfield network and the Hamming net (reviewed in [1]), such a neuron can have a recurrent connection. In this case an axon (output) forms a synapse with a dendrite of the same neuron (input) or dendrites of upstream neurons. However, the signal direction in the networks remains the same. Moreover, such connections are rather the exception than the rule for a brain structure.

Modeling of the processes taking place in a biological brain was the initial purpose of the elaboration of the ANN theory. However, the model of FN is simplified as compared to the biological neuron. Such an excessive simplification does not allow building realistic models.

One of the least plausible features of known neural network models and training algorithms is their "digital" computing nature. The question that arises is this: who can perform in a brain all these algorithmic calculations for training of a network? Almost in all models, known up to date the neural network represents a passive object of training. Neurons themselves do nothing for the training. This is done by a certain program that is external, in relation to the neural network. Deficiencies in the current approach require looking for alternative solutions.

2 Statement of the new concept

2.1 Locality of processes

According to the theory of Hebb [3], in the absence of the global trainer, only local interaction between neurons is supposed. In this case training is an exclusively local phenomenon covering only two neurons and the synapse connecting them. A global system of feedback for the training of the synapse is not required. In other words, there can be no mechanisms of passive training of neurons (i.e., there are no any external, towards a neuron, programs that can apply information about other neurons). Only the neuron as such can train itself, using for this purpose the information, which is received due to its connections with other neurons (synapses). All processes, occurring in a neuron are localized. Except the information coming through the physical connections, no other source of information about the remaining network is available for the neuron. Based on the acquired information, the neuron itself regulates the activity and is trained by this means. Although there are other mechanisms of mass regulation of neurons, for example, based on hormones, these mechanisms do not provide the process of training as such, and only regulate mass activity of groups of neurons (so called global switches).

The neuron is a completely autonomous self-sufficient minimum adaptive unit of a network. Training of the whole network is composed of the sum of neurons, trained separately. It provides full parallelism of the training process. Thus, both main neural network processes (training and recognition) are parallel by nature.

2.2 Internal feedback on the basis of retrograde signal

According to one of the basic principles of cybernetics, any adaptive system, which includes also a neuron, must have a feedback. If to exclude an external recurrent connection since it is absent at the majority of biological neurons, it is necessary to assume that the feedback is implemented in the neuron in a different way (such as an internal feedback).

Direct connection in a neuron is implemented by the action potential (AP) that is a charge occurring in the direction from a soma to axon terminal and further, through synapses to dendrites of downstream neurons. At the same time a number of the facts allow suggesting that an internal feedback of other nature is also available in a neuron. For instance, the display of presynaptic long-term potentiation (LTP) has been shown in a response to activation of postsynaptic receptors that assumes the existence of the retrograde intermediary in synapse [4, Ch.16] The interconnected microtubules of the cytoskeleton can fulfill a role of the internal feedback, delivering a retrograde signal from the presynaptic axon terminal through the axon and soma to the neuron dendrites. In a neuron, also within the axon and dendrites, the microtubules form the interconnected network covering the whole cell. Microtubules, which may reach several mm in length, interconnect by means of protein-based MAP bridges. Microtubules are connected *via* dendritic spines even with synapses by means of protein-based actinic threads (Fig. 2).

Each microtubule represents the hollow cylinder, which consists of rows of tubulin dimers. Each molecule of tubulin can exist in two conformations. It has been shown that microtubules can work as cellular automata, transmitting complex signals as waves of different states of tubulin molecules electric polarization. In other words, each microtubule is capable to transmit messages at a high rate of speed [5]. Besides, it is suggested that microtubules are also responsible for maintenance and change of synapses intensity.

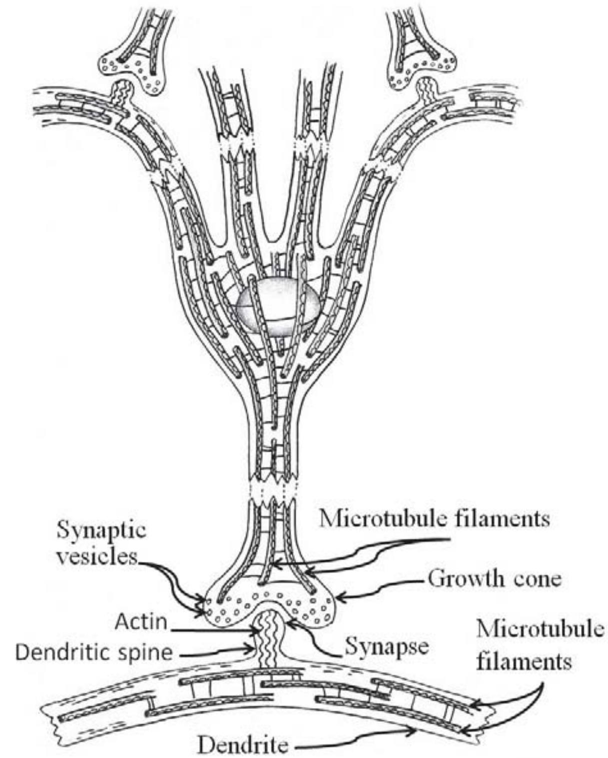


Fig. 2 Network of microtubules in neuron

2.3 Pulse-frequency modulation of output signal of a neuron

As it has been shown [4, Ch.1], AP has a certain threshold above which the amplitude and duration of AP do not depend on stimulation parameters. Separately taken AP basically is not capable to describe fine intermediate states in a wide range. It has been shown [4, Ch.1] that intensity of neuron excitation is coded not by a separate impulse (AP), but by its frequency. The more effective stimuli cause greater depolarization and, as a result, a higher frequency of generation of AP in a neuron. Pulse-frequency modulation is confirmed also by the fact that the frequency of AP has impact on secretion of mediators in a synaptic cleft [4, Ch.15].

Thus, if to assume that a basis of output signal of a neuron is not a separately taken AP, which is actually binary by its nature, but the frequency of AP generation in a neuron, we have rather different function of neuron activation, than a threshold one. Frequency of the neuron can smoothly change from several hertz to kilohertz. Thus, the function of neuron activation is really smooth, continuous and increasing. Notably, the function of activation is not necessary a sigmoid, it can even be a linear function.

2.4 New model of the neuron in equilibrium

2.4.1 Soma, adder, activation function

Soma of a neuron, as it was supposed earlier, performs analog summing of the signals incoming through synapses, so called summation of all dendritic AP. Based on the sum of signals the neuron creates its own excitation level, which is expressed by certain frequency of single APs. Thus, the modulation of monotonously increasing function of activation in the neuron is frequency-encoded. In principle, it can be a sigmoid function, but a type of this function is not important, for instance, it can be a straight line. Natural resistance of dendrites can be described as a vector of non-negative constants:

$C = \{c_1, c_2, \dots c_n\}$,

where n is a number of dendrites.

The signals passing through dendrites can be described as a vector: $A = \{\alpha_1, \alpha_2, \dots\alpha_n\}$.

Then their weighed sum:

$$S = \sum_{i=1}^{n} c_i \cdot \alpha_i \quad (1)$$

Function of neuron activation is a function of this sum.

2.4.2 Synaptic mechanism

In the modern theory of ANN the model of synapse reflects so called Dale's Principle, namely "one neuron — one transmitter" [6]. However to date it is known that in one chemical synapse more than one type of mediator can be released. Moreover, their set is constant for a particular cell. Several tens of various mediators are known. It is very probable that the larger number of mediator substances remains unexplored [4, Ch.15].

Various mediators act synergistically on the same synapse. For emission of mediators with low molecular weight the single impulses are sufficient, whereas for release of neuropeptide cotransmitters pulse trains are often required. Release of each mediator on each synapse can be represented by function of release of $v_i(x_i)$ (Fig. 3).

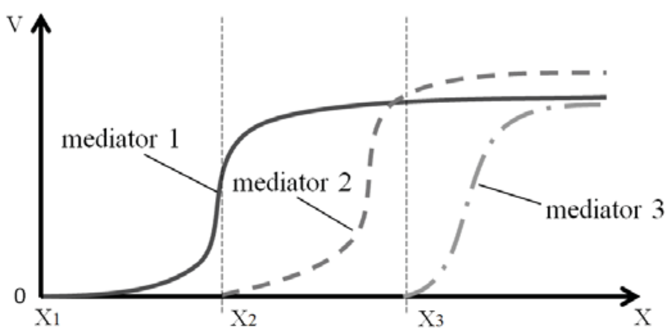


Fig. 3 Release of mediators (V) depending on a signal (X)

x_1, x_2, x_3 – values of signal at which release a new mediator begins.

Thus, a sequence of intervals forms. On each interval a successive type of mediator is joined to the total sum of mediator signals. In other words, each interval corresponds to one certain, just joined mediator or some subgroup of just connected mediators.

A plurality of methods of signal transfer regulation by mediator is revealed. For example, for each type of mediators in a postsynaptic cell exist the receptors that are specific to them. Moreover, the same mediator can have different types of postsynaptic receptors, both excitatory (positive signal transmission), and inhibitory (negative signal transmission). Density of such chemoreceptor in a zone of synapse determines the level of the signal transmitted by mediator, both positive and negative ones [4, Ch.16].

For simplicity of a model let's integrate all stages of regulation of a signal transmission for each j-th mediator in one signal transmission function $\mu_{ji}(v_i)$ that is common for i-th synapse, where vi = vi(xi). Hence, a function of signal transfer by mediator can be represented as the function depending on a signal X: $\mu_{ji}(x_i)$ (Fig. 4a).

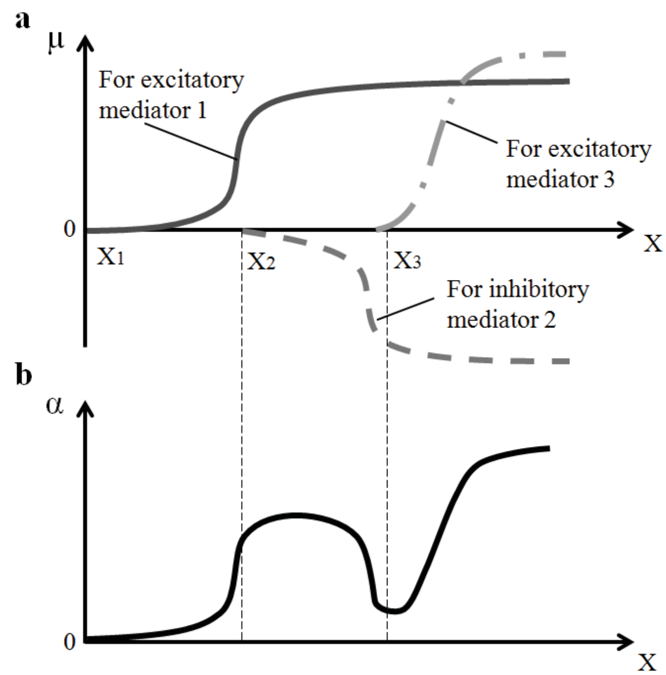


Fig. 4 Mediator functions of signal transfer (a) and synaptic function of signal transfer (b)

X – input signal of synapse
x_1, x_2, x_3 – values of signal at which release of a new mediator begins.
μ – level of signal transfer function by mediator
α – level of synaptic signal transmission function

Just as APs from all dendrites are summed up in soma, the contribution of each mediator to forming of dendritic AP is summed up in a dendrite. Therefore, the total signal transmitted through synapse can be described as follows:

$$\alpha_i = \alpha_i(x_i) = \sum_{j=1}^{m} \mu_{ji}(x_i) \quad (2)$$

where m is quantity of the mediators perceived by receptors on a synapse of i-th dendrite; x_i – output signal of upstream neuron, with which the i-th dendrite has created a synapse, and $\mu_{ji}(x_i)$ – transfer function by j-th mediator of a signal on i-th synapse (Fig. 4b).

Thus, the transmitted signal $\alpha_i(x_i)$ can be described by a vector: $\mu_i = \{\mu_{1i}(x_i), \mu_{2i}(x_i), \ldots \mu_{mi}(x_i)\}$

By substituting (2) in (1) the sum of signals in a neuron is, as follows:

$$S = \sum_{i=1}^{n} c_i \cdot \alpha_i(x_i) = \sum_{i=1}^{n} c_i \cdot (\sum_{j=1}^{m} \mu_{ji}(x_i)) \quad (3).$$

2.4.3 Training as an equilibrium process

As was shown by Hebb [3], the stimulation of presynaptic fiber without support by stimuli from the postsynaptic side leads to synaptic depression, i.e. decrease of postsynaptic potentials in response to presynaptic stimuli. To the contrary, the presynaptic stimuli supported by postsynaptic ones, lead to synaptic potentiation, i.e. increase of potentials.

Taking into consideration a retrograde signal (under stimulation of the postsynaptic side), it may be assumed that direct AP leads to synaptic depression, and the retrograde signal results in synaptic potentiation. Notably, extent of changes both in the first and second case, directly depend on the stimulus size.

It has been shown that AP of a cell appears in a point of axon just behind a soma and extends not only along the axon in the forward direction, but also through the soma to dendrites in backward direction. Retrograde calcium dendritic AP, in turn, can cause local changes of calcium concentration and to influence synaptic transfer [4, Ch.8]. It might be that this mechanism is in the core of synaptic depression under direct AP.

One of the main ideas of current work is the following: the mechanism of counter neuron signals brings synaptic weights to equilibrium state, by reducing and increasing them. This is the basis of training. As it was mentioned above, for training a neuron there should be used only direct signals arriving from upper synapses and retrograde signals coming from lower synapses. It is possible to say that the retrograde signal contains information about the expected level of a neuron excitation for the preset input signals. The nature of the retrograde signals will be a subject of a separate publication.

For a certain synapse the following situations are possible:

1. The input image leads to weak output neuron signal, due to the specifics of its previous training. However, the strong output signal was expected that is expressed in a strong retrograde signal. As it was stated above, weak direct AP results in weak depression of synapses. If the retrograde signal exceeds a direct one, then the level of synapses potentiation will exceed the level of their depression. Next time, in case of the same input image, the excitation level of a neuron will be higher.
2. The input image results in strong neuron output signal. However, the weak output signal was expected that it is expressed in a weak retrograde signal. Thus, a strong direct AP leads to the strong depression of synapses, and a weak retrograde signal results in their weak potentiation. Level of synapses depression in that case will exceed the level of their potentiation. Next time, at the same input image, the excitation level of the neuron is lower.
3. Strong direct AP and strong retrograde signal lead to the strong depression on one hand and to strong potentiation on the other hand, which mutually neutralize each other and leave excitability level at the former (high) level.
4. Weak direct AP and weak retrograde signal are similar to the case 3, with only that difference that the weak depression and weak potentiation mutually neutralize each other, having left a neuron at the former level of weak excitability.

In other words, training of a neuron is the establishment of balance between depression and potentiation of synapses at direct and retrograde signals.

The proposed equilibrium-based model of memory formation correlates well with the rules of Hebb. However, these rules do not work in a situation, when a retrograde signal is stronger than a direct signal that leads to connection strengthening.

The described mechanism is strongly simplified. It seems very probable that the given equilibrium process is unevenly distributed between all synapses of a neuron. For example, it has been shown that the backpropagation of AP from soma depends on the input resistance of different branches (dendrites). The input resistance, in turn, depends on degree of activity of excitatory and inhibitory synapses. Thus, the backpropagation of excitation to dendrites depends on synaptic activity [4, Ch.8].

2.5 The localized equilibrium-based process

The equilibrium-based mechanism of training, described above, works not at the whole synapse, and only at some of its appropriate synaptic mediators.

As it was stated above, different groups of mediators correspond to different levels of a presynaptic signal. It means that the equilibrium-based mechanism of error compensation works for a certain image at a certain group of mediators. For other image the group of mediators will be already different, i.e. different images are characterized by a sets of mediators only partially similar, from one image to another one.

2.6 Training of a neuron

Training of a neuron can be described by the following simple procedure:

1. The input image is fed to the synapses of neuron.

2. Input signals on synapses activate certain groups of mediators.
3. Synaptic weights for the chosen mediators are summed up in a soma creating direct AP.
4. The size of a direct AP determines the depression level of selected mediators (decrease of weights).
5. A retrograde (expected) signal is fed to the output of a neuron causing potentiation of the mediators selected by the input image (increase of weights). And, the level of this potentiation depends on the size of retrograde signal. As a result of direct PD and retrograde signal the neuron comes to equilibrium state, thus being trained in the current image.
6. The next image is fed to inputs and outputs of neuron, and steps 1 – 5 are repeated for the new image.
7. After processing all images (one epoch), the process for these images can be repeated again – until the necessary level of training accuracy is achieved, or until the achievement of the minimum speed of convergence.

In the course of training to a new image, the level of weights forming the previous memory is only partially modified in a new way. At sufficiently high amount of synapses and possible levels of a signal (mediators) the influence of such distortions is extremely small. Moreover, at repeated training the weights are gradually redistributed, by finding state with maximum level of the general balance. Such system itself finds the state of absolute minimum.

3 Conclusions

The new models describing a formal neuron, neural network training and recognition, as well as memory formation are proposed, which are in good correspondence with experimental data. The described model of a neural network is much more similar to the biological neural network, than the ANNs known to date.

The proposed new model of formal neuron and network includes:

- Synapse with a set of mediators (weights) and nonlinear function of signal transmission.
- The internal feedback implementing a retrograde signal transmission in the direction from axon terminal to dendrites and synapses with upstream neurons.

The model of training of this neuron and network includes:

- Selection of the mediators in a synapse, participating in training and recognition. This depends on the input signal.
- Correcting weights on synapses by depression and potentiation at direct and retrograde signals. The value of decrease in synaptic weights directly depends on forward AP of a neuron, and the value of increase of these weights directly depends on retrograde signal.

Experiments [7] with virtual models of the proposed network have shown that the network has the following characteristics:

- The high training speed, which depends linearly on the size of network and data volumes, unlike other ANN models with their exponential dependence. The proposed network requires dozen times less number of training epochs than any classical ANN.
- The scalability that allows building networks of any size and complexity.
- Simplicity in implementation in analog or digital form. This does not require the external trainer in the form of the computer or chip with its long and complex digital calculations [8].

This proposed concept is an important step towards understanding the memory formation mechanisms in a brain. The described ideas are confirmed by virtual experiments, which indicate breakthrough in technologies of processing and storage of information.

4 Acknowledgements

I appreciate fruitful discussions and support by A. Boudichevskaia, V. Proseanic, B. Zlotin, V. Pesceanskaia, S. Visnepolschi, and A. Zusman. I am also very grateful for the support by the Progress Inc. working team and investors.

5 References

[1] S.O. Haykin "Neural Networks and Learning Machines" (3rd Edition), Pearson Education, Inc., 2009

[2] W.S. McCulloch, W.H. Pitts "A logical calculus of the ideas immanent in nervous activity". Bulletin of Mathematical Biophysics, Vol. 5, p. 115-133, 1943

[3] D.O. Hebb "The organization of behavior". Wiley, New York, 1949

[4] J.G. Nicholls, A.R. Martin, P.A. Fuchs, D.A. Brown, M.E. Diamond, D.A. Weisblat "From Neuron to Brain, Fifth Edition", Publisher: Sinauer Associates, 2012

[5] T.J.A. Craddock, D. Friesen, J. Mane, S. Hameroff, J.A. Tuszynski. "The feasibility of coherent energy transfer in microtubules". Journal of the Royal Society, Interface / the Royal Society, 11(100):20140677, 2014

[6] H.H. Dale "Pharmacology and nerve endings". Proc Roy Soc Med 28:319-332, 1935

[7] D. Pescianschi, A. Boudichevskaia, B. Zlotin and V. Proseanic "Analog and digital modeling of a scalable neural network", presented for the current conference.

[8] D. Pescianschi "Neural network and method of neural network training" PCT patent application US1519236, 2015

An Application of Neural Networks to an Autonomous Car Driver

K. Albelihi[1] **and D. Vrajitoru**[1]
[1]Computer and Information Sciences Department, Indiana University South Bend, South Bend, IN, USA

Abstract—*In this paper, we present a car driving system called "Gazelle" for a simulated racing competition. For this application, we used both procedural methods and a neural network capable of learning. We hoped that using neural networks could lead the controller to derive more accurate equations for driving the car based on data acquired in the training process. We also expected that the more the network is trained, the more precisely it will calculate the driving information.*

Keywords: neural networks, intelligent systems, learning

1. Introduction

In this paper we present a study on various methods that can be applied for successfully driving a car in a simulated environment in the presence of opponents. A longer and more detailed version can be found in [1].

The interest in developing autonomous vehicles increases day by day with the purpose of achieving high levels of safety, performance, sustainability, and enjoyment. Driverless cars are ideal to use in crowded areas, on highways, and because they ease the flow of the cars. Autonomous cars can also reduce the chances of occurring accidents caused by other cars or by pedestrians. There are many research centers established around the world for developing smart systems for driverless cars, such as the Center for Automotive Research at Stanford University (CARS) [2].

In this paper we present an adaptive racing car controller called Gazelle developed within TORCS (The Open Racing Car Simulator) [3]. The TORCS system visualizes racing cars with complex graphics based on physics principles [4]. TORCS attracts a wide community of developers and users, and it is the platform for popular competitions organized every year as a part of various international conferences [5]. The program offers a server which implements races combining multiple cars on a variety of tracks. A client module can be written by the user supplying the actions of an individual car. Our driver, called Gazelle, was submitted to the TORCS competition organized by the Genetic and Evolutionary Computation Conference in 2013 [6].

In this environment, each car is controlled by a process that can access the current state of the car in the race, consisting of information about the track, the car, and the opponents [7]. Based on this information the controller makes decisions to modify the following control units:

- the steering wheel with values in the range [-1, +1] for a change in direction: -1 corresponds to -45^o while +1 to 45^o;
- the gas pedal [0, +1] for accelerating; a value of 0 will result in losing speed;
- the brake pedal [0, +1] for decelerating;
- the gearbox with possible values in the set -1,0,1,2,3,4,5,6 for choosing the gear [5].

Figure 1 shows a snapshot of the TORCS application. The upper screen displays the client car and its information such as the car's rank, the total time that the car spent from the beginning of the race, the best time that has been taken to complete a lapse, and other measurements. The lower screen shows the race from another angle that can highlight opponent cars if any are present. We can also see some statistics of the car such as gear levels and the speed of the car.

Fig. 1: A Screenshot of TORCS during the race

The work in this paper is based on the EPIC controller as presented by Guse and Vrajitoru in [5]. Epic was submitted to the GECCO 2009 competition [8]. The Epic driver is based on two components: determining the target angle for turning in each frame, and determining the target speed in the next frame. The controller calculates the target angle based on the free available distance ahead. It also provides a sharp turn detecting system which adjusts the target speed for an approaching sharp turn to keep the car inside the track. It uses Hill-Climbing techniques to adapt the speed parameters to new tracks. However, this controller lacks a component to handle opponents, and the movement along the track requires more fluency. The Gazelle driver improves the Epic code on these two aspects.

Many approaches can be found in the literature for track prediction with the purpose of optimizing the performance. Such an example is the track segmentation approach, in which the track is divided into pieces that are classified as pre-defined types of polygons. Then the controller reconstructs a full track model from these polygons, as presented in [9]. Another controller based on the track segmentation principle is proposed by Onieva et al. [4]. The architecture of the controller consists of simple modules that control gear shifting, steer movements, and pedals positions. In addition, the target speed is adjusted by the "TSK fuzzy system". The most important aspect of this work is the opponent modifier. It controls the driving behavior in situations where an opponent is nearby by adjusting the steering controller and the braking controller immediately.

A more recent work [7] introduces a driving controller called AUTOPIA for the simulated racing car competition. It provides a full driving architecture including six separate main tasks: gear control, pedal control, steering control, stuck situation manager, target speed determination, opponent modifier, and learning module. The paper provides a simple and a powerful architecture especially for the opponent modifier using heuristic rules.

Many learning approaches are used to find the optimal path for the car to reduce the time required to complete the race. An evolutionary learning approach for this purpose is presented in [10]. Here, a self-adaptive evolutionary strategy (SAESs) is used to derive the parameters involved in determining the target speed in an efficient and easy to generalize way. This driver also lacks an opponent handling system. Another controller using an evolutionary learning system is presented in [9]. This controller uses a simple evolutionary learning approach to plan the path ahead for the car.

More recently, another learning approach has used hyper-heuristics in a real-valued mode in [11]. This system approaches the TORCS-based car system as a real valued optimization problem and studies the performance of different methodologies. These include a set of heuristics and their combination controlled by a selection hyper-heuristic framework. The study shows that hyper-heuristics perform well in the TORCS environment.

Artificial neural networks (NN) are also used as a learning system, well recognized by the computer science community and with many applications [12]. In [13], a human-like controller using NN was submitted to the 2010 Simulated Racing Car Championship. The controller builds a model of the tracks using the NNs to determine the trajectory of the car and the target speed. The NNs were trained with data retrieved from a human player. This work shows satisfying results of predicting the trajectory on new tracks; however, the target speed is slower than the human's on the same tracks.

The remainder of the paper is organized in the following way. Section 2 introduces the procedural algorithms in Gazelle. Section 3 presents the application of NNs to compute the target direction. Section 4 shows results from the experiments, and the paper ends with conclusions.

2. Procedural Drivers

We started this research from a procedural driver previously developed in [5] called Epic, submitted to the GECCO 2009 competition [8]. After several improvements that we will describe below, the new driver is called Gazelle. We compare its performance with Epic as well as with a pilot provided by the TORCS software called Simple Driver.

2.1 Epic Driver

The general algorithm of Epic consists in the following steps [5]:

- calculate the target direction and speed,
- determine the correct gear,
- calculate the target angle based on the target direction,
- calculate the acceleration and the brake based on the target angle and speed.

First, for the target direction, Epic starts by deciding if the car can continue to travel in the current direction. If the car is inside the track, close enough to the centerline, and there is enough free distance ahead, then the car can persist in the same direction. Otherwise Epic takes a new direction by modifying the steering angle to get closer to the road centerline.

Second, the target speed is computed. If the car is going almost straight, the free distance ahead is large enough, and no sharp turn is expected shortly, then the car speeds up towards the maximum value. In any other case, a large value for the target speed is set to start with, which is first scaled by the sine of the target angle for steering the angle and with the available free distance in the target direction.

Epic used a simple Hill Climbing technique to adjust several parameters that affect the control units of the car. Thus, the controller uses a dynamic adaptation mechanism to tune the racing car's behavior to a new track. If no damage has been recorded during this first lapse, the parameters used for calculating the maximal speed in each situation are incremented to make the car go faster. Otherwise every time the car gets out of the track or records damage without an opponent being close by, the pilot will keep the same values for these parameters or will decrease them to make its behavior safer.

Epic has several well-developed functions, however, it required some improvements such as handling opponents, enhancing the trajectory stability on the road, and anticipating sharp curves better.

2.2 The Gazelle Controller

The Gazelle controller consists of three components: the target direction unit, the target speed unit, and the opponent adjuster. The target direction unit controls the direction in which the car is moving. The target speed unit adjusts the speed based on the target direction, while the opponent adjuster adjusts the direction and speed based on the opponents' presence. Below we will describe each unit in more detail.

Target Direction Unit

The unit determines the target angle using the following guidelines:

If the current direction of the car is close enough to the road centerline, there is enough distance straight ahead, and the car is safely inside the track, then the car can continue in the same direction.

Otherwise, we start from the direction of the road centerline, and scan by 10 degrees in the direction in which the distance ahead increases, until we find an angle at which it decreases, or we reach the maximal turn angle of $\pm 45^o$.

If the car is too close to the border of the road or gets outside, we add a direction change to move it back inside. Currently, the borders threshold, denoted by $safelyInsideTrack$, is at 85% distance from the center of the road, to account for the width of the car. Let $trackPos$ be the current position of the car on the road, taking values between -1 and 1. If $|trackPos| > safeInsideTrack$, then the new target angle is computed as:

$$-25 * \text{sign}(trackPos)(|trackPos| - safeInsideTrack)$$

where the function sign returns -1 for a negative number, 0 for 0, and 1 for a positive number. This formula scales 25 degrees by how far the car is from the threshold. If the computed target angle already has a value of the same sign but of a larger absolute value, then this new target angle is not used because the normal method is performing the adjustment already.

If the current turning angle is good enough, we maintain it for movement continuity. This is determined by comparing the free distance ahead with the free distance 10 degrees left and right; if the distance ahead is the largest of the three values, then we can maintain the current angle. This is an addition to the Gazelle controller to improve the fluency of the car's movement.

Target Speed Unit

The target speed is computed once we know the target angle. The unit determines the speed using the following guidelines:

If we are going almost straight or on a fast curve, if the distance ahead is large enough, and if no sharp turn is coming ahead, we aim for a configurable high speed parameter called $sundayDriver$.

Otherwise the target speed starting from the $sundayDriver$ value is first scaled directly proportional with the cosine of the target angle for the change in direction and with the available distance in the aimed direction. This way, the smaller the turning angle is, the larger the speed will be. Similarly, the more distance is available ahead, the faster the car will go.

Let $safeSpeed$ be a value for the speed that we think will be safe for any curve, such as 30 km/h. Let $spaceFactor$ be the available free distance in the aimed direction normalized by the maximal sensor range (100m). The speed is computed as:

$$\begin{gathered} targetSpeed = \\ safeSpeed + (sundayDriver - safeSpeed) * \\ \cos(targetAngle) * spaceFactor^2 \end{gathered}$$

The resulting target speed is scaled afterwards by a factor depending on the sharpest turn in the road detected ahead, 20 degrees left and right of the aimed direction. The purpose of this is to anticipate situations where the speed needs to be reduced. Thus, a small difference in the free distance ahead represents a possible sharp turn approaching that requires a slower speed to be taken safely.

The sharp turn detection algorithm and the basic ideas in computing the speed are the similar in Gazelle to Epic, but the practical equations they comprise have been refined.

Opponent Adjuster Unit

We put more efforts into building a component for dealing with opponents because the car's performance can be improved by handling the opponents properly. As we mentioned previously, most of the controllers we discussed in the introduction don't handle the opponents well or at all. Neither the Simple Driver, the controller provided as an example by the TORCS competition, nor the Epic controller can deal with the opponents. In our opponent adjuster, if an opponent violates the chosen tolerance values of closeness as determined by the opponent sensors in each direction, then the gas/brake control and steering control will be modified to avoid the collision the following way:

If there is an opponent at a distance of 200m or less, then a test will determine if it violates the safe distance (the tolerance values) in each of the available sensor directions.

If there is an opponent in the front of the car, on the sides, or in the rear of the car within an unallowable space, the following flags are turned on, causing a reaction of the respective modules:

A *Brake flag* for an opponent in the front. This flag takes care of the sensors in the range of -40^o to 40^o [4]. If an opponent is found within an unallowable space and its speed is close to ours, the car should brake immediately by modifying the brake/accelerate value to half of the current speed. The tolerance values are shown in Table 1 and were adopted from [4].

Orientation of the Opponent Sensor	Tolerance Value
$\pm 40^o$	6m
$\pm 30^o$	6.5m
$\pm 20^o$	7m
$\pm 10^o$	7.5m
0^o	8m

Table 1: Opponents adjuster over the gas and brake action

A *Steering flag* for an opponent in the front or on the side takes care of the opponent sensors in the range of -100^o to 100^o, also adopted from [4]. An overtaking maneuver requires to modify the steering angle if the opponent violates the tolerance values. The tolerance values are shown in Table 2.

Orient. of the Opponent Sensor	Tolerance Value	Increment Value
0^o, $\pm 10^o$	20 m	$\pm 0.20^o$
$\pm 20^o$	18 m	$\pm 0.18^o$
$\pm 30^o$	16 m	$\pm 0.16^o$
$\pm 40^o$	14 m	$\pm 0.14^o$
$\pm 50^o$	12 m	$\pm 0.12^o$
$> \pm 50^o$	10 m	$\pm 0.10^o$

Table 2: Opponent sensors values for overtaking

An *Accelerating flag* for an opponent at the rear of the car driving at an equal or higher speed than ours. Increments values are summarized in the third column of Table 2.

Trouble Spots Register

This component was added in order to avoid accidents caused by mistakes in predicting the right steering angle, leading the car out of the track. In TORCS competitions, the race starts with a training lapse which allows drivers to learn the track. After that, the actual race takes place in the second lapse. Thus, we introduced the "Trouble Spots Register" detecting and storing places in the track where the car gets out of the road starting from the training lapse. In the subsequent lapses of the circuit, to avoid repeating these mistakes, we use a method decreasing the speed whenever the car is close to a trouble spot, by an amount inversely proportional to the distance to the trouble spot.

A list of "trouble spots" on the road will be stored by the Gazelle driver in a persistent memory space in order for it to be accessible at later points during the race. To achieve this, the latest position of the car on the road is stored in each frame. Then when the code detects that the car got out of the road, this position is added to the list.

In each frame, the current position of the car is compared to the trouble spots. If we are close enough to one of them, the speed will be adjusted as mentioned above. The closer we are to the trouble spot, the faster the car will decelerate. The issue arises from the fact that the visibility of the driver is limited to 100m ahead and that it is difficult to break down the speed fast enough if the situation requires it. For this reason we adopted the approach of detecting a sharp turn on a road combined with the trouble spots detector.

3. Neural Network Application

We employed a NN in the Gazelle system to determine the direction of movement of the car. We used a classic perceptron NN with one input layer, two hidden layers, and one output layer. We trained it with the back-propagation algorithm and *tanh* as the neuron activation sigmoid function. We used recommendations from literature [14], [15] and experiments to determine the number of hidden layers and the number of neurons on each of them.

In our model, the input layer consists of 5 neurons for the following input values:

- the last target angle in the previous frame,
- the current angle with the road direction,
- the lateral offset of the car with the center of the road,
- $distance1$ and $distance2$, computed as the difference between the free distance ahead in the direction of movement and the free distance ahead at an angle of $\pm 10^o$. This should indicate in which direction the road is turning.

The output of the NN consists of the target angle for steering the car. We tested two separate models for it: one with a single output neuron, and one with 5 output neurons. For the second model, we divided the output interval $[-34^0, 34^o]$ for the output into 5 intervals: $[-34^0, -17^o)$, $[-17^0, -0^o)$, $[0^0, 17^o)$, and $[17^0, 34^o]$. This is based on the observation that $\pm 34^o$ was the highest value for steering the vehicle in the data we collected for training. We assigned a neuron to each of the values -34^o, -17^o, 0^o, 17^o, 34^o, then we aggregated the output as a sum of the assigned values weighted by the neurons' output.

When training the NN, we first mapped the value of an angle to the closest interval, which identified two neurons with non-zero output. Then we assigned target values between 0 and 1 to these two neurons based on the placement of the angle in the interval.

Data Collection and Training

For this experiment, we trained the NN with data collected from the procedural Gazelle. Thus, while the car is driving with the pilot described in the previous section, we collect the input as described in each frame, and then the output angle determined by this method as intended output for the NN. The model with 5 output neurons does not need separate data collection, as the angles can be mapped to the output for each of the 5 neurons on the spot as needed.

We used three tracks offered by TORCS for our experiments: Alpine2, E-Road, and Forza, of total length 3773.47m, 3260.43m, and 5784.10m respectively. Figure 2 shows the shape of these tracks.

The raw data collected this way is not ready for training. The number of individual points we collected varies between 3765 and 13140, which is too much. Another problem we had to deal with is the uneven distribution of the data. Thus,

Fig. 2: Test tracks: Alpine 2 (left), E-Road (center), and Forza (right)

as the car goes straight for a large portion of each track, data points where the output angle is close to 0 are more numerous than any of the others. This can result in an over-specialization of the NN to these angle values.

The collected data are also non-uniformly distributed from a sequencing point of view. For the same reason as above, data points where the output angle is almost 0 also tend to appear *together* in clusters representing straight stretches of the road. This can also contribute to over-specialization.

The first operation that we performed was filtering the data to have a more even distribution. We started by dividing the interval of observations for the output angle into interval of 0.1 radians length. We isolated one special interval between [-0.01, 0.01] radians. This represents parts of the track where the car goes almost straight. The rest of the interval between -.5 to .6 radians was divided into 11 intervals of a length of 0.1 radians. We added one interval for any angle <-.5 radians and one for any angle >.6 radians. This gave us 14 intervals total.

For each of these intervals, we selected about 50 random data points from each of them. For intervals where the number of observations was less than 50, we kept all of the observations. Figure 3 shows an example of the number of data points in each interval before and after this filtering process. We can see that the middle intervals initially contained a lot more data than the extreme ones. After filtering, the data is better distributed between the intervals.

The second operation performed on the data was to randomize it. We performed a simple fair shuffle on the data before feeding it to the NN for training.

In a first phase, we tested the NN trained with the data collected from each track on the track itself. This testing is called *retrospective*. In the second phase, we selected the set of data that performed the best retrospectively, and tested the NN trained with it on all the tracks. The first evaluation shows if the NN is capable of learning. The second evaluation shows if the NN can learn from one track and then perform well on another one.

During the training process, we stored the weights of the NN in the configuration where the average error is the lowest. We called these the *best weights*. We compared the performance of these settings with the use of the weights in the NN at the end of the training process. The latter are called the *last weights*.

Finally, we tested two modes for training the NN: one

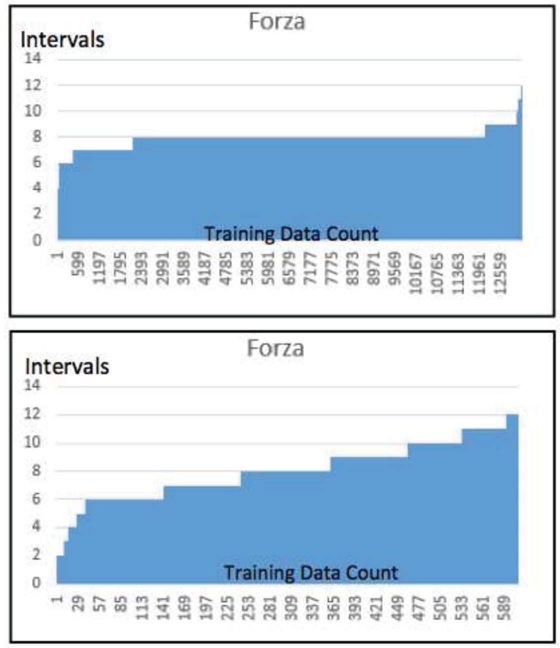


Fig. 3: Training data collected from Forza before filtering (top) and after (bottom)

where we fed the data file to the NN in 100 iterations, and one where we fed it in 500 iterations. The experiments show that the system doesn't learn significantly more in 500 iterations, though the best weights are often recorded between 100 and 200 iterations.

4. Experiments

In this section we present the results of the various models on the 5 tracks with the parameters we described above. For each track, we set the race at two lapses in all the experiments. This allows us to see the effect of any learning process taking place in the first lapse over the performance of the car in the second lapse. An example of such a process is the recording of the trouble spots. We used the value of 100 km/h as the safe speed and 150km/h as the maximum speed for all the experiments, unless specified otherwise.

At the end of the two lapses, the program itself outputs some information, such as the total time and the damage. We will also report some other measures of performance: the total damage to the car and the total distance covered by the end of the race. The more distance is covered in one lapse, the less efficient the driver is.

Table 3 shows a comparison of the procedural Gazelle with the Epic and the Simple drivers. The table contains the total time to finish two lapses (in seconds), the total distance covered by the car (in meters), and the total damage to the car. A time marked as N/A means that the car did not finish the two lapses. The races are configured to terminate early if the car takes longer than 12 minutes to finish, or if the damage is too high.

Track	Driver	Time (s)	Distance (m)	Damage
Alpine2	Simple	7:08	7573.55	0
	Epic	6:44	7574.1	0
	Gazelle	5:33	7571.96	3314
E-Road	Simple	5:38	6547.46	0
	Epic	5:09	6546.23	0
	Gazelle	3:36	6546.05	0
Forza	Simple	6:14	5783.91	0
	Epic	N/A	5783.73	2186
	Gazelle	2:42	5784.09	1051

Table 3: Procedural Gazelle results on 3 tracks, single car

From this table we can tell that the procedural Gazelle is an improvement over Epic and the Simple Driver. It completed all 3 tracks faster than the two other drivers and with a slightly more efficient trajectory, based on the total distance. Although the damage was rather high for the Alpine2 track, it is still within the allowed amount.

The next experiment shown in Table 4 tested the three drivers in the presence of opponents on the road. For this purpose we added 3 drivers provided by the TORCS environment: berniw1, lnfHist1, and inferno10. These opponents have various levels of performance: high, medium, and low respectively, as specified in the TORCS manual [16].

Track	Driver	Time (s)	Dist. (m)	Damage	Rank
Alp2	Simple	3:06 + 1 lap	4379.32	3169	4/4
	Epic	3:06 + 1 lap	4394.89	1634	4/4
	Gazelle	3:03 + 1 lap	5919.3	3250	4/4
E-Rd	Simple	3:28	6548.62	403	4/4
	Epic	3:29	6648.71	139	3/4
	Gazelle	2:17 + 1 lap	4347.97	0	4/4
Forza	Simple	2:40 + 2 laps	2745.69	0	4/4
	Epic	2:40 + 2 laps	2745.78	0	4/4
	Gazelle	2:46 + 1 lap	5809.10	0	4/4

Table 4: Procedural Gazelle, race with 3 opponents

In this table, the total time shown in each case is the time when the race was finished. Races with multiple cars are terminated when all but one car have finished the two lapses, so that the complete ranking of the cars can be determined. The time shows how many lapses the car had not completed by the time the race was finished.

On Alpine all 3 drivers finished last and did not complete the last lapse. However, Gazelle managed to cover a much longer distance than the Simple Driver and Epic before the race was finished. The higher amount of damage could indicate that this car kept closer to the opponents and was damaged more by interacting with them. On E-road Gazelle did worse than the two other drivers, but with no damage. On Forza, Gazelle outperformed the two other drivers: it completed one lapse before the race finished, while the others didn't, and it covered about twice as much distance. Overall, we can conclude that Gazelle performs better than Epic and Simple Driver in the presence of opponents.

The next set of experiments shows the results obtained by the Gazelle using the NN for the target direction. Table 5 starts with the retrospective evaluation where the NN is trained for each track with the data collected from that same track. In this table, the time shown as "+1 lap" means that the race was terminated before the last lapse was completed, because the car ran out of time.

Track	Output	Weights	Time (s)	Dist. (m)	Damage
Alpine2	1	last	8:13	7572.02	0
	1	best	10:08	7572.78	510
	5	last	9:02	7572.1	24
	5	best	8:13	7573.58	0
E-Road	1	last	+1 lap	439	2133
	1	best	6:24	6546.11	0
	5	last	+1 lap	375.207	1454
	5	best	6:22	6545.95	40
Forza	1	last	+1 lap	2771.34	0
	1	best	+1 lap	9682.41	148
	5	last	6:03	9751.94	85
	5	best	11:44	11595.5	0

Table 5: Retrospective NN testing, 100 training iterations

Based on this table, we selected the data collected from the Alpine2 track, as the NN trained with it performed well more consistently than the others. Table 6 shows the results of this data set in 500 training iterations. As we can see, training the NN with additional iterations didn't seem to improve its performance.

Track	Output	Weights	Time (s)	Distance (m)	Damage
Alpine2	1	last	9:12	7572.8	1065
	1	best	10:00	7573.33	840
	5	last	9:26	7572.76	1604
	5	best	10:41	7572.28	0

Table 6: Retrospective NN testing, 500 training iterations

The data collected from the Alpine2 track is probably better for training because this track is particularly challenging by nature, presenting many curves of various difficulty in both directions. Figure 4 shows a close-up of this track, together with a snapshot of the car on this track.

Fig. 4: A screenshot of the Alpine2 track (left) and the car driving on it (right)

In the next evaluation stage, we have chosen the data file that showed the most promise: Alpine2, and tested the NN trained with it on all the tracks. Table 7 shows the results of the NN trained with the data from the Alpine2 file in 100 iterations, with 5 output neurons, and using the best weights model. These settings showed the best performance in the retrospective evaluation. For easier comparison, we also added the results on the Alpine2 track itself, and the results without the NN.

Track	NN	Time (s)	Distance (m)	Damage
Alpine2	yes	8:13	7573.58	0
	no	5:33	7571.96	3314
E-Road	yes	8:31	6547.28	0
	no	3:36	6546.05	0
Forza	yes	+1lap	5150.39	1
	no	2:42	5784.09	1051

Table 7: Results of the NN trained on Alpine 2, 5 output neurons, 100 iterations, best weights

These results show that our NN can learn indeed to drive the car from the data we collected, but the performance is not as good as the procedural algorithm. However, the observed driving style of the car with a NN is smoother and less prone to sudden changes in direction than the procedural methods. With a mechanism that allows it to continue training while being in use, we hope to eventually achieve better performance than the procedural method on all aspects.

The procedural Gazelle has been submitted to the TORCS competition organized as part of GECCO 2013 [6]. We passed the qualification round and ended up in 7th place in the final round.

Future Work

We intend to extend this research in two directions. First, we are currently working on developing some methods to allow the NN to continue training during the race, so that it can better adapt to new tracks. This involves some measures to let the driver know when the trajectory is not ideal, as for example, when exiting the road, or when colliding with the road shoulders. It also means that we must be able to estimate a better direction at that moment and feed it to the NN so it can train with it. A second direction involves collecting better training data, so that the NN has a chance to outperform the procedural algorithm. Some possible better sources of data would be either data collected from humans driving the car, or data collected by observing other pilots. It is also possible to use a trajectory optimization algorithm such as the one suggested in [17].

5. Conclusion

After the experiments we presented, we can conclude that Gazelle is an improvement over Epic because the Gazelle was able to handle the opponents efficiently and it succeeded in avoiding the damage caused either by colliding with other opponents or by hitting the hard shoulders on the side of the road. Also, we can conclude that the NN can learn to drive the car efficiently and is adaptable to new tracks of any type or any shape. Even though the Gazelle took a longer time to complete the race with the NN than with the procedural methods, we can adopt a higher value for the maximum speed for the next competition to improve the total time.

Our experiments showed that it is not always useful to train the NN a lot more than 100 iteration with one set of data. In addition, using the weights in the network sampled when the lowest value of the error is achieved is better. Dividing the output value into output for several neurons proved efficient, as each output neuron has less to learn. Finally, the quality of the data for training and the proper processing of this data is also important for a good performance of the NN.

References

[1] K. Albelihi, “The gazelle adaptive racing car pilot,” Master’s Thesis, Indiana University South Bend, 2014.

[2] “The center for automotive research at stanford,” http://me.stanford.edu/groups/design/automotive/.

[3] D. Locaiono and L. Cardamone, “Simulated racing car championship competition software manual,” April 2013.

[4] E. Onieva, D. A. Pelta, J. Alonso, V. MilanÃl’s, and J. PÃl’rez, “A modular parametric architecture for the TORCS racing engine,” in *Proceedings of the IEEE Symposium on Computational Intelligence and Games*, 2009, pp. 256–262.

[5] C. Guse and D. Vrajitoru, “The epic adaptive car pilot,” in *Proceedings of the Midwest Artificial Intelligence and Cognitive Science Conference*, South Bend, IN, April 17-18 2010, pp. 30–35.

[6] C. Blum, Ed., *Proceedings of the Genetic and Evolutionary Computation Conference (SIGEVO)*, Amsterdam, The Netherlands, July 6-10 2013.

[7] E. Onieva and D. A. Pelta, “An evolutionary tuned driving system for virtual racing car games: The autopia driver,” *International Journal of In telligent Systems*, vol. 27, no. 3, pp. 217–241, Oct 2012.

[8] G. Raidl, Ed., *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, Montreal, Canada, 2009.

[9] J. Quadflieg and M. Preuss, “Learning the track and planning ahead in a racing car controller,” in *Proceedings of the IEEE Conference on Computational Intelligence and Games (CIGâĂŹ10)*, Copenhagen, Denmark, August 18-21 2010, pp. 395–402.

[10] T. S. Kim and J. C. Na, “Optimization of an autonomous car controller using a self- adaptive evolutionary strategy,” *International Journal of Advanced Robotic Systems*, vol. 9, no. 73, pp. 1–15, 2012.

[11] M. Kole and A. Etaner-Uyar, “Heuristics for car setup optimisation in TORCS,” in *Proceeding of IEEE Symposium on Computational Intelligence and Games,*, Edinburgh, UK, 2012, pp. 1–8.

[12] P. N. S. Russell, *Artificial Intelligence: A Modern Approach*, 3rd ed. Prentice Hall, 2009.

[13] J. Muñoz and G. Gutierrez, “A human-like torcs controller for the simulated racing car championship,” in *Proceedings of the IEEE Symposium on Computational Intelligence Games (CIG)*, Copenhagen, Denmark, August 18-21 2010, pp. 473–480.

[14] A. Blum and R. Rivest, *Advances in Neural Information Processing Systems*. San Mateo, CA: Morgan Kaufmann, 1989, ch. Training a 3-node neural network is NP-complete, pp. 494–501.

[15] *Applying Neural Networks: A Practical Guide*. Academic Press, 1996.

[16] D. Loiacono and L. Cardamone, “Simulated racing car championship competition software manual,” April 2013.

[17] E. Velenis and P. Tsiotras, “Minimum time vs maximum exit velocity path optimization during cornering,” in *Proceedings of the 2005 IEEE International Symposium on Industrial Electronics*, Dubrovnic, Croatia, June 2005, pp. 355–360.

Analog and Digital Modeling of a Scalable Neural Network

D. Pescianschi[1], A. Boudichevskaia[1,2], B. Zlotin[1], and V. Proseanic[1]
[1] Progress Inc., West Bloomfield, MI, US, pro@p-progress.com
[2]Darmstadt University of Technology, Applied Plant Science, Darmstadt, Germany

Abstract - *Proposed are the new types of fast training, scalable analog and digital artificial neural networks (p-networks) based on the new model of formal neuron, described in [1]. The p-network includes synapses with a plurality of weights, and devices of weight selection based on the intensity of the incoming signal. Versions of the p-networks are presented that are formed with resistance elements, such as, memristor elements. Also described are the matrix methods of training and operation for the proposed network. Training time for the new network is linearly dependent on the size of the network and the volume of data, in contrast to other models of artificial neural networks with the exponential dependence. Thus, p-network training time is dozens time faster than training time of the known networks. The obtained results can be applied in existing artificial neural networks, and in development of a neural microchip.*

Keywords: Neural Network, Analog, Memristor, Matrix, Training Algorithm

1 Structure of the p-network

The new model of the formal neuron (hereinafter - p-neuron), which is the core of the p-network is based on the following principles:

- Using multiple mediators in each synapse. In p-network the role of mediators may be carried by elements that are called corrective weights, which can be physically represented by electrical resistance, conductivity, voltage, electric charge, magnetic property, or other physical matter.
- Selection of corrective weight for each incoming signal at the synapse should be based on the value of the signal.
- The function of activation is not necessarily described by the sigmoid. Moreover, the output signal can be represented even by a simple sum of signals entering the neuron.
- Correction of the p-neuron weights – i.e., training is provided by not gradually correcting weight values for one image after another (gradual gradient descent), but with a one-step operation of error compensation during the retrograde signal. This takes into account only the information received by neuron in training from its synapses. However, the state of other neurons is not taken into account. Training the network to the next image does not depend on its training to the previous image. For each image used in training the complete compensation of training error is provided.
- Each neuron weight correction is provided by counter signals: the direct signal resulting from recognition by neuron of input image and the retrograde signal represented by the expected output. Correction of weights in analog form is provided as follows:
    1. Direct signal obtained during recognition reduces the value of weights selected at the synapses in proportion to the magnitude of this signal.
    2. Retrograde signal (the expected output signal) supplied to the output of a neuron increases the value of weights selected at the synapses in proportion to the magnitude of this retrograde signal.
- Independent correction of the weights of each neuron allows for complete parallelization of network training.

The above principles are the basis for the development of new training algorithms and allow creation of new p-network properties.

The Fig. 1a presents the proposed p-neuron. In the p-neuron the input signal reaches the device, which assesses the value of the signal and selects one of corrective weights corresponding to the value of this signal. The role of such a device can be performed, for example, by a demultiplexer. The Fig. 1a shows that the device selects the corrective weight 3 corresponding to the value of the input signal I. There can be a variant, wherein the selection of several corrective weights from available weights is provided.

By using the proposed p-neurons the network can be built with any desired topology, including topologies of classical neural networks that are built based of formal neurons. Fig. 1b a network including the proposed p-neurons – i.e., the p-network.

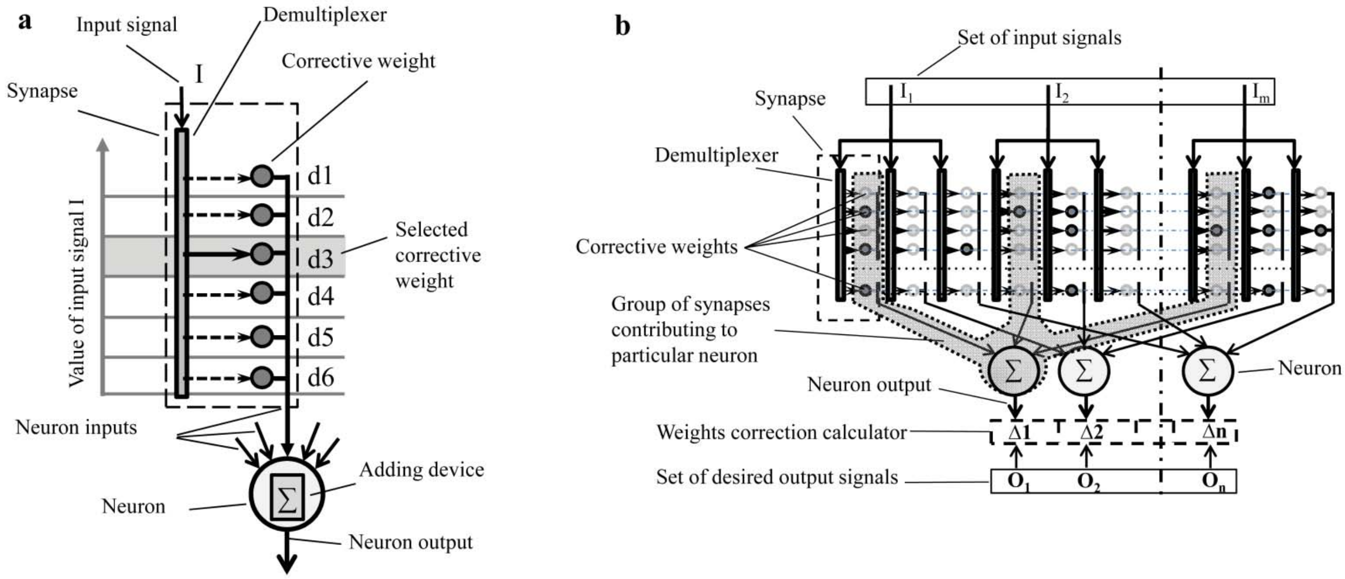


Fig. 1. Proposed p-neuron (a) and p-network including p-neurons (b)

2 Analog modeling of the p-network

Biological network is completely analog by nature and its training and recognition mechanisms are accordingly analog. An example of artificial modeling of analog p-network could be the development of a p-network based on resistor elements, for example, memristors.

Each synapse stores information not in one weighting element (weight), but in a set of weights, each of which corresponds to a certain level (range of values) of the presynaptic signal. Resistors that are connected in parallel can be used as weight elements. The signals in this case would be coded by the currents in the circuits. Parallel connection of resistors provides the automated analog summation of signals in a neuron via summation of the currents on the conductor.

For each synapse it is necessary to provide two sets of weighting resistors: excitatory and inhibitory. Each such set of resistors is connected to the summing circuit of its own. Inhibitory signals are subtracted from the excitatory signals and result in the neuron output signal. Thus, to store not only positive but also negative resistance values, the circuit of the resistors should be bipolar. Otherwise active resistors would have been required to be capable of receiving negative values.

2.1 Recognition

The circuits of the upper three lines (Fig. 2a) have opposite sign to the circuits of the lower lines. Input signals X_1, X_2, X_m enter the control inputs of the device of choice, for example, the control inputs of the demultiplexers DMX_1, DMX_2, DMX_m, which select the circuits that become active (shown in bold). The power circuit A completes the selected circuits. The rest of the network circuits are disconnected. Thus, the set of parallel connected resistors are formed that are the selected weights of neurons. The currents are summed and form the outputs signals of neurons Y_1, Y_2, Y_n.

2.2 Training

The memristors can be used as weighting resistors. Correction of memristor resistors is provided via applying voltage pulses to the same circuits that are used for adding neuron signals. In other words, training of the memristor-based p-network can be provided in the same way as it happens in biological networks – by direct and retrograde signals. The network in Fig 2b is trained in two stages: recognition and training.

During recognition, memristors work as simple resistors (Fig. 2a). The input signals X_1, X_2, X_m are sent to control inputs of the demultiplexers DMX_1, DMX_2, DMX_m, which complete the circuits they have selected depending on the signal (indicated in bold). At the outputs the neurons' current output signals Y_1, Y_2, Y_n are generated.

In the training mode network is trained in the same way as its biological prototype – by equilibrium process between the direct and retrograde signals. The immediate correction of weights is provided by voltage pulse U (y) that proportionally dependents on the signal y. As shown in Fig. 2b memristors in bipolar circuits have counter orientation. Thus, the training impulse increasing, for example, the resistance of the exciting circuit (reducing positive weights), at the same time reduces the resistance of inhibiting circuits (increases negative weights), and vice versa.

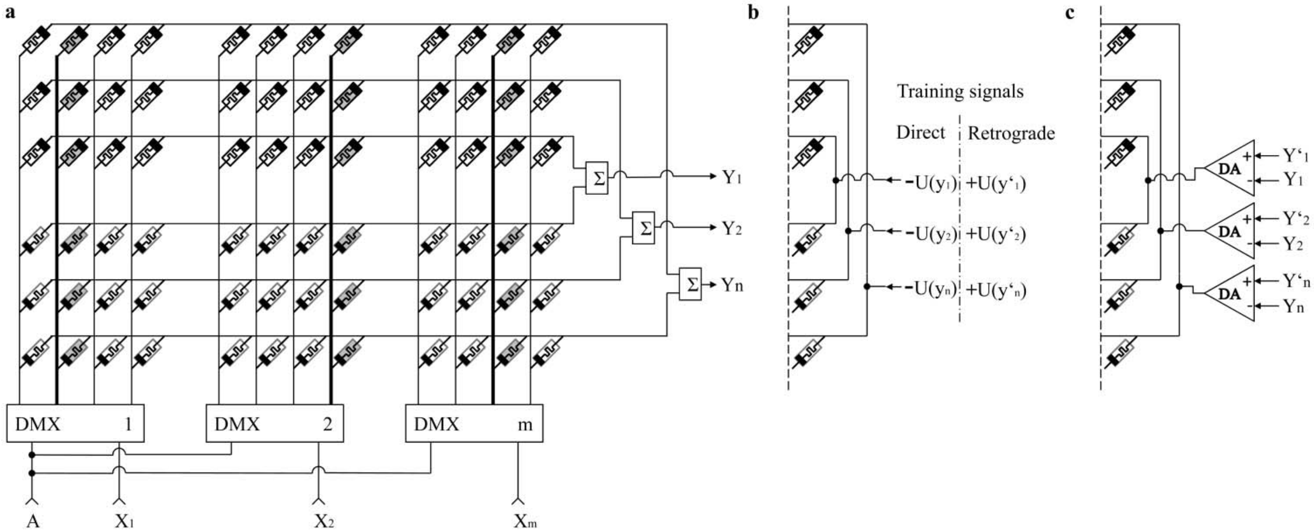


Fig. 2. P-network built on resistors or memristors. Fig. 2a. P-network built on resistors or memristors for image recognition. Fig. 2b. P-network on memristors that is trained by sending direct and retrograde signals in turn. Fig. 2c. P-network on memristors that is trained by sending direct and retrograde signals simultaneously. Fig. 2b and Fig. 2c present only fragments that make the corresponding variants different from the p-network presented in the Fig. 2a.

As in the biological prototype the direct training signal leads to weight reduction (synaptic depression), i.e. - to increased resistance in memristors of the excitatory circuit and to reduced resistance in memristors of the inhibiting circuit. For this purpose, the pulses - U (y_i) are applied to the neuron circuits, wherein, y_i is the output signal of the corresponding neuron.

Retrograde training signal, as in the biological prototype, leads to the weight increase (synaptic potentiation), i.e. - to reduced resistance in memristors of the excitatory circuit and to increased resistance in memristors of the inhibiting circuit. For this purpose, the neuron circuits are supplied with pulses + U (y'_i), wherein, y'_i – is the expected output signal of the corresponding neuron.

Correction of weights (memristors' resistances), at training occurs only in complete circuits, i.e., in circuits already selected by demultiplexers DMX_1, DMX_2, DMX_m. This corresponds to the training of synapses only with selected mediators in natural networks.

2.3 Training optimization

The described mechanism (Fig. 2b) requires separation in time of the direct and the retrograde training signals. To combine these two steps (in order to increase the training speed), a differential amplifier (DA) can be used, one input of which is the current output signal of the neuron (Y_i), and the second input - the expected output signal of the neuron (Y'_i) (Fig. 2c).

This DA generates an output voltage proportional to the difference between the actual and the expected outputs, which is a measure of training error. In the training mode the pulse output voltage of the DA is sent to the memristor circuit, which leads to a change in memristor resistance. Moreover, the higher the error, the higher is the voltage, and therefore, the higher are the changes in memristor resistance. The voltage polarity of such pulse depends on the sign of the error.

The error determines the voltage of correction pulse. Thus, the higher the neuron error, the higher is the voltage at the output of DA, the stronger are the changes at memristor circuit and, thus, the faster is the approach to precise training, i.e., absence of error. Training pulses are repeated until the predetermined threshold of training precision is not reached Fig. 3).

Fig. 3. The process of neuron iteration training

In addition to the conventional features of memristor chips, such as low power consumption and energy independence, memristor p-network has a number of new features:

- Analog recognition and training processes provide:
  - Significant increase in computing speed
  - The ability to store large volumes of data. It is well known that memristors allow storing resistance ranging from several ohms to several mega ohms, i.e., one memristor can store real numbers (several bytes of information).
- Increased yield of reliable microchips with p-network during their manufacture comparing to existing chips: Artificial neural network (ANN) does

not use damaged connections during its training and can use memristors with deviated parameters without compromising the chip operation. Therefore, ANN is not very dependent on the imperfections of the manufacturing process.

- Durability of memristor neuro-chip: P-network ANN has substantially more weights on each synapse than the known ANNs. The information is distributed among a plurality of weights, and, therefore, loss or modification of the individual weights cause negligible distortion of information. Thus, some failures in memristor and connections in the process of application are compensated by other elements.
- Fast recover of memristor neuro-chip due to retraining of a defective microchip. Training speed of the proposed network enables training in real time.
- Resistance to the low accuracy of weight recording in individual memristors. P-network compensates such inaccuracy during training process due to the large number of elements, its architecture and training algorithm.
- Parallel memory operation: in contrast to the conventional method of memory operation, when the information can be read from individual nodes and written only to the exact node addresses, step by step, extracting and recording information in the memristor p-network is provided in parallel and without any node addresses. This allows processing in one step of the dozens' times larger volumes of information than it is possible in the digital address memory. Thus, it also leads to increased speed.
- Conversion of memory from a device for storing information into a device for both: information storage, and processing. Moreover, unlike the sequence digital processing on the CPU and controllers, the process of parallel processing of information is implemented in the p-network. Thus, additional operations for the transfer of data between the CPU and memory are not required. Therefore, data processing is much faster.

3 Digital modelling of the p-network

It is easy to provide not only analog but also digital modeling of the p-network.

Moreover, its digital model can also process information in parallel.

The digital p-network for single and multi-processor systems can be described with the help of matrix algebra.

In particular, the array of memristor elements of Figure 2b can be represented as a two-dimensional matrix

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1k} \\ w_{21} & w_{22} & \dots & w_{2k} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nk} \end{bmatrix} \quad (1)$$

with dimensions of n x k, where n – the number of neurons (outputs), and k - the number of weights in a neuron.

The signals in the circuits after de-multiplexers DMX_1, DMX_2, DMX_m can be represented as a binary matrix of one row $I = \begin{bmatrix} i_1 & i_2 & \dots & i_k \end{bmatrix}$, with dimensions of 1 x k, i.e., as a line of ones and zeros, where the ones correspond to the selected complete circuits and zeros – to the rest of the (disconnected) circuits.

The vector of output signals can be represented by a matrix of one column

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} \quad (2)$$

with dimensions of n x 1.

3.1 Recognition

Recognition with the p-network is the summation of the matrix W elements for each row (neuron), and, only for active (selected) columns, which correspond to ones in the matrix I. Thus, the output image Y can result from multiplication of weight matrix W by the transposed matrix of input image I^T consisting of one column:

$$Y = W \times I^T = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1k} \\ w_{21} & w_{22} & \dots & w_{2k} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nk} \end{bmatrix} \times \begin{bmatrix} i_1 \\ i_2 \\ \dots \\ i_k \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} \quad (3)$$

Batch recognition can be provided, that is, the recognition of a set of images at once. For this purpose, a number of input images can be presented as a matrix I with dimensions of v x k, where v – the number of recognizable images. Each row of the matrix I is a single image, subjected to recognition.

Thus,

$$Y = W \times I^T = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1k} \\ w_{21} & w_{22} & \dots & w_{2k} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nk} \end{bmatrix} \times \begin{bmatrix} i_{11} & i_{21} & \dots & i_{v1} \\ i_{12} & i_{22} & \dots & i_{v2} \\ \dots & \dots & \dots & \dots \\ i_{1k} & i_{2k} & \dots & i_{vk} \end{bmatrix}$$

$$= \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1v} \\ y_{21} & y_{22} & \dots & y_{2v} \\ \dots & \dots & \dots & \dots \\ y_{n1} & y_{n2} & \dots & y_{nv} \end{bmatrix} \quad (4)$$

that is, multiplication of the matrix W, with dimensions of n x k, by the transposed matrix I^T, with dimensions of k x v, produces the matrix Y, with dimensions of n x v, containing the required sums of selected elements in the rows of the weight matrix W for all recognizing images. Each column of the matrix Y is a single output image obtained by the recognition of the corresponding column of the matrix of the input images I.

3.2 Training

As described above, during training with the next image the retrograde signal completely compensates for the error, in the same way every time and uniformly (by the same impulse), thus, correcting all the selected weights of the neuron. In digital mode, the total error of the neuron is distributed between all selected weights of a neuron. That is, to obtain the correction value for each selected weight of the neuron it is necessary to calculate the total error for the neuron. Then, the resulting error is divided by the number of selected neuron weights to provide the correction value for the selected weights.

Error matrix E of the same dimensions as the matrix of the output image Y, is calculated, as follows:

$$E = Y' - Y = \begin{bmatrix} y'_1 \\ y'_2 \\ \dots \\ y'_n \end{bmatrix} - \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} = \begin{bmatrix} y'_1 - y_1 \\ y'_2 - y_2 \\ \dots \\ y'_n - y_n \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ \dots \\ e_n \end{bmatrix} \quad (5)$$

where Y' – the matrix containing the image expected as the result of training, and Y - the matrix of the real output image. Matrixes E, Y' and Y have the same dimensions.

Matrix of corrections (D), which contains the value of the necessary correction for each selected element of the matrix W, for each of the rows of the matrix (each neuron), is calculated by dividing each member of the matrix E by the m:

D = E / m, where m - the number of selected columns of the matrix W for the image (the number selected weights for a single output).

$$D = E / m = \begin{bmatrix} e_1 \\ e_2 \\ \dots \\ e_n \end{bmatrix} / m = \begin{bmatrix} e_1 / m \\ e_2 / m \\ \dots \\ e_n / m \end{bmatrix} \quad (6)$$

Where the error of each neuron is divided by the value of m.

To correct each selected element of the weight matrix W by the corrective value from the respective row of the matrix D, one should create the correction matrix C via multiplying the correction matrix D by the matrix of input image I.

$$C = D \times I = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_n \end{bmatrix} \times \begin{bmatrix} i_1 & i_2 & \dots & i_k \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1k} \\ c_{21} & c_{22} & \dots & c_{2k} \\ \dots & \dots & \dots & \dots \\ c_{n1} & c_{n2} & \dots & c_{nk} \end{bmatrix} \quad (7)$$

The matrix C has dimensions of n x k, as the weight matrix W; each element in each row of the matrix C is equal to 0, if it is in the unselected column, and is equal to an element of the matrix D of the corresponding row, when it is in the selected column. The selected column of the matrix W - is the column corresponding to the element of matrix I equal to one. The unselected column - is the column corresponding to the element of the matrix I equal to zero. Weight correction (training) is performed by adding the matrixes W and C, resulting in the matrix of corrected weights W':

$$W' = W + C = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1k} \\ w_{21} & w_{22} & \dots & w_{2k} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nk} \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1k} \\ c_{21} & c_{22} & \dots & c_{2k} \\ \dots & \dots & \dots & \dots \\ c_{n1} & c_{n2} & \dots & c_{nk} \end{bmatrix}$$

$$= \begin{bmatrix} w_{11}+c_{11} & w_{12}+c_{12} & \dots & w_{1k}+c_{1k} \\ w_{21}+c_{21} & w_{22}+c_{22} & \dots & w_{2k}+c_{2k} \\ \dots & \dots & \dots & \dots \\ w_{n1}+c_{n1} & w_{n2}+c_{n2} & \dots & w_{nk}+c_{nk} \end{bmatrix}$$

$$= \begin{bmatrix} w'_{11} & w'_{12} & \dots & w'_{1k} \\ w'_{21} & w'_{22} & \dots & w'_{2k} \\ \dots & \dots & \dots & \dots \\ w'_{n1} & w'_{n2} & \dots & w'_{nk} \end{bmatrix} \quad (8)$$

Thus, p-network is trained to one image in a single operation. The whole process of training a network to one image can be described by the formula:

$$W' = W + (((Y' - Y) / m) \times I) \quad (9)$$

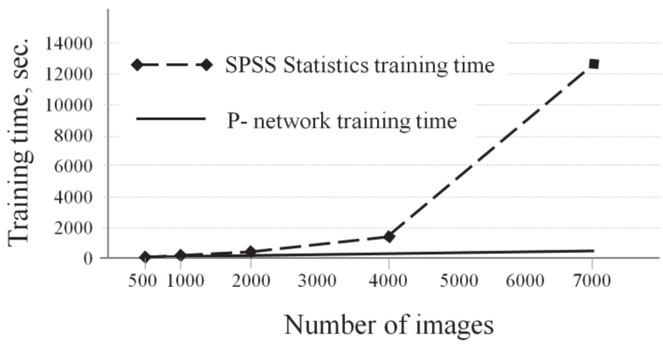
The same training operations are performed for all the images from the training set. The cycle including all the images is the training epoch. If the error level after one epoch is still too high, the training cycle for all the images is repeated.

Training and operation of multilayered networks have their own characteristics and need to be considered in a separate publication.

4 Test results

Experimental p-network, built on the given algorithm, has been developed as a single-threaded program. Testing was performed with laptop Dell Inspiron 5721, Intel CORE i5 1.80 GHz, Windows 7, by comparing the p-network with classical neural networks NeuroSolution and IBM SPSS Statistic 22. Tests were provided with the same data.

Training parameters were selected as follows: 1000 inputs, 20 outputs and 500 to 7000 images (records)



Network	Images	Training time, sec.
Progress P-network	7000	4
IBM SPSS Statistic 22	7000	13400 = 3hour 43minutes

Fig. 4. Comparison of p-network and conventional ANN IBM SPSS Statistic 22

Test results are shown in the Figure 4. As seen in the Fig. 4, when the number of images is around 7000 the p-network is 3250 times faster. With increase in number of records the IBM SPSS Statistics 22 increases training time exponentially. The same increase in p-network increases training time linearly.

The tests have been conducted with the multithreaded versions of the p-network. In particular, the GPU version of the software was developed for running on video cards from NVIDIA supporting CUDA. With the GPU the 100% paralleling of the training and recognition processes was demonstrated. The linear speed increase has been demonstrated with the growth of the number of GPU. The increase is tens of times per GPU compared to the single-threaded version.

5 Conclusions

1. Proposed are the fast training scalable analog and digital models of a new type of artificial neural network (p-network), described in [1].
2. Presented are the analog and digital versions of networks formed with resistance elements, and in particular, with memristor elements.
3. The proposed networks include synapses with a plurality of weights, and devices of weight selection depending on the intensity of the incoming signal.
4. Presented are the matrix methods of training and operation for the proposed network.

This network provides:

- High speed training, due to multiple weights on each synapse and due to the new training algorithm.
- Training time is linearly dependent on the size of the network and the volume of data, in contrast to other models of ANN with the exponential dependence. The proposed network requires many times smaller number of training epochs than any classic ANN.
- Scalability, which allows building such networks of any size and complexity.
- Ease of implementation in the form of analog or digital circuits requiring no "external trainers" in the form of a computer, or a chip, which provide long-term and complex calculations.
- Batch processing of images, which significantly improves performance.

The proposed network is complementary to the memristors technology in creation of a highly reliable neural microchip. P-network also compensates for inaccuracies of manufacture and for unreliable operation of such microchips.

6 References

[1] D. Pescianschi, “Main principles of the general theory of neural network with internal feedback”, presented for the current congress.

Acknowledgements

We appreciate useful advice and support by S. Visnepolschi, A. Zusman, G. Peschanskiy. We are thankful for the support by the Progress Inc. working team and investors.

Training Artificial Neural Networks to Learn a Nondeterministic Game

Thomas E. Portegys
DigiPen Institute of Technology
9931 Willows Rd. NE, Redmond, WA, 98052 USA
portegys@gmail.com

Abstract - *It is well known that artificial neural networks (ANNs) can learn deterministic automata. Learning nondeterministic automata is another matter. This is important because much of the world is nondeterministic, taking the form of unpredictable or probabilistic events that must be acted upon. If ANNs are to engage such phenomena, then they must be able to learn how to deal with nondeterminism. In this project the game of Pong poses a nondeterministic environment. The learner is given an incomplete view of the game state and underlying deterministic physics, resulting in a nondeterministic game. Three models were trained and tested on the game: Mona, Elman, and Numenta's NuPIC.*

Keywords: Mona artificial neural network, Elman artificial neural network, NuPIC hierarchical temporal memory, nondeterministic learning, game learning.

1 Introduction

Games, like many endeavors, are about reacting to and predicting events in the pursuit of goals. Games often also feature sequential actions as part of their play. Artificial neural networks (ANNs) have demonstrated considerable success in sequence prediction [1, 2]. For "conventional" multilayer perceptron (MLP) types of ANNs, sequences to be recognized are trained into the network beforehand as a set of static patterns. Because of this, reacting to untrained events is not a strength. This is an impediment to the use of MLPs in types of games that require this capability. Recent notable achievements playing arcade-style games [3] rely on the power of pattern classification rather than sequence recognition.

The aim of this project is to examine ANN architectures applied to learning a game that features both unpredictable events and sequential actions. These features are manifested in a nondeterministic finite automaton (NDA) [4]. Much of the world is nondeterministic, taking the form of unpredictable or probabilistic events that must be acted upon. If ANNs are to engage such phenomena, as biological networks do so readily, then they must be able to learn nondeterministic environments.

It is well known that recurrent MLPs, e.g. an Elman network, can learn deterministic finite automata [5, 6]. Learning the Reber Grammar is an example of this [7]. Learning nondeterministic finite automata is another matter. NDAs can produce event streams that are impossible to predict, making comprehensive ANN training infeasible. This is anathema for MLPs that rely on such training to be effective.

In this project the game of Pong provides a nondeterministic environment. While a deterministic game of Pong can readily be learned by an ANN given the ball position and velocity [8, 9], in this project the learner is given an incomplete view of the game state and underlying physics, resulting in a nondeterministic game.

Three ANN models were trained and tested on the game:

- Mona, a goal-seeking network [10, 11].
- Elman, a popular MLP recurrent network [12].
- Numenta's NuPIC, a model of hierarchical temporal memory (HTM), which is closely based on neurological structure and function [13, 14].

2 Description

2.1 Pong game environment

The computer game of Pong [15] involves striking a moving ball with a movable paddle in a two-dimension playing area on a computer screen. Two paddles, controlled by opposing players, are positioned at the left and right ends of the playing area where they can be moved up and down to meet the ball. The ball can bounce off of the sides of the area as well as the paddles. A player loses when the ball gets by the player's paddle without being struck successfully.

In this project there is only one player, the ANN learner, controlling a paddle that is located on the right side of the playing area. A loss occurs when the ball passes the paddle, and a win is signified by a successful paddle hit. From the player's point of view, the playing area is overlaid by a 5x5 grid. The grid does not affect ball movements. The learner possesses sensors and response capabilities that are only effective in its currently located grid cell.

2.1.1 Sensors

There are two sensors, one each for the ball and paddle states. Their values are supplied by the underlying game mechanics.

The ball sensor values:
BALL_ABSENT, BALL_PRESENT,
BALL_MOVING_LEFT, BALL_MOVING_RIGHT,
BALL_MOVING_UP, BALL_MOVING_DOWN
If the ball is moving up or down but also left or right, the ball sensor will report a vertical movement.

The paddle sensor values:
PADDLE_ABSENT, PADDLE_PRESENT

2.1.2 Responses

The learner can express these responses:

- CHECK_BALL: ask the physics for a ball sensor reading; this is only effective when the ball is in the current grid cell.
- TRACK_BALL_LEFT: move the learner's current grid cell left one cell, which is the correct response to the BALL_MOVING_LEFT sensor value.
- TRACK_BALL_RIGHT: move current grid one cell right.
- PAN_LEFT: move current grid cell left across the playing area until the ball or the left side is encountered.
- PAN_RIGHT: move current grid cell right until ball, paddle, or right side encountered.
- MOVE_PADDLE_UP: move paddle and current grid cell one cell up; this is only effective when the paddle is present.
- MOVE_PADDLE_DOWN: move paddle and current grid cell one cell down when the paddle is present.

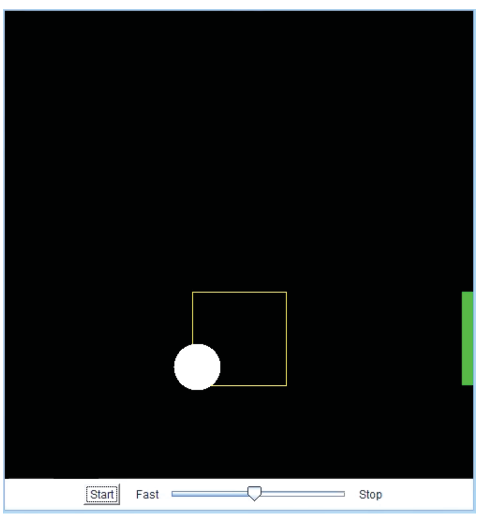


Figure 1 – Pong play. Box indicates sensory area.

Figure 1 is a snapshot of a game in progress. A video is also available on the web at http://youtu.be/Urdu9AJxoA0. Figure 2 shows the state space for winning games. The states are annotated with sensor states and the edges are annotated with responses. State transitions inputs are defined by implicit "step" signals which can have multiple target states, hence the state space embodies a nondeterministic automaton.

A game begins with the ball in the center of the playing area and the paddle in the center position. The ball is set to a random direction with a speed that is normalized so as not to outstrip the learner's ability to track it. Game play is turn-based, with the game mechanics changing the ball position and direction and the learner responding to sensor inputs.

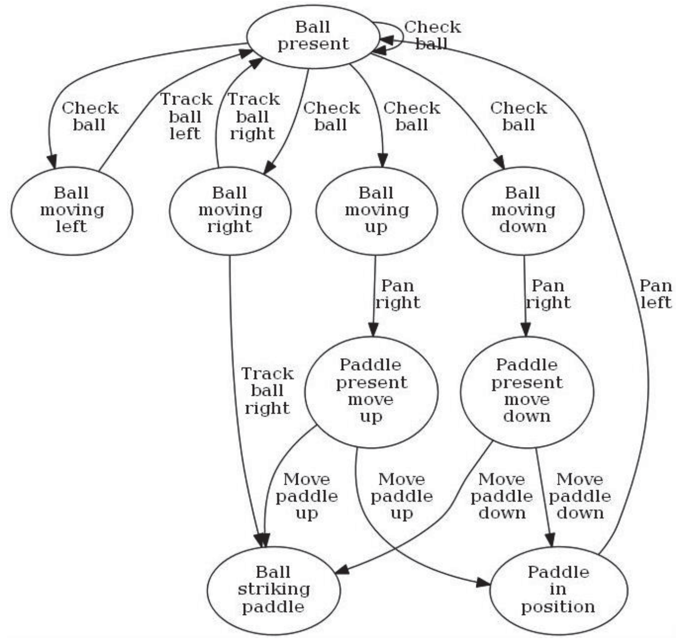


Figure 2 – Pong state space for winning games.

2.2 Training

The learner is trained to track with the ball as it moves left and right. When sensors indicate that the ball is moving up or down, the learner is trained to (1) pan right to the paddle, (2) move the paddle up or down to remain aligned with the ball, and (3) pan left to locate the ball. The paddle-movement sequence is particularly challenging to train for two reasons: (1) the learner must remember which way to move the paddle without sensing the ball, and (2) the learner must remember that after it has moved the paddle and while continuing to sense the paddle, it must pan left to the ball.

2.3 ANN models

The following ANN models were trained and tested on the Pong game task. It should be mentioned that a fourth model, BECCA (Brain Emulating Cognitive Control Architecture) [16], was considered for comparison but was not included due to time constraints. BECCA was exhibiting preliminary promising performance but was not optimally trained as of this writing.

2.3.1 Mona

Mona [10, 11] is a goal-seeking ANN that learns hierarchies of cause and effect contexts. These contexts allow Mona to predict and manipulate future events. The structure of the environment is modeled in long-term memory; the state of the environment is modeled in working memory. Mona uses environmental contexts to produce responses that navigate the environment toward goal events that satisfy internal needs. Because of its goal-seeking nature, Mona is also an example of reinforcement learning. Mona was selected for this task to illustrate the plasticity of a goal-seeking network in dealing with an NDA environment.

For the Pong task two sensors were configured, one for the ball and one for the paddle. A set of values between 0 and 1 were defined for the 6 ball sensor values and the 2 paddle sensor values. The response output ranged from 0 to 6 to encode the 7 possible response values.

2.3.2 Elman

An Elman network [12], also known as a Simple Recurrent Network, contains feedback units that allow it to retain temporal state information useful in classifying sequential input patterns. These feedback units reside in a context layer as shown in Figure 3. Each hidden layer unit has a connection to a corresponding context unit with a fixed weight of 1. An Elman network was selected for this task as a means of comparing non-MLP models with a popular MLP model.

For the Pong task the Elman network was created with Lens (Light efficient network simulator) [17]. The network was configured with 8 input units for the 6 ball sensor plus 2 paddle sensor values; 20 hidden and 20 context units; and 7 output units for the 7 possible response values. “Off”/“on” sensor values were 0/1. Outputs were similarly trained to values of 0 and 1. The learning rate was set to 0.2.

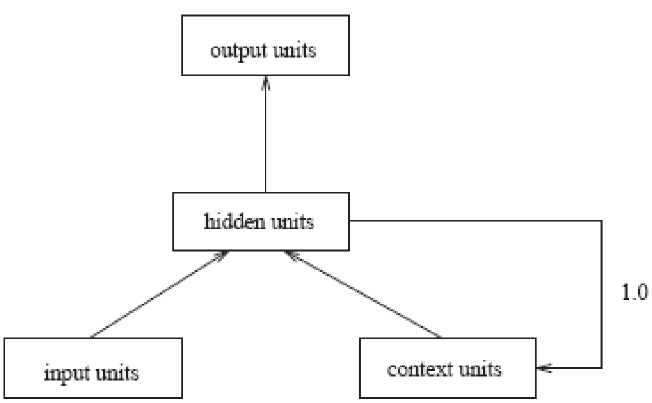


Figure 3 - Elman recurrent network.

2.3.3 NuPIC

NuPIC, the Numenta Platform for Intelligent Computing [13, 14], comprises a set of learning algorithms that attempts to faithfully capture how layers of neurons in the neocortex learn. NuPIC was selected for this task based on its successful performance in a number of sequential prediction tasks. At the heart of NuPIC is Hierarchal Temporal Memory, or HTM.

From an algorithmic point of view there are three principle properties:

- Sparse Distributed Representations (SDRs): a sensor encoding technique that permits both noise tolerance and efficient pattern comparisons.
- Temporal inference: prediction of upcoming patterns in a stream.
- On-line learning: learning and prediction are concurrent.

For the Pong task the inputs and output were configured as they were for Mona.

3 Results

3.1 Training

For training, fifty random games of Pong were generated. For Mona, the BALL_PRESENT sensor state was defined as the only goal, which motivates the network to produce responses to navigate to the ball. As Figure 2 demonstrates, returning to the BALL_PRESENT state will generate winning Pong actions.

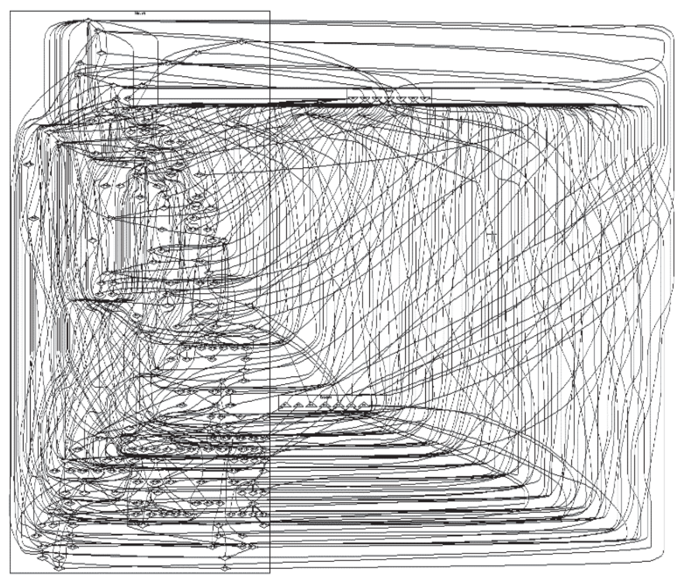


Figure 4 – Mona network after training.

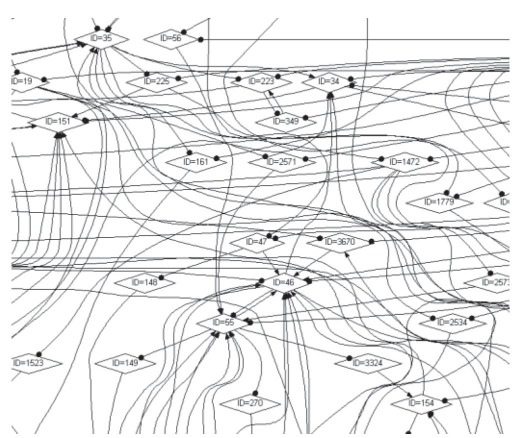


Figure 5 – Some mediator neurons in the trained Mona network.

A single pass of the training set was given with the correct responses enforced on the network, and working memory cleared before each game. The Mona network after training is shown in Figures 4 and 5.

For the Elman network, 5000 training epochs were performed. For NuPIC, a swarm optimization using the training set was performed to select the optimal internal parameter values.

3.2 Testing

For testing, a separate set of fifty random Pong games was generated. Each game was scored according to the percentage of initial consecutive correct responses toward winning the game. So if there were 10 responses to win a game and the learner output the first 8 correctly, the score for the game would be 80%. The rationale for this scoring scheme is that making any error will cast the learner off course from a winning response sequence.

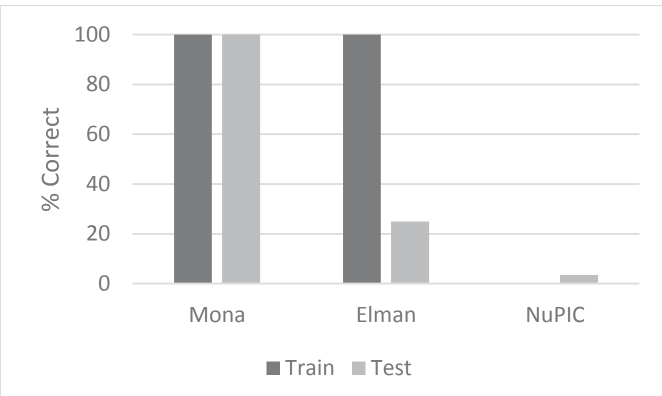


Figure 6 – Testing results.

Figure 6 shows results of testing with the test set as well as the training set for comparison (except for NuPIC). The Mona network performed perfectly for both sets. As might be expected, the Elman network performed perfectly for the set it was trained on, but much poorer for the test set, where it frequently encountered game play sequences that it was not trained to handle. This task obviously was not suitable for NuPIC, at least in its current form.

4 Conclusion

Even a simple nondeterministic game environment can pose significant problems for some ANN models, as the results show. For Mona, the goal-seeking component of its architecture is a major reason for its success on the task: it provides a mechanism for dynamically propagating motivation through a plethora of possible game sequences. For the Elman network, its success in predicting sequences that it has been trained on it notable. However, when sequences vary as they do in different untrained games, a marked decrease in performance ensues. For NuPIC, it seems clear that handling unpredictability is currently not a strong point. However, NuPIC remains under development as new neurological mechanisms are incorporated into it.

Modeling the brain has produced significant successes in the area of pattern classification for ANNs. However, the brain obviously has much more to teach in the domain of learning and executing behaviors that interact successfully with real-world environments. An aim of this project is to highlight the capabilities of models other than the prevalent multilayer perceptrons. These models can be complementary as well: for example, the pattern classification prowess of deep learning networks might be meshed with a behavioral oriented network such as Mona or a high fidelity neurological network model to form formidable hybrid architectures.

The open source code for Mona and the Pong project is available at http://mona.codeplex.com/ See the Readme in src/pong.

5 References

[1] Edwards, T., Tansley, D, S. W., Davey, N., Frank, R. J. (1997). Traffic Trends Analysis using Neural Networks. Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications 3. pp. 157-164.

[2] Bengio, S., Fessant F., Collobert (1995). D. A Connectionist System for Medium-Term Horizon Time Series Prediction. In Proc. Intl. Workshop Application Neural Networks to Telecoms pp. 308-315.

[3] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518, 529-533 doi:10.1038/nature14236

[4] Martin, J. (2010). Introduction to Languages and the Theory of Computation. McGraw Hill. p. 108. ISBN 978-0071289429

[5] Giles, C. L., Miller, C. B., Chen, D., Chen, H. H., Sun, G. Z., Lee, Y. C. (1992). Learning and Extracting Finite State Automata with Second-Order Recurrent Neural Networks. Neural Computation, Vol. 4, 393-405.

[6] Firoiu, L., Oates, T., Cohen, P. R. (1998). Learning a deterministic finite automaton with a recurrent neural network. Lecture Notes in Computer Science, Volume 1433, pp. 90-101.

[7] Cerňanský, M., Benuškov, L. (2001). Finite-State Reber Automaton and the Recurrent Neural Networks Trained in Supervised and Unsupervised Manner. Artificial Neural Networks — ICANN 2001. Lecture Notes in Computer Science, Volume 2130, 2001, pp. 737-742.

[8] Fyfe, C. (2005). Local vs Global Models in Pong, Artificial Neural Networks: Formal Models and Their Applications –

ICANN 2005, Lecture Notes in Computer Science, Volume 3697, 2005, pp 975-980.

[9] Pong video: https://www.youtube.com/watch?v=LD6OgKEj5JE

[10] Portegys, T. (2010). A Maze Learning Comparison of Elman, Long Short-Term Memory, and Mona Neural Networks. Neural Networks, 23(2):306-13.

[11] Mona white paper: http://tom.portegys.com/research/MonaWhitepaper.pdf

[12] Elman, J. L. (1990). Finding structure in time. Cognitive Sci. 14 179-211.

[13] Numenta website: http://numenta.org/ White paper: http://numenta.org/htm-white-paper.html

[14] Hawkins, J. (2004). On Intelligence (1 ed.). Times Books. p. 272. ISBN 0805074562.

[15] Pong: http://en.wikipedia.org/wiki/Pong

[16] Rohrer, B. (2012). BECCA: Reintegrating AI for natural world interaction, AAAI Spring Symposium on Designing Intelligent Robots: Reintegrating AI.

[17] Lens neural network simulator: http://web.stanford.edu/group/mbc/LENSManual/

SESSION

LATE BREAKING PAPERS: ARTIFICIAL INTELLIGENCE AND APPLICATIONS

Chair(s)

TBA

A model and appearence-based approach for gait recognition using SVM

Weder Cabral Mendes[1], **Gustavo Teodoro Laureano**[1]
Anderson Soares da Silva[1], **Clarimar José Coelho**[2] **and Caio Marcelo Nunes**[1]
[1]Instituto de Informática – INF, Universidade Federal de Goiás – UFG
[2]Departamento de Computação – CMP, Pontifícia Universidade Católica de Goiás – PUCGO
Goiânia, Goiás, Brazil
{wedermendes,gustavo,anderson}@inf.ufg.br,{clarimarc,kiomarcelo}@gmail.com

Abstract— *The human gait is gaining prominence as biometric recognition feature because it enables the identification of people from low-quality videos in a non-invasive and non-cooperative mode. However, the identification of the human gait with the presence of different clothes and accessories is a challenging task. The proposed method in this paper considers the use of silhouette width and the evolution of gait dynamics from the positions and angles of joints of a simplified model of human skeleton for recognizing the march using SVM for the classification. The tests are performed in CASSIA-B and the Cumulative Match Score database (CMS) is used to evaluate the performance of the classifier. The results using CMS Rank 10 show 99% accuracy for variations in bag and walking clothing.*

Keywords: gait recognition, SVM, feature fusion

1. Introduction

The human gait recognition by image has been recieving increasing attention by computer vision and biometric areas in order to the potential applications in intelligent video surveillance and human-computer interaction. Gait is a behavioral biometric feature, periodic and unique to each individual [1], [2]. It is a important method of identification because it does not require the subject's cooperation, it is more appropriate to be used in less controlled environments [3], [4], can be measured by different types of sensors [5], [6], it is characterized by the reduced quality of data [7] and can be captured from multiple views [8].

The main challenges of gait recognition by images are related to the internal and external factors that affects the gait. Internal factors include inherent conditions of the person, such as oldness, lesions and drunkenness. External factors are characterized by changes of camera view, dynamic environments, irregular floor, clothing and carrying accessories [9]. Among this factors, clothing and carrying accessories are common problems that have receieved intense investigation of researches in gait recognition.

The approaches for gait recognition are divided into two main categories: appearance-based and model-based[10], [2], [9]. The appearance-based approaches represents the motion features extracted from silhouette shape, such as width, texture, height and optical flow. Silhouette-based methods are sensitive to imagem quality, changes in appearance (clothes, shoes, obesity) and changes in viewing angle. However, the main advantage of this approach is the low computational cost.

The work of Han and Bhanu [11] considers a spatial-temporal representation of the gait cycle as the average normalized silhouettes, named Gait Energy Image (GEI). The gait cycle is represented by a single image that carries the statistical information of the measured gait movement sequence. Bhuyan Jagan [12] propose the use of two features named width of the binary vector silhouette and the MPEG-7 Angular Radial Transform (ART) as descriptors to represent the gait. Principal Component Analysis (PCA) is used to reduce the dimensionality and Hidden Markov Models (HMM) to the gait recognition.

The model-based approaches use geometric shapes to represent the body parts (head, arms, legs) and extract static and dynamic model characteristics. The model-based approaches generally are more robust to changes of appearance and viewing angle. However, they are sensitive to quality of image sequences and have high computational cost [13], [14].

[13] model the human body from static information of the distance between the members of the body and the silhouette of the time during the march. [15] use the distance and time that are covered during one gait cycle as biometrics features. [16] consider default measurements of anthropometric body to find the height of the members of the body and create a model of polygons to represent each member of the human body. The center of mass, orientation, and the area of each polygon compose the feature vector. In the work of [17], the march is represented by angles and angular velocities of five connected lines, which model the joints of the lower limbs of the human body.

This work presents a gait recognition method based on a lateral visualization of silhuete of a person. The gait is represented by a hybrid model that considers the positions of the joints, their angles and the width of silhueta as representative factors of the gait cycle dynamics. Support

Vector Machine (SVM) is used int the classification stage to evaluate the separability of proposed descriptors.

This work is organized in five sections. Section 2 presents the extraction method of the joints from the human body silhouette and the feature vector used for the recognition of march. The classification method used in this work is described in section 3. Section 4 presents the results of the classification and the conclusions in section 5.

2. Joint detection and feature extraction

In this paper, the human gait is represented by a sequence of images containing the silhouette of a person where each silhouette is a state of the march with its own characteristics. The silhouette is a binary image obtained by applying background removal algorithms such as Gaussian Mixture Models (GMM) [18], [19]. The width of the 2D position of the joints of the body image and the angles of the lines joining these joints are considered as characteristic of the silhouette.

Given the cyclical nature of human gait, in this work, only one cycle is considered. The beginning of the cycle is defined by the first crossing of the legs and the end of the cycle for the third crossing.

2.1 Silhouette width

Silhouette images are normalized to form same height silhouette, which are extracted 140 widths for each silhouette $w = [w_1,\ w_2, \ldots, w_{140}]$. The Figure 1 show an example of the 140 silhouette width.

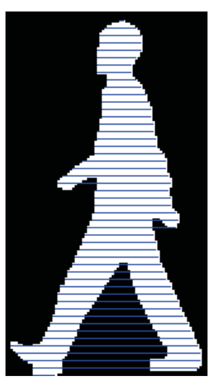


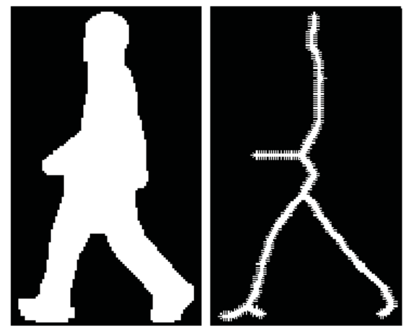
Fig. 1: Example of the 140 width vector used in this work.

2.2 Joint positions and angles

The human body is modeled by connected joints, similar to human anatomical skeleton. For simplification purposes, the model is formed by a reduced number of joint points: shoulder, elbow, waist and knee; and the end points: head, one hand and feet.

The position of each point is extracted from a skeletonized image after morphological thinning operation on the silhouette image [20]. Figure 2(b) shows the skeleton obtained from silhouette image of Figure 2(a).

Using the skeletonized image, the position of the head is given by the highest pixel which belongs to skeleton and the height of the skeleton H is determined as the distance from the head to the lower pixel belonging to the skeleton. The detection of other points is performed in regions of the skeleton estimated by height ratios for each joint, defined in the work of [21]. The estimated y position is given in the Figure 3.



(a) (b)

Fig. 2: (a) Silhouette and (b) skeleton image.

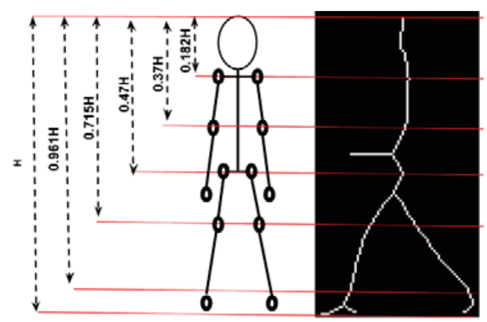


Fig. 3: Estimated y position of the joint points given by [21].

The final coordinates of the points are defined as following: the head, shoulder and waist positions are extracted by the intersection of the horizontal line at respective estimated y position with the skeleton; hand point is the most left pixel between elbow and waist line; left foot and left knee are the left most point at its respective y horizontal lines; right foot and right knee are the right most pixel at its respective y horizontal lines. The x and y coordinates of each joint is stored in the vector $p = [p_{1x}, p_{1y}, p_{2x}, p_{2y}, \ldots, p_{9x}, p_{9y}]$ for each frame. In adition, the angles between elbow and hand, waist and left knees (left and right) are stored in vector $a = [a_1, a_2, a_3]$.

3. Gait recogition

The gait representation of each silhouette frame is give by a vector $f = [w,\ p,\ a]$. The complete gait cycle is expressed by a matriz $F = \left[f^{(1)},\ f^{(2)},\ \ldots,\ f^{(n)}\right]$, where n is the nth frame in the image sequence. For classification is used SVM (Support Vector Machine) [22], [23].

The SVM is a classifier that creates border (hyperplane) of optimal decision between two classes. Where the hyperplane chosen to maximize margins between hyperplane and the two classes. For this, in training are only used the Support Vectors of each class, which is the data that are exactly on the border between two classes. The Figure 4 illustrates a data decision boundary in 2D.

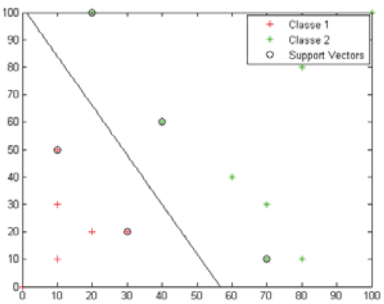


Fig. 4: Example of decision boundary of SVM.

As shown in Figure 4 test data that is left of the border are classified as Class 1 and if it is the right belongs to Class 2.

4. Results and discussion

The CASIA-B Gait Database are designed to evaluate the proposed method. This database has 124 subjects with 11 variations of view angle. Each subject consists in 10 sequences, out of which 6 sequences are walking with normal walking (NW), 2 sequences carrying bag (BW) and 2 sequences are walking wearing different clothes (CW).

In this paper, the lateral view (90º) of silhouette is used. Some subjects in this dataset do not have all sequences or do not have a complete gait cycle. Consequently, only 120 of all subjects are considered in these experiments.

The training set consisted of 3 sequences of normal walking named as (NW-1,NW-2,NW-3), 1 sequence with carrying bag named as (BW-1) and 1 sequence wearing different clothes named as (CW-1). The anothes sequences named as NW-4, NW-5, NW-6, BW-2 and CW-2 are used for testing.

The classification performance of the proposed methodology was evaluated using the Cumulative Match Score (CMS) [24]. The CMS presents the cumulative probability of the classifier accuracy from a class (person) and a test sample. The correct recognition rate (CRR) is measured as the ratio between the number of scores and the number of samples of each class. The highest correct recognition rate for a class given a sample is interpreted as rank 1. The recognition rate of incorrectly classified samples are accumulated to the second highest rate of correct recognition, defining the rank 2 and so on until Rank-N.

The Figure 5 shows the CMS to the set of test data. The curves express the performance of the classifier for each test sample. The more asymptotically increasing the curve is, the better the performance of the classifier in relation to the CRR. The CMS illustrates that from the rank-2 the proposed method overcomes 90 % accuracy for all samples considered.

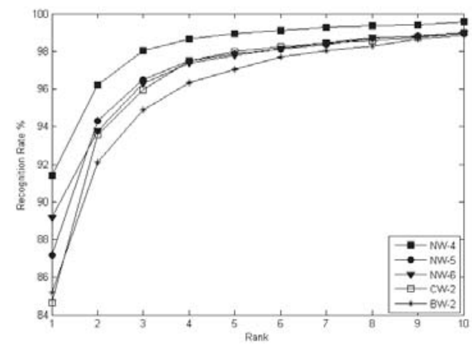


Fig. 5: Cumulative Match Score

The Table 1 lists the values accumulated the correct recognition rates of Ranks 1, 2, 3, 4, 5 and 10 compared with the results of[25]. The performance of the proposed method obtains better results for sequences of carraying bag and wearing clothes at the same time that the mean CRR is 87.512 and it is close of the CRR of the compared work that is 87.662 for Rank 1.

5. Conclusion

In this paper, we propose a computer vision method for recognizing persons, a hybrid approach was used to represent dynamic and static features of human gait. The evolution of positions and angles of joints in a simplified model of the human skeleton are used as features. To evaluate the feature extraction method, we used the SVM classifier. The results demonstrate that the proposed combination of features shows similar results to the methods of the state-of-art, with improved accuracy rate in experiments with clothing and bags changes.

References

[1] Y. Chai, J. Ren, W. Han, and H. Li, “Human gait recognition: Approaches, datasets and challenges,” 2011. [Online]. Available: http://digital-library.theiet.org/content/conferences/10.1049/ic.2011.0100

[2] A. K. Jain, P. J. Flynn, and A. A. Ross, *Handbook of biometrics*. New York, N.Y.: Springer, 2008. [Online]. Available: http://public.eblib.com/choice/publicfullrecord.aspx?p=371949

[3] J. Kovač and P. Peer, “Human skeleton model based dynamic features for walking speed invariant gait recognition,” *Mathematical Problems in Engineering*, vol. 2014, 2014.

[4] K. Ricanek, M. Savvides, D. Woodard, and G. Dozier, “Unconstrained biometric identification: Emerging technologies,” *Computer*, vol. 43, no. 2, pp. 56–62, Feb 2010.

	Metodologia Proposta						[25]					
	Taxa de Reconhecimento/ Rank (%)						Taxa de Reconhecimento/ Rank (%)					
Testes	1	2	3	4	5	10	1	2	3	4	5	10
NW-4	91.39	96.19	98	98.64	98.94	99.56	**93.33**	**97.5**	**100**	**100**	**100**	**100**
NW-5	87.16	94.31	96.5	97.48	97.85	98.98	**96.66**	**100**	**100**	**100**	**100**	**100**
NW-6	89.18	93.75	96.31	97.37	97.77	98.94	**92.5**	**95.83**	**100**	**100**	**100**	**100**
BW-2	**85.18**	**92.12**	**94.90**	**96.32**	**97**	**98.87**	76.66	85.83	89.16	90.83	91.66	96.66
CW-2	**84.65**	**95.53**	**95.95**	**97.47**	**97.99**	**99**	79.16	86.66	90	91.66	94.16	97.5
Média	87.51	**94.48**	**96,33**	**97.45**	**97.91**	**99.07**	**87.66**	93.16	95.83	96.49	97.16	98.83

Table 1: Results obtained with the proposed method and the results of [25].

[5] E. Gianaria, N. Balossino, M. Grangetto, and M. Lucenteforte, "Gait characterization using dynamic skeleton acquisition," in *Multimedia Signal Processing (MMSP), 2013 IEEE 15th International Workshop on*. IEEE, 2013, pp. 440–445. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6659329

[6] T. Whytock, A. Belyaev, and N. M. Robertson, "Dynamic distance-based shape features for gait recognition," *Journal of Mathematical Imaging and Vision*, vol. 50, no. 3, pp. 314–326, Nov. 2014. [Online]. Available: http://link.springer.com/10.1007/s10851-014-0501-8

[7] C. Padole and H. Proença, "An aperiodic feature representation for gait recognition in cross-view scenarios for unconstrained biometrics," *Pattern Analysis and Applications*, pp. 1–14, 2015. [Online]. Available: http://dx.doi.org/10.1007/s10044-015-0468-0

[8] C. Chen, J. Liang, and X. Zhu, "Gait recognition based on improved dynamic bayesian networks," *Pattern Recognition*, vol. 44, no. 4, pp. 988–995, Apr. 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320310005054

[9] J. Wang, M. She, S. Nahavandi, and A. Kouzani, "A Review of Vision-Based Gait Recognition Methods for Human Identification." IEEE, Dec. 2010, pp. 320–327. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5692583

[10] D. Gafurov, "A survey of biometric gait recognition: Approaches, security and challenges," in *Annual Norwegian Computer Science Conference*. Citeseer, 2007, pp. 19–21. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.324.975&rep=rep1&type=pdf

[11] J. Han and B. Bhanu, "Individual recognition using gait energy image," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 2, pp. 316–322, 2006. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1561189

[12] M. K. Bhuyan and A. Jagan, "Person identification using gait by combined features of width and shape of the binary silhouette," *World Academy of Science, Engineering and Technology, WASET*, vol. 56, pp. 1059–1066, 2011. [Online]. Available: http://www.waset.org/publications/494

[13] A. F. Bobick and A. Y. Johnson, "Gait recognition using static, activity-specific parameters," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–423. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=990506

[14] D.-U. Jung, W. G. Oh, and J.-S. Choi, "Model-based gait tracking method: A review of recent development gesture interaction," in *Frontiers of Computer Vision,(FCV), 2013 19th Korea-Japan Joint Workshop on*. IEEE, 2013, pp. 250–253. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6485498

[15] C. BenAbdelkader, R. Cutler, and L. Davis, "Stride and cadence as a biometric in automatic person identification and verification," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*. IEEE, 2002, pp. 372–377. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1004182

[16] N. V. Boulgouris and X. Huang, "Gait recognition using hmms and dual discriminative observations for sub-dynamics analysis," *Image Processing, IEEE Transactions on*, vol. 22, no. 9, pp. 3636–3647, 2013.

[17] W. Zeng, C. Wang, and Y. Li, "Model-Based Human Gait Recognition Via Deterministic Learning," *Cognitive Computation*, vol. 6, no. 2, pp. 218–229, June 2014. [Online]. Available: http://link.springer.com/10.1007/s12559-013-9221-4

[18] P. KaewTraKulPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," in *Video-based surveillance systems*. Springer, 2002, pp. 135–144.

[19] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 2. IEEE, 2004, pp. 28–31. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1333992

[20] R. C. Gonzalez, *Digital Image Processing*. Pearson, 2004, ch. Some Basic Morphological Algorithms.

[21] W. T. Dempster and G. R. Gaughran, "Properties of body segments based on size and weight," *American Journal of Anatomy*, vol. 120, no. 1, pp. 33–54, 1967. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/aja.1001200104/abstract

[22] S. a. K. K. Theodoridis, *Pattern Recognition*. Elsevier, 2009.

[23] R. O. Duda, P. E. Har, and D. G. Stork, *Pattern Classification*, 2008.

[24] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The feret evaluation methodology for face-recognition algorithms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 10, pp. 1090–1104, 2000.

[25] H. Kumar and H. Nagendraswamy, "Gait recognition: An approach based on interval valued features," in *Computer Communication and Informatics (ICCCI), 2013 International Conference on*. IEEE, 2013, pp. 1–6.

User Interfaces for Representing Knowledge Stemming from Debates: Evaluating the Impact of Threading Models (Reviews) on Online Products

Abdulrahman. Alqahtani [1,2], and Marius. Silaghi[2]
[1]Department of Computer Science, Najran University, Najran, Saudi Arabia
[2]Department of Computer Science, Florida Institute of Technology, Melbourne, FL, USA

***Abstract*—Most consumers trust online reviews, such as personal recommendations. The main goal of the research described in this paper is to study the impact of reviews of online products for the classification of threading models of electronic debates. We present DirectDemocracyP2P Android application as a user interface for decentralized debates. Based on the results of our online survey of 1027 participants, we found that participants preferred reviews (comments/threads) for online products to be a structured platform, used to help extract a conclusion of arguments around the product. In this paper, we evaluate and analyze the reviews (threads) that were collected from online surveys to improve our DirectDemocracyP2P Android application. Moreover, We discuss the factors that attract users who read or tape reviews (comments/threads) for any given online product.**

Keywords: The techniques to evaluate knowledge representation, User Interfaces for Representing Knowledge, DDP2P, Evaluation of the Impact of Threading Models on Online Products, and Results

I. Introduction

Many online products made by companies asking customers to leave a review/comment on their site exist. Most of the previous studies focus on the impact of online reviews on product sales. They tend to ignore the structure of user interfaces, and they assume the impact of reviews is the same across different types of online products [1]. Online networks, including Facebook and MySpace, provide an easy way to create circles of friends. These applications enable the people to connect with a worldwide community. However, there are some limitations with respect to privacy issues. PeerSoN Social Networks can be used to solve this privacy issue problem [2], [3]. PeerSoN provides Privacy-Preserving P2P Social Networks to integrate peer-to-peer social networking with ubiquitous computing and delay-tolerant networks [4]. DebateDecide (one of the DDP2P platforms) is a web-based platform for debate threading. It also has a version for client-based viewers called DirectDemocracyP2P. DirectDemocracyP2P (DDP2P) system is an engine with multiple graphical interfaces [5]. In this paper, we present DDP2P Android application for debate threading. This study investigates the impact of reviews/threads activity on electronic debates by assessing consumer comments with use of ratings information as a means of evaluating the threading models on online products. However, there are many risks consumers face online today, such as the lack of information between consumers and online sellers, system security issues, poorly designed user interfaces, and fraud [6].

A. Background

Here we present some techniques that can be used to evaluate knowledge representation:

- Surveys: Surveys can be used for knowledge representation evaluation. In 2009, McDonald et al evaluated three formats of privacy policies for six companies. There were 749 internet users for this study and each filled out the survey to evaluate a company's privacy with one format [7].
  In 2005, Janez Brank presented a survey of the state of the art in ontology evaluation [8]. He grouped the ontology evaluation approaches depending on level of evaluation, as follows:
  - Lexical, vocabulary, or data layer:
    This level focused on concepts, instances, facts, etc, which are included in the ontology. Also it considers the vocabulary that is used to represent these concepts. Evaluation in this level involves comparison with different sources of data.
  - Hierarchy or taxonomy: It studies the relation between concepts
  - Other semantic relations:
    This level includes other relations between concepts besides is-a. Evaluation of this level contains measures such as precision and recall.
  - Context or application level:
    The Context level occurs when an ontology becomes part of a huge collection of ontologies, which is mentioned by separate definitions in those

other ontologies. Evaluation at this level may help to take the context into account while evaluating different definitions. application is another form of context where the ontology is to be used.
  - Syntactic level:
    The Syntactic level matches the syntactic requirements of the particular formal languages that describe the ontology.
  - Structure, architecture, design:
    This is primarily manually constructed ontologies that evaluate certain pre-defined design principles or criteria for further development.
- Case Studies:
  A few case studies exist that have been to improve the knowledge representation evaluation. In 2003, Kim and Chan proposed a way to evaluate reordered Google search results by associating each item with a score provided by clusters of human users and aggregating the total score based on the position of the valuable items [9].
- Panels:
  The panel (group of people) can be used to compare different forms of knowledge representation. Bobrow, in his research (a panel on knowledge representation), mentions that he asked panel members to briefly answer three questions in order to compare different forms of knowledge representation [10]. The three panel questions were: "What are the most important premises underlying your approach to knowledge representation, the critical ideas, and major mechanisms used in your system [10]".

In our study we use online surveys to collect the data from online users for evaluation of online product debates to improve the interface of the DDP2P application.

II. User Interfaces for Representing Knowledge

In a DDP2P application, all debates and arguments start with a motion which is relevant to a given organization. Users can vote on it with justifications. Here we introduce an Android application for DDP2P according to motions and justifications that are in the exit organization.

a) Motions:: A motion is a proposal related to a statute, constitutional amendment, or discussion issue that is raised for the vote of a committee or constituency. In some organizations (e.g., U.S. towns), the motions can be submitted to the town council[1] only by a certain number of members of the council. In other organizations (e.g., Swiss towns) a motion can be submitted by any group of citizens who gather signatures [5]. Motions are called to the citizen initiatives available in some organizations (e.g, Florida, California, Switzerland). Proposals for laws that are

[1]The council will move on the motion.

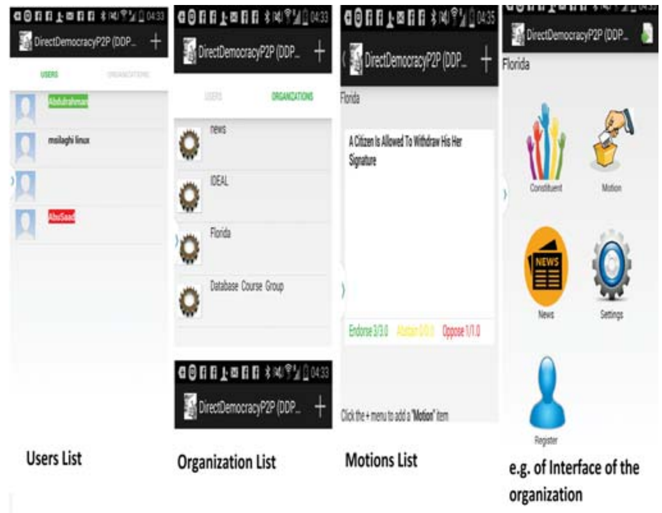


Figure 1: User Interfaces for DirectDemocracyP2P Android application1

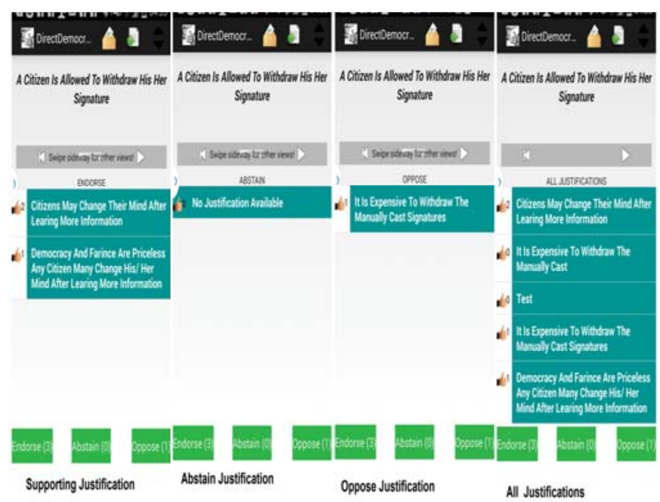


Figure 2: User Interfaces for DirectDemocracyP2P Android application2

based on a certain number of signatures gathered can be placed on a referendum ballot. DirectDemocracyP2P allows for a motion to be disseminated among the members of a group of peers (the citizens/members of an organization) for collecting signatures in view of a potential submission for a vote.

Prior to moving on a motion, in general, one only gathers support signatures for it. That does not allow for the debates and collaborative filtering that is needed when a large number of motions may be vehiculated. The traditional ways to filter the motions are:

- Only enable the members of a small committee to submit motions (e.g., town councils, parliaments)

- Request a large number of signatures prior to moving on the motion, the cost of which filters out poorly financed or very unpopular proposals (e.g., citizen initiatives and statutory petitions)

The first type of filter (reduction to a small committee) reduces the quality of the democracy and the trend is to abandon it for the second type [5]. The second approach also reduces the democracy when it reduces itself to barriers based on high financing requirements. When signatures can be gathered using social media, the financing differences are leveled and the motions compete solely based on their popularity/support. However, the number of motions raised to the attention of eligible supporters is now expected to increase, and new filtering mechanisms are required to increase the visibility of meritorious ones.

Various social media (e.g., news forums, Amazon) are already facing this problem of abundance of inputs (proposals, comments), and mechanisms of collaborative filtering are popular under the form of voting on entries. However, most of these social media are easily attacked by *response rate manipulation*, when interested peers can spam the channel with large numbers of submissions.

In the case of motions, similar mechanisms based on enabling negative voting (besides the positive support, which is the final purpose), can yield an appropriate filtering. Moreover, if each constituent is restricted to one single current input (which is appropriate with votes or signatures on one motion), then the response rate manipulation can be reduced even more.

The mechanism of disseminating motions can be used to help the community converge towards enhanced versions. Discovery of better versions of a motion can be boosted by an appropriate threading mechanism, where each new motion can refer to the previous motions on which it claims to improve. These references create a thread that can be traversed by a user, or can be used by automatic reasoning tools that can help users in locating promising motions.

b) Justifications:: One of the benefits of gathering votes for a motion is that constituents can get an understanding concerning the position of their peers, and therefore better grasp the implications of a given motion on their organization and on peer members. Namely, if they see that many peer members disagree with a motion that they consider to be obviously good, they may think twice and potentially discover problems with that motion, problems which can make the constituent withdraw his/her support. Withdrawing support for an unpopular motion will save the time of the other constituents who will be less tempted to spend time reading it and will save the organization the money needed to move on it and organize an official ballot.

Understanding the opinion of one's peers can be further enhanced by enabling the submitters of votes to associate a *justification* of their support or opposition to the motion. Since each constituent can submit a single current vote for the motion, it can become acceptable for them to only have one current justification (measure that can help to tame the *response rate manipulation* attacks). Since the existence of a large number of constituents in some organizations can lead to large numbers of justifications, it is important to also enable additional collaborative filtering mechanisms for justifications. In DirectDemocracyP2P, users can support somebody else's justification, as an alternative to providing his/her own justification. Justifications with a large base of support can be favored by viewers, as they may better represent the opinion of the group.

A further mechanism to help users locate relevant justifications is based on threading (just as with the enhanced motions). Namely new justifications can point to old justifications that they claim to refute or enhance. Thereby people visualizing old justifications are notified of the presence of the refutation and enhancement claims.

Common (and DirectDemocracyP2P default) alternatives when voting on a motion are *Support, Oppose*, and *Abstain*. However each submitted motion can be customized to allow for any set of possible reactions as appreciated by its designer. Poor choices are supposed to be correctable by enhancements.

c) Problem: The problem that emerges is to scientifically decide whether the collaborative filtering mechanisms proposed in debate systems truly offer the enhancements for which they were proposed.

III. Methodology

A survey is the best tool that can be used to collect data for program improvement. A survey process has five main steps of designing and implementing a survey [11]:

- Design Survey Process
- Develop Questions
- Test/Train
- Collect Data
- Analyze Data

Any type of survey for a specific objective is the result of making decisions based on the following techniques:

- Contacting information (Email, telephone, mail, etc.)
- Submitting the questions (writing, typing, interview)
- Recording answers (paper, voice, video)

The researcher can use one of these three techniques according to his/her research goals and timeline. Also, the choice of technique may affect the quality, cost, and timeliness of results. In our study we use the second technique (submitting the questions) in an online survey to collect the data from online users.

A. Evaluation of the Impact of Threading Models on Online Products

We collected the information through an online survey from May to June 2015 in which we presented a survey to participants and asked them to answer its questions. We used Survey Monkey for designing the questions. All questions of the survey were multiple choice. Reading questions was a requirement for completing an online survey. We designed one question as a test for validation of our survey. We designed and distributed questionnaires in our survey in a couple of languages (English and Arabic) to track in which language a participant took the survey, since most of our media network members knew one or both of these two languages. This element allows you to track in which language a participant takes the survey.

B. Study Questions

Study questions contained three groups:

- Participation Agreement
  The first question in our survey is a participation agreement. Participation is voluntary. Refusal to participate will involve no penalty or loss of benefits to which you are otherwise entitled. You may discontinue participation at any time without penalty or loss of benefits to which you are otherwise entitled.
- General Information: We collected general information like gender, age range, secondary language, and level of education to evaluate our survey population. Also, this general information will help us to build the user interfaces of the DDP2P Android application according to the level of people who deal with reviews (threads).
- Understanding Questions: Participants answered a chain of multiple choice questions to determine the factors that attract the users while reading or taping reviews (comments/threads) for any online product. These factors will help us to improve our DirectDemocracyP2P Android application.

C. Goals of Our Survey

Each question of our survey was used to measure two things:

- How to evaluate the debate of reviews for online products.
- How to improve user interfaces for debate application which is a DD2PP Android application in our case.

IV. Results and Discussion

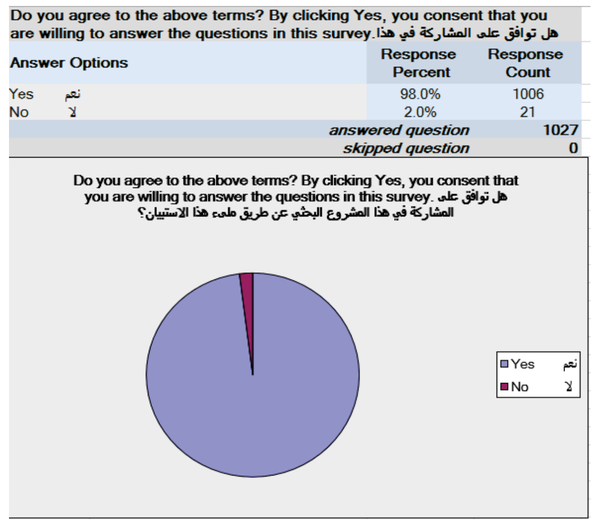
A. Participation Agreement (Institutional Review Board (IRB))

The first question in our survey asked the user to accept a participation agreement.

We asked:
Do you agree to the above terms? By clicking Yes, you consent that you are willing to answer the questions in this survey.

Key Finding:
98 percent accepted to answer our survey.
2 percent rejected to participate in our survey.

Analysis:
Most of the participants accepted to answer the questions in our survey (1006 of 1027) as shown in Figure 3. This was an easy question because this question confirmed the participation in our survey.



Do you agree to the above terms? By clicking Yes, you consent that you are willing to answer the questions in this survey.هل توافق على المشاركة في هذا

Answer Options	Response Percent	Response Count
Yes نعم	98.0%	1006
No لا	2.0%	21
answered question		1027
skipped question		0

Figure 3: Institutional Review Board (IRB)

B. General Information:

There were four questions about general information. Every question had to fulfill the purpose of the study.

1) Gender: This question aimed to know who was more active in the debate, men or women. This question will help to structure debate user interfaces (DDP2P applications) according to the interaction of humans.

We asked: What is your gender?

Key Finding:
78.5 percent of participants were male.
21.5 percent of participants were female.

Analysis:
The majority of participants were male from both languages of the survey (725 of 924) as shown in Figure 4. This meant males were more willing to debate than females, according to this survey. This question will lead to focusing on men more, by using ads, news, topics, etc, in a debate user interface (DDP2P applications), in order to attract them into successful debates.

2) Age Range: the age range question targeted the age range of participants who are willing to debate.

We asked: What is your age range?

Key Finding:
4.3 percent of participants were less than 20.
30.3 percent of participants were between 20 and 30.
50.3 percent of participants were over than 30.

Analysis:
The greatest age range of participants, who were willing to

What is your gender? الجنس		
Answer Options	Response Percent	Response Count
Male ذكر	78.5%	725
Female أنثى	21.5%	199
	answered question	924
	skipped question	103

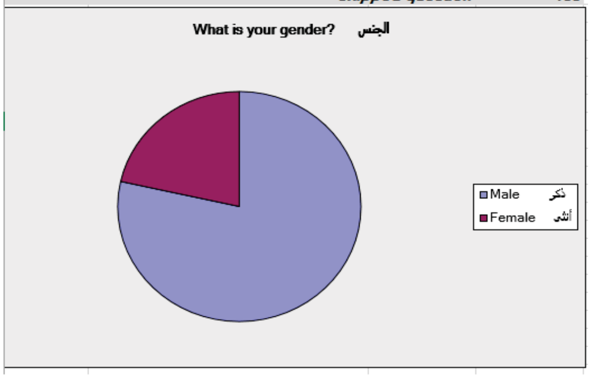


Figure 4: Gender percent of participants in our survey

debate in this study, was older than 30 (465 of 924), then between 20 and 30 (419 of 924) as shown in Figure 5. This question gave us the age range of participants whom we should focus on when we improve the user interface of DDP2P applications.

What is your age range? كم عمرك؟		
Answer Options	Response Percent	Response Count
Less than 20 أقل من ٢٠	4.3%	40
Between 20 and 30 بين ٢٠ و٣٠	45.3%	419
Over than 30 أكثر من ٣٠	50.3%	465
	answered question	924
	skipped question	103

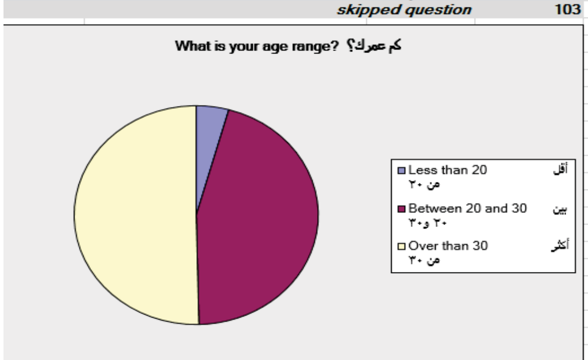


Figure 5: Age Range of participants in our survey

3) Secondary Language: The secondary language question aimed to discover which languages are the most popular in the debate.
We asked: What is your secondary language, if any?
Key Finding:
78.5 percent of participants whose second language was English.
0.2 percent of participants whose second language was Chinese.
0.8 percent of participants whose second language was French.
1.5 percent of participants whose second language was Spanish.
11.1 percent of participants whose second language was Other.
Analysis:
We found English was the most popular language in our study (798 of 924) as shown in Figure 6. From this question, in DDP2P applications, we will suggest using English as a formal language to communicate between users. Also, we will put English as the default user interface for DDP2P applications.

What is your secondary language, if any? ماهي لغتك الثانوية		
Answer Options	Response Percent	Response Count
English الأنجليزية	86.4%	798
Chinese الصينية	0.2%	2
French الفرنسية	0.8%	7
Spanish الأسبانية	1.5%	14
Other (please specify) أخرى (الرجاء التحديد)	11.1%	103
	answered question	924
	skipped question	103

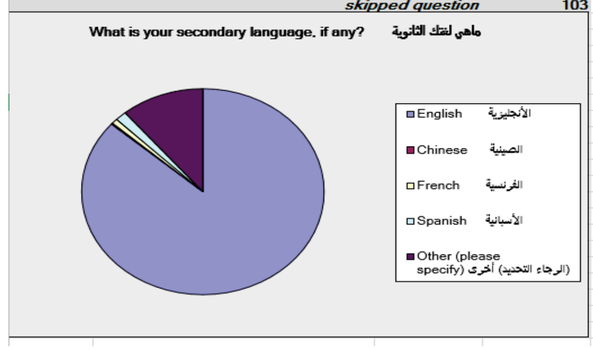


Figure 6: Secondary Language of participants in our survey

4) Level of the Education: The level of education question referred to the impact of level of education on the debate.
We asked: What is your education level?
Key Finding:
15 percent of participants have a High School degree
51.1 percent of participants have a Bachelor's degree
27.1 percent of participants have a Master's degree
6.8 percent of participants have a Ph.D degree
Analysis:
Most participants have a Bachelor's degree (472 of 924) as shown in Figure 7. This question showed us that most participants could be familiar with any updates or developments for improving the user interface of DDP2P applications because the majority of participants had a Bachelor's degree.

C. Survey Validity

We had a question to test the validity of our online survey. The validity question depended on asking questions which measured what we were supposed to be measuring.

Answer Options		Response Percent	Response Count
High School	الثانوية العامة	15.0%	139
Bachelor	البكالريوس	51.1%	472
Master	ماجستير	27.1%	250
Ph.D.	دكتوراه	6.8%	63
		answered question	924
		skipped question	103

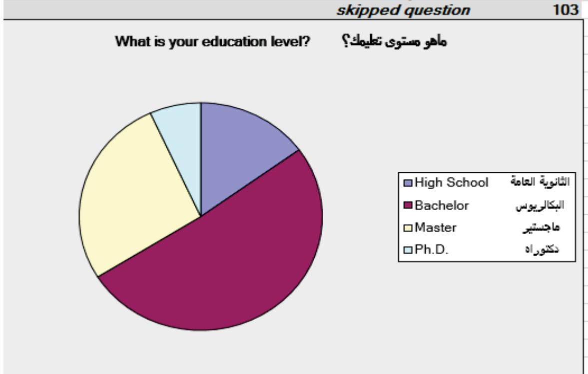


Figure 7: Level of Education of participants in our survey

How likely would you be to read a product review (comments/threads) before making a decision to purchase it? ماهو إحتمال قرأتك لتقييم منتج معين			
Answer Options		Response Percent	Response Count
Usually- Frequently	دائما	43.9%	406
Sometimes	بعض الأحيان	46.2%	427
Never	أبدا	9.8%	91
		answered question	924
		skipped question	103

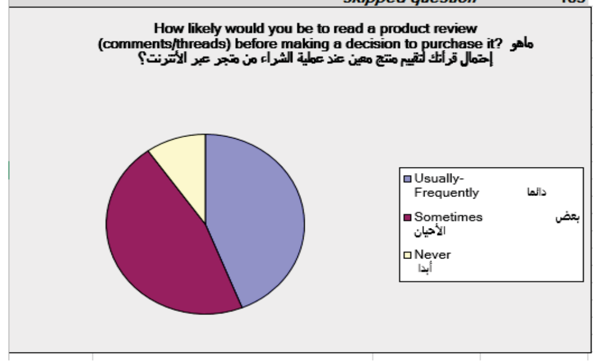


Figure 8: The Validity of our Survey

We asked: How likely would you be to read a product review (comments/threads) before making a decision to purchase it?
Key Finding:
43.9 percent of participants usually read a product review (comments/threads) before making a decision to purchase it
46.2 percent of participants sometimes read a product review (comments/threads) before making a decision to purchase it
9.8 percent of participants never read a product review (comments/threads) before making a decision to purchase it.
Analysis:
The result of this question referred to whether the majority of participants would read a product review "Usually" (409 of 924) or "Sometimes" (427 of 927) as shown in Figure 8. Whoever answered "Never" for this question could not continue to the next series of questions because the remaining questions focused on actual readers of product reviews.

D. Threads Questions:

We have several questions which focused on reviews of online products. Our samples were the participants who read reviews. They were supposed to answer a chain of multiple choice questions to determine the factors that attract users while reading or taping reviews (comments/threads) for any online products. The results of those questions will help us to improve our DirectDemocracyP2P applications.

1) Trusting the Justifications: Fifty-three percent of participants were trusted to read a brief review (388 of 720) as shown in Figure 9.
We asked: What types of reviews (comments/threads) do you trust the most?
Key Finding:
53.9 percent of participants were trusted to read a brief review.
37.1 percent of participants were trusted to read a long review.
9.0 percent of participants were not likely to trust any online review
Analysis:
Designing the brief review by Limiting the length of the motion will help to attract users to debate according to the results of this question. Limitation of the length of the debate arguments will directly affect users' acquisition and, in turn, trusting the justifications about any given motion in the DDP2P applications.

What types of reviews (comments/threads) do you trust the most? ماهو نوع التقييم الذي تثق فيه أكثر؟			
Answer Options		Response Percent	Response Count
Brief review	التقييم القصير	53.9%	388
Long reviews	التقييم المطول	37.1%	267
None	لاشيء	9.0%	65
		answered question	720
		skipped question	307

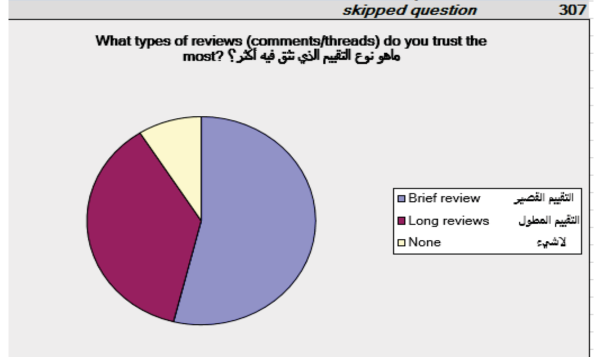


Figure 9: Trusting the Justifications

2) Sorting the Important Justification : Most users would read up to 10 reviews according to the results of this question

(558 of 720) as shown in Figure 10.
We asked: How many reviews (comments/threads) do you normally read before buying a specific product from an online store?
Key Finding:
37.4 percent of participants read 5 or less reviews before buying a specific product from an online store
36.4 percent of participants read 10 or less reviews before buying a specific product from an online store
20.8 percent of participants read more than 10 reviews before buying a specific product from an online store
5.4 percent of participants do not read reviews before buying a specific product from an online store.
Analysis:
In DDP2P applications, sorting the important justifications among the top ten justifications (around a given motion) will give the user opportunity to read them.

Answer Options		Response Percent	Response Count
5 or less	أقل من خمس	37.4%	269
10 or less	أقل من عشر	36.4%	262
More than 10	أكثر من عشر	20.8%	150
I don't read reviews.	لا أقرأ التقييمات	5.4%	39
		answered question	720
		skipped question	307

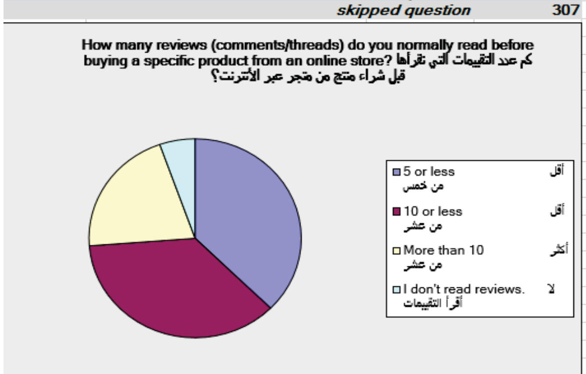


Figure 10: Sorting the Important Justification

3) Separating the Justification: Seventy-one percent of participants were likely to read both sides of the arguments (Positive or Negative Reviews)for any debate (517 of 720) as shown in Figure 11.
We asked: When you read reviews (comments/threads) for a product, do you focus on positive reviews or negative reviews?
Key Finding:
11.8 percent of participants were likely to read positive sides of arguments for any debate.
16.4 percent of participants were likely to read negative sides of arguments for any debate.
71.8 percent of participants were likely to read both sides of arguments (Positive or Negative Reviews) for any debate.
Analysis:
In DDP2P applications, we have already separated the justification on a motion whether *Support, Oppose*, or *Abstain*.

When you read reviews (comments/threads) for a product, do you focus on positive reviews or negative reviews? * عندما تقرأ تقييم لمنتج ما هل تركز			
Answer Options		Response Percent	Response Count
Positive	الجانب الإجابي	11.8%	85
Negative	الجانب السلبي	16.4%	118
Both	كلاهما	71.8%	517
		answered question	720
		skipped question	307

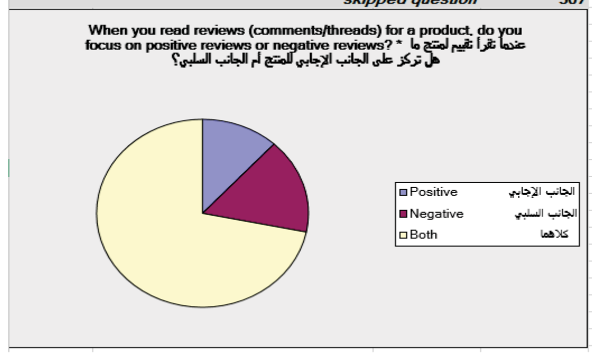


Figure 11: Separating the Justification

4) Showing the number of Justifications and Witnesses: Fifty-five percent of participants agreed with the statement, "Would a number of positive reviews (comments/threads), the number of stars, or other rating criteria, be enough for you to buy a specific product from an online store?" (517 of 720) as shown in Figure 12.
We asked: Are a number of positive reviews (comments/threads), the number of stars, or other rating criteria, enough for you to buy a specific product from an online store?
Key Finding:
55.4 percent of participants answered (Yes).
44.6 percent of participants answered (No).
Analysis:
Showing a number of positive reviews (comments/threads), the number of stars, or other rating criteria will attract users to read and write reviews and make good arguments. In DDP2P, a number of justifications, the number of witnesses, or other rating criteria should be shown in the first page of the user interface for the motion.

5) Form for Attention-Grabbing-Words: Sixty-one percent of participants would be attracted by any type of online product reviews (Positive or Negative Words)(447 of 720) as shown in Figure 13.
We asked: What types of words attract you the most while reading reviews (comments/threads) for any online product?
Key Finding:
22.1 percent of participants were attracted by positive words of arguments.
15.8 percent of participants were attracted by negative words of arguments.
62.1 percent of participants were attracted by both sides of

Are a number of positive reviews (comments/threads) and the number of stars, or other rating criteria, enough for you to buy a specific product from		
Answer Options	**Response Percent**	**Response Count**
Yes نعم	55.4%	399
No لا	44.6%	321
	answered question	720
	skipped question	307

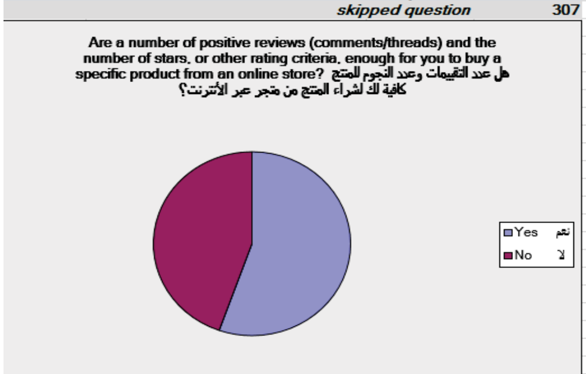


Figure 12: Showing a number of positive reviews (comments/threads) and the number of stars, or other rating criteria

arguments (Positive or Negative words).

Analysis:

In DDP2P applications, we could design a form for attention-grabbing-words which would attract users to become more involved in the debate.

What types of words attract you the most while reading reviews (comments/threads) for any online product? مانوع الكلمات التي تجذب أنتباهك		
Answer Options	**Response Percent**	**Response Count**
Positive words الكلمات الإجابية	22.1%	159
Negative words الكلمات السلبية	15.8%	114
Both كلاهما	62.1%	447
	answered question	720
	skipped question	307

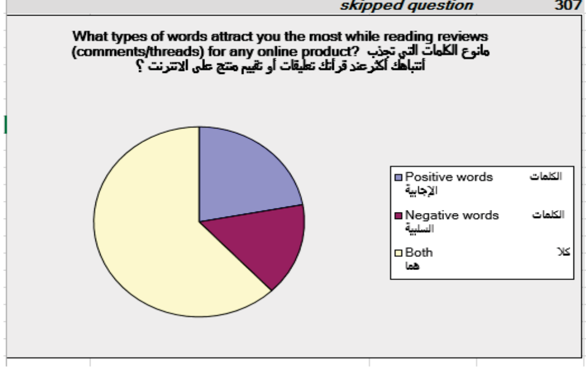


Figure 13: Types of words that attract users

6) Form for emphasizing words: Forty-seven percent of participants would expand words for emphasis while the same percentage would never do that as shown in Figure 14. **We asked:** When you type a comment in reviews (comments/threads) for any online product, do you expand some words for emphasis? For example verrrrrrrrrrry

Key Finding:

5.6 percent of participants were always likely to expand some words for emphasis when they typed a comment in online reviews.

47.2 percent of participants were sometimes likely to expand some words for emphasis when they typed a comment in online reviews.

47.2 percent of participants were never likely to expand some words for emphasis when they typed a comment in online reviews.

Analysis:

In DDP2P applications, we can design a form for emphasizing words which will attract users to become more involved in the debate.

When you type a comment in reviews (comments/threads) for any online product, do you expand some words for emphasis? For example		
Answer Options	**Response Percent**	**Response Count**
All the time دائما	5.6%	40
Sometimes بعض الأحيان	47.2%	340
Never أبدا	47.2%	340
	answered question	720
	skipped question	307

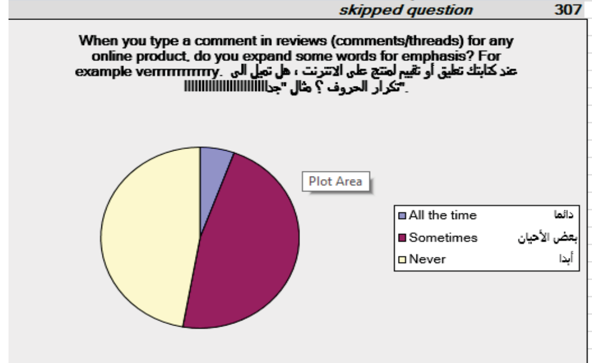


Figure 14: Expanding some words for emphasis

7) Form for Translating Words of the User Region: Fifty-three percent of participants would use argot language from their region (385 of 720) as shown in Figure 15.

We asked: When you type a comment in a review (comments/threads) for any online product, do you use argot language from your region?

Key Finding:

41.2 percent of participants were always likely to use argot language from their region.

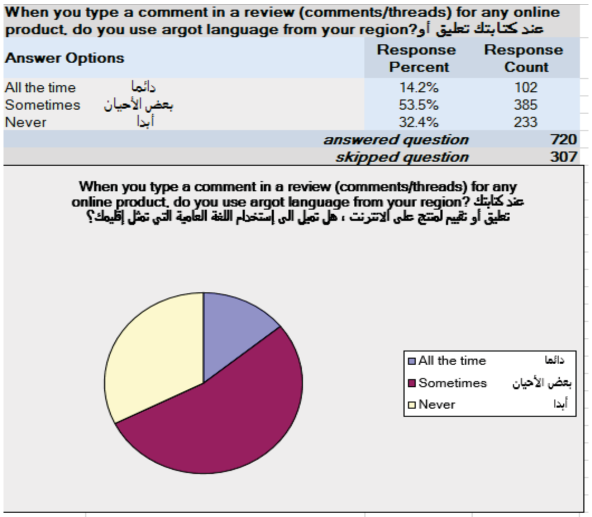
53.5 percent of participants were sometimes likely to use argot language from their region.

32.4 percent of participants were never likely to use argot language from their region.

Analysis:

In DDP2P applications, we should design a form for translating words of the user's region to English, and give some space to clarify these words (enhancement).

8) Form for Supporting Translation: Sixty-eight percent of participants were never likely to use words from other languages (385 of 720) as shown in Figure 16. **We asked:** When you type a comment in a review (comments/threads)



When you type a comment in a review (comments/threads) for any online product, do you use argot language from your region?عند كتابتك تعليق أو		
Answer Options	Response Percent	Response Count
All the time دائما	14.2%	102
Sometimes بعض الأحيان	53.5%	385
Never أبدا	32.4%	233
answered question		720
skipped question		307

Figure 15: Using argot language from user region

for any online product, do you use some words from other languages?

Key Finding:

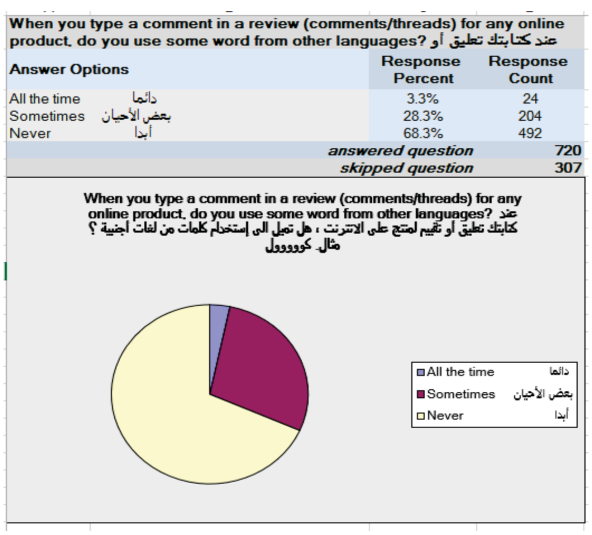
3.3 percent of participants were always likely to use words from other languages

28.3 percent of participants were sometimes likely to use words from other languages

68.3 percent of participants were never likely to use words from other languages

Analysis:

In DDP2P applications, we should design a form for supporting translation of words of the users' languages, and give the users space to explain these words (explanation).



When you type a comment in a review (comments/threads) for any online product, do you use some word from other languages? عند كتابتك تعليق أو		
Answer Options	Response Percent	Response Count
All the time دائما	3.3%	24
Sometimes بعض الأحيان	28.3%	204
Never أبدا	68.3%	492
answered question		720
skipped question		307

Figure 16: Using any word from other languages

9) Benefit of Study Threads: Forty-percent of participants said that reviews about any online products are positive reviews while thirty-six percent described online reviews as serious reviews. A few participants considered online reviews as negative reviews as shown in Figure 17.

We asked: From your perspective, how would you generally describe reviews (comments/threads) about any online product?

Key Finding:

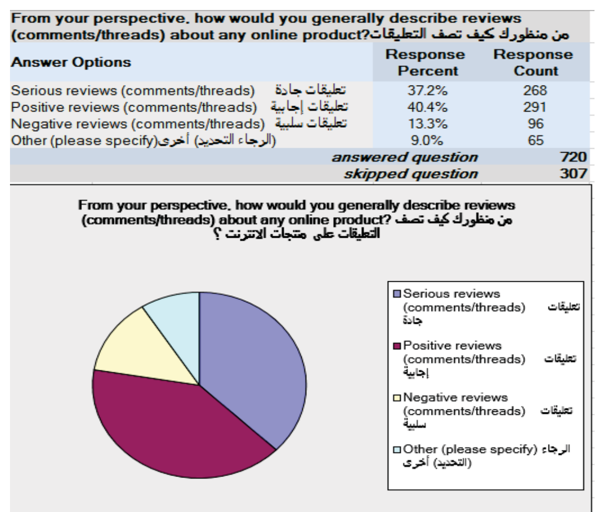
37.2 percent of participants described online reviews as serious reviews

40.4 percent of participants described online reviews as positive reviews

9.0 percent of participants described online reviews as negative reviews

Analysis:

The results of this question gave us the benefit of studying online reviews (comments/threads). There are a lot of users who trust online reviews, especially if they are serious and positive reviews.



From your perspective, how would you generally describe reviews (comments/threads) about any online product?من منظورك كيف تصف التعليقات		
Answer Options	Response Percent	Response Count
Serious reviews (comments/threads) تعليقات جادة	37.2%	268
Positive reviews (comments/threads) تعليقات إجابية	40.4%	291
Negative reviews (comments/threads) تعليقات سلبية	13.3%	96
Other (please specify)أخرى (الرجاء التحديد)	9.0%	65
answered question		720
skipped question		307

Figure 17: Benefit of Study Reviews/Threads

10) Structured/Unstructured Platform for Threads: Fifty-six percent of participants were likely to prefer reviews (comments/threads) for online products to be structured platforms, which could be a specific question that the user should answer or comment on. Structured platforms helped extract a conclusion of arguments around the product as shown in Figure 18.

We asked: How do you prefer reviews (comments/threads) for online products to be structured platforms or unstructured platforms?

Key Finding:

56.9 percent of participants preferred reviews (comments/threads) for online products to be structured platforms

43.1 percent of participants preferred reviews (comments/threads) for online products to be unstructured platforms

Analysis:

Forty-one percent of participants were likely to prefer reviews (comments/threads) for online products to be unstructured platforms. In DDP2P applications, we should have those two types of platforms. Unstructured platforms could be used for peers to join or create any organizations/motions, and structured platforms could be used for voting to post only one justification for any given motion, and whether they support it or are against it.

How do you prefer reviews (comments/threads) for online product to be كيف تفضل ان تكون التعليقات على منتجات الانترنت ؟?		
Answer Options	**Response Percent**	**Response Count**
Structured Platform (it could be a specific question	56.9%	410
Unstructured Platform (users can tape whatever they	43.1%	310
	answered question	**720**
	skipped question	**307**

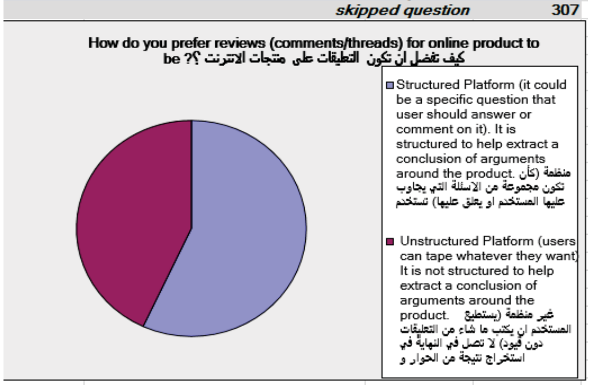


Figure 18: Structured/Unstructured Platform for Reviews/Threads

V. Acknowledgment

We are sincerely grateful to Dr. Muzaffar Shaikh, Dr. John Lavelle, Dr. Khalid Abuhasel for their support and sharing their truthful and illuminating view on a number of issues related to the survey.

VI. Conclusion

Here we have analyzed the impact of reviews (threads) for online products based on an online survey. While online platforms utilize comments to support or negate the author's message and/or other comments in the thread, none of them have been analyze for threading models. We have presented DDP2P Android applications as a user interface for decentralized debates. In our survey we provided some measure that may be used to evaluate the debate system. Also, we have mentioned the following suggestions and techniques that may be used to improve DDP2P applications:

- Limitation of the Words of Debate Arguments.
- Sorting the Important Justifications among the Top Ten Justifications
- Separating the Justification
- Showing the Number of Justifications and Witnesses
- Form for Attention-Grabbing-Words
- Form for emphasizing words
- Form for Translating Words of the User Region
- Form for Supporting Translation
- Benefit of Study Threads
- Structured/Unstructured Platform for Threads

References

[1] A. J. Flanagin, M. J. Metzger, R. Pure, and A. Markov, "User-generated ratings and the evaluation of credibility and product quality in ecommerce transactions," in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*. IEEE, 2011, pp. 1–10.

[2] G. Kreitz, O. Bodriagov, B. Greschbach, G. Rodríguez-Cano, and S. Buchegger, "Passwords in peer-to-peer," in *Peer-to-Peer Computing (P2P), 2012 IEEE 12th International Conference on*. IEEE, 2012, pp. 167–178.

[3] B. Greschbach, G. Kreitz, and S. Buchegger, "The devil is in the metadataâĂŤnew privacy challenges in decentralised online social networks," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*. IEEE, 2012, pp. 333–339.

[4] S. Buchegger, D. Schiöberg, L.-H. Vu, and A. Datta, "Peerson: P2p social networking: early experiences and insights," in *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*. ACM, 2009, pp. 46–52.

[5] M. C. Silaghi, K. Alhamed, O. Dhannoon, S. Qin, R. Vishen, R. Knowles, I. Hussien, Y. Yang, T. Matsui, M. Yokoo *et al.*, "Directdemocracyp2pâĂŤdecentralized deliberative petition drivesâĂŤ," in *Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on*. IEEE, 2013, pp. 1–2.

[6] Z. Lee, I. Im, and S. J. Lee, "The effect of negative buyer feedback on prices in internet auction markets," in *Proceedings of the twenty first international conference on Information systems*. Association for Information Systems, 2000, pp. 286–287.

[7] A. M. McDonald, R. W. Reeder, P. G. Kelley, and L. F. Cranor, "A comparative study of online privacy policies and formats," in *Privacy enhancing technologies*. Springer, 2009, pp. 37–55.

[8] J. Brank, M. Grobelnik, and D. Mladenić, "A survey of ontology evaluation techniques," 2005.

[9] H. R. Kim and P. K. Chan, "Learning implicit user interest hierarchy for context in personalization," in *Proceedings of the 8th international conference on Intelligent user interfaces*. ACM, 2003, pp. 101–108.

[10] D. G. Bobrow, G. G. Hendrix, W. A. Martin, J. McCarthy, A. Newell, R. Schank, B. C. Smith, and N. Sridharan, "A panel on knowledge representation," in *Proceedings of the 5th international joint conference on Artificial intelligence-Volume 2*. Morgan Kaufmann Publishers Inc., 1977, pp. 983–992.

[11] N. Thayer-Hart, J. Dykema, K. Elver, N. Schaeffer, and J. Stevenson, "Survey fundamentals: A guide to designing and implementing surveys," *Office of Quality Improvement*, 2010.

Intelligent Highway Vehicle Traffic Flow Monitoring and Control System

Theko Emmanuel Marie and * Barnabas Ndlovu Gatsheni (Prof)

Vaal University of Technology, Private Bag X021, Vanderbijlpark 1900, Gauteng, South Africa

*University of Johannesburg, Auckland Park 2006, Gauteng, South Africa

Abstract - ***In this paper multilayer perceptron (MLP) is used to help in monitoring and controlling of the vehicle traffic on the highway. The MLP achieve this by using the last 5 minutes data to predict the next 5 minutes traffic condition. Thereafter, the predicted traffic status is displayed on the highway for the motorists to read. The proposed model shows 100% accuracy. Implementation of this work will definitely decrease traffic congestion across main arteries of Johannesburg. Pollution normally experienced when cars idle for a long time during congestion will be reduced by free highway traffic flow. Frequent servicing of motor vehicles will no longer be required by the motorists. Furthermore the economy of Gauteng and South Africa as a whole will benefit due to increase in production. Consumers will also benefit in obtaining competitive prices from organizations that depends on haulage services.***

I. INTRODUCTION

South Africa is currently experiencing traffic jams of unprecedented levels especially in the big cities of Johannesburg and Pretoria and also along Ben Schoeman highway which connects these cities. The development of alternative routes to link these urban metropolitan areas is hindered by the already established building infrastructure. Consequently, it takes one twice the free flow time to reach the intended destination.

With 3.8 million registered vehicles in Gauteng [1] there is continual annual traffic increase rate of 7%. The increase in traffic in-turn increases the average travel time to work with 50 minutes in Johannesburg. Most industries consequently experience economic uncertainty due to high traffic congestion in the main arteries of Johannesburg.

Currently the multi-billion rand rapid train link called Gautrain has been introduced. This is a stretch of over-ground and under-ground railway spanning 80 km that links the cities of Johannesburg and Pretoria and also links these cities with OR Tambo International Airport. Even though not enough, there is approximately 30% reduction of traffic congestion along Ben Schoeman highway (link between Johannesburg and Pretoria) as the result of the Gautrain project [2]. Vehicles can park at a cost of R12.00 per day and vehicle owners can make use a train to travel to intended destination [2].

Furthermore, South African National Roads Agency Limited (SANRAL) has installed about 36 electronic tollgates along freeways (N1, N3, N4, N12) in Gauteng province, whereby all motorists will be required to pay a fee at these tollgates [3]. The anticipation is that due the high cost at these tollgates might force some motorists to use public transport thus resulting in traffic decongestion [3].

Related work on short term vehicle traffic flow prediction models in Bejing China Xu Ting, SUN Xiaoduan, HE Yulong and XIE Changrong [4] propose the use of radial basis function (RBF) neural network and wavelet analysis in speed forecasting of Beijing urban freeway. The gray model (GM) optimized using a sine function relation to make gray action into a dynamic time variable is proposed [5] while Wang Yan, Wang Hua and Xia Limin uses back-propagation neural network optimised using genetic algorithm [6] to predict the changing trend of the future. The proposed models successfully predicted dynamic traffic flow with less accuracy than the proposed model in the current study.

Complementing current efforts by SANRAL and Gautrain by building an intelligent system that can predict traffic congestion along the highway will be beneficial to all road users, the economy of Gauteng and the economy of South Africa than any other strategy that has been employed before.

II. LITERATURE REVIEW

Real time traffic information collected over 24 hours per day using inductive loop detectors is used by Xu *et al.* [4]. A wavelet technology is applied to a signal derived from the loop data in order to remove noise from data. RBF neural network is used to predict three traffic statuses which include free flow, transition status and congestion according to occupancy. RBF prediction models seem to confuse free flow and transition traffic statuses by predicting the same traffic pattern between 70 to 80 km/h for both.

Due to high uncertainty of traffic flow Mao *et al.* [5] use a gray model theory to forecast short term traffic flow for the system containing incomplete information and uncertainty factors. Mao *et al.* use original traffic data to find the changing laws of the system to generate a strong regularity of the data series, thus predict the changing trend of the future. The superiority of the used gray model is its ability to do prediction with some of the information known and partial information not known. However Mao *et al* forecast future trend for the next one hour, using a time series of one hour contributes a little towards traffic decongestion since traffic dynamics changes frequently within a period of one hour.

The past seven days traffic is used to predict the following 5 days by Wan *et al* [6]. The model is formed by training the artificial neural network with back-propagation algorithm until a minimum training error is obtained and immediately upon reaching minimum training error, genetic algorithm is used to search the weight and thresholds of the neural network thus forming an initial population and offspring population. The model shows success rate of 91%. However, upon prediction of congestion on the freeway for the following 5 days Wan *et al* introduce a human element by appointing policeman to regulate traffic on the highway under inspection. Introduction of human element can greatly re-introduce congestion.

III. EXPERIMENTS AND RESULTS

The vehicle traffic flow along the Ben Schoeman freeway was observed. Then based on this observation, the vehicle traffic flow data was randomly generated using the Matlab randperm function. Manual clustering of these random traffic data was done before training the MLP model to order vehicles speeds in such a way as to portray the vehicle traffic flow as was observed on the Ben Schoeman freeway. During clustering each instance represent an average speed value arising from multiple speeds in the last 5 minutes. In order to ensure that all the classes to be predicted (congested, into congested, out congested and normal) are properly represented during training, the dataset of 326 instances (66% plus 15 % of 402 random generated data) was used to train and validate the MLP model. MySQL database was used to store these 326 instances.

Experiment 1: Predicting traffic status using multilayer perceptron

The first 5 minutes data was first used to train the MLP model, thereafter every 5 minutes data was used to predict the next 5 minutes traffic condition. A complete artificial neural network model, the MLP is shown in Figure 1, where t_1 - t_5 is time in minutes from the 1st minute to the 5th minute and where $P_{t_1-t_5}$ is vehicle input pattern feed to the MLP model from the 1st minute to the 5th minute. The 7 inputs feed to the model during the first 5 minutes includes discreticised vehicle speeds from each of the four classes (Vbelow_hresh t_1-t_5 ,Vabove_thresh t_1-t_5 ,Vvery_ slow t_1-t_5 and Vright t_1-t_5), was it raining in the last 5 minutes? i.e. weather information (rainy t_1-t_5), any accidents in the last 5 minutes? i.e. accident information (accidents t_1-t_5) and presence of people who do road maintenance on the highway in the last 5 minutes (roadworkers t_1-t_5).

The aim of this MLP model in Figure 1 is to predict either “normal”, or “into congestion”, or “out congestion” or “congestion” as a traffic status condition likely to happen in the next 5 minutes ($P_{t_{10}}$) as shown in Figure 1. In Figure 1 if the predicted traffic condition is A (Congestions) it means

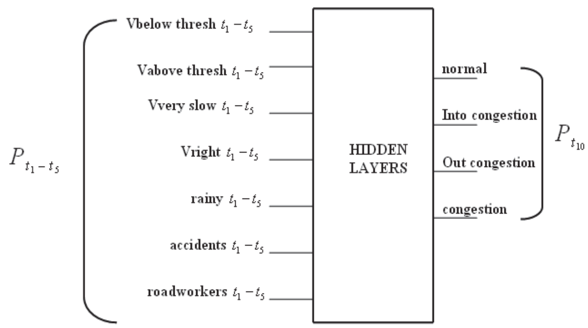


Figure 1: MLP prediction model used for 5 minutes ahead prediction

that vehicle speed on the highway will be between 0 to 45.4km/h. If the predicted traffic condition is B (Into congestion) it means that vehicle speed on the highway will be between 45.5 to 60.5km/h. If the predicted traffic condition is C (Out of Congestion) it means that vehicle speed on the highway will be between 60.5 to 80.4km/h. If the predicted traffic condition is D (normal highway traffic) it means that vehicle speed on the highway will be between 80.5 to 120.4km/h.

The experiments were carried out using different architectures as shown in Table 1 through Table 4 before the final winning model was chosen. The remaining 19 percent of vehicle traffic data (separate data saved on an excel spreadsheet) was used to test each of the architectures. Data shown in Figure 2 was used to train and validate the MLP model. This vehicle traffic data was pushed as it is shown in the Figure 2.

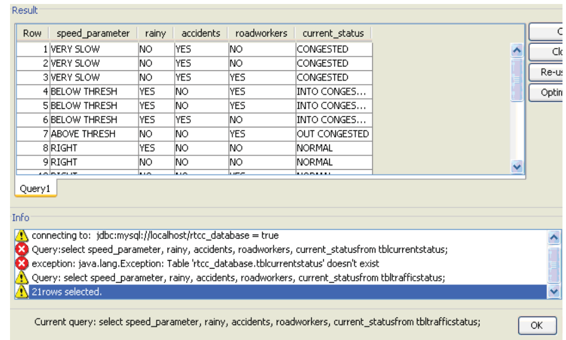


Figure 2: Shows loaded data in WEKA from MySQL database

At the beginning, experiments were carried out by varying the number of inputs as shown in Table 2, Table 3 and Table 4. Furthermore, the numbers of hidden units were also varied. When varying the number of inputs not all the inputs could be varied since from intuition it is known that the vehicle speed pattern forms the integral part of the prediction model. When varying the number of inputs, *speed_parameter = below thresh*, *very slow*, *above thresh* and *right* were shown in Figure 2 were not touched. *Roadworkers*, *rainy* and *accidents* are the inputs which were randomly varied in order to see how the model fares when any of these inputs are omitted. The experiments started by using all 7 inputs while varying the number of hidden units as shown in Table 1.

Table 1: The results of varying the number of hidden units, with the number of inputs and outputs remaining constant

MLP architecture	*Training performance*			*Testing Accuracy in %*
	Training accuracy in %	*RMSE*	*Time taken to train the model in seconds*	
7:1:4	86	0.23	0.4	48
7:2:4	99	0.02	0.3	84
7:3:4	90	0.07	0.6	80
7:4:4	95	0.09	0.7	83
7:5:4	100	0.01	1.0	100
7:6:4	89	0.27	0.8	79
7:7:4	87	0.26	0.8	83
7:14:4	79	0.31	1.02	68

The criterion used for selecting the best model was a root mean square error (RMSE) value and the accuracy shown by the MLP model during training. If a model performed poorly (having a large RMSE), the adjustment of the number of hidden units was done. At the end of the experiments in Table 1, the architecture 7:5:4 showed better performance with 100% accuracy during both training and testing while maintaining a low RMSE value of 0.01. Thus this indicates that using 5 hidden units the model provides a better performance.

Once the architecture 7:5:4 was identified, in the next experiments shown in Table 2 the inputs: *roadworkers*, *accidents* and *rainy* were randomly varied in order to see if the same performance can be achieved when any of these inputs were omitted by the model. In the 1st experiment in Table 2, *roadworkers* is the feature that was left out. In the 2nd experiment, *roadworkers* and *rainy* are the inputs which were left out. In the 3rd experiment, *roadworkers*, *rainy* and *accidents* were all left out.

Table 2: The results of using 5 hidden units while varying the number of inputs with outputs remaining constant

MLP architecture	*Training performance*			*Testing Accuracy in %*
	Training accuracy in %	*RMSE*	*Time taken to train the model in seconds*	
6:5:4	91	0.23	0.9	81
5:5:4	83	0.29	0.7	76
4:5:4	76	0.37	0.5	68

Varying the number of inputs did not improve the model results as shown in Table 2. Decreasing the number of inputs resulted in the increase of the RMSE value thus affecting the accuracy of the model during training and testing. Seen that the results for the architectures 7:2:4 (99%) and 7:4:4 (95%) during training were not far away from 7:5:4 (100%) as shown in Table 1, in Table 3 and Table 4 the number of inputs were also varied for these 2 architectures. The same sequence as in Table 2 was followed when varying the number of inputs (In the 1st experiment, *roadworkers* is the feature that was left out. In the 2nd experiment, *roadworkers* and *rainy* are the inputs which were

and *accidents* were all left out). The results are shown in Table 3 and Table 4.

Table 3: The results of using 2 hidden units while varying the number of inputs with outputs remaining constant

MLP architecture	*Training performance*			*Testing Accuracy in %*
	Training accuracy in %	*RMSE*	*Time taken to train the model in seconds*	
6:2:4	86	0.37	0.5	67
5:2:4	80	0.30	0.4	67
4:2:4	80	0.29	0.2	65

Varying the number of inputs for the two competing models as shown in Table 3 and Table 4 did not improve these models accuracy, as a result 7:5:4 architecture was chosen as the best model to be used to forecast the next 5 minutes vehicle traffic status.

Table 4: The results of using 4 hidden units while varying the number of inputs with outputs remaining constant

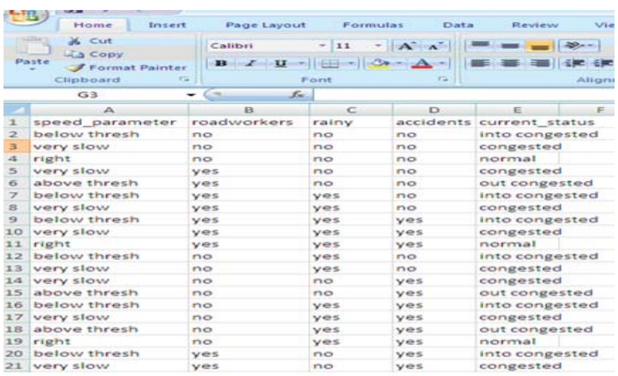
MLP architecture	*Training performance*			*Testing Accuracy in %*
	Training accuracy in %	*RMSE*	*Time taken to train the model in seconds*	
6:4:4	91	0.38	0.8	76
5:4:4	90	0.38	0.5	71
4:4:4	86	0.37	0.4	67

The chosen MLP model in Table 1 was made up of 7 inputs, 5 hidden nodes and 4 outputs (7:3:4 model). The MLP model was constructed using 66% training data (more data during learning) with 15% of data used as a validation set to fine tune the parameters of the classifier. Thereafter 19% of the remaining data (76 instances not used during training) was used to test the model. The training of the MLP using the back-propagation algorithm was used and 326 instances loaded in Figure 2 were used to train the network. The parameters chosen were a momentum of 0.2, a learning rate of 0.3, a validation set of 15 percent, validation threshold of 20 and 500 epochs. The proposed model used the early stopping approach to prevent noise which can cause over-fitting. The network was configured to automatically reset when a lower learning rate was obtained.

Cross validations using 10 folds were used in order to mitigate any bias which might be caused by a particular sample. Using 10 folds means repeating training and testing 10 times using 10 different random samples. In each iteration 66% of the data was randomly selected for training. Each of the 10 models was tested and the performance metrics of all the 10 tests were averaged to give the final model.

Using the trained MLP model to predict novel traffic data

An excel spreadsheet shown in Figure 3 was used to provide new vehicle traffic data not used during training (76 instances). To classify these novel instances whose targets are known, the objective was to find out how many of these instances the model was going to correctly classify.



	speed_parameter	roadworkers	rainy	accidents	current_status
2	below thresh	no	no	no	into congested
3	very slow	no	no	no	congested
4	right	no	no	no	normal
5	very slow	yes	no	no	congested
6	above thresh	yes	no	no	out congested
7	below thresh	yes	yes	no	into congested
8	very slow	yes	yes	no	congested
9	below thresh	yes	yes	yes	into congested
10	very slow	yes	yes	yes	congested
11	right	yes	yes	yes	normal
12	below thresh	no	yes	no	into congested
13	very slow	no	yes	no	congested
14	very slow	no	no	yes	congested
15	above thresh	no	no	yes	out congested
16	below thresh	no	yes	yes	into congested
17	very slow	no	yes	yes	congested
18	above thresh	no	yes	yes	out congested
19	right	no	yes	yes	normal
20	below thresh	yes	no	yes	into congested
21	very slow	yes	no	yes	congested

Figure 3: New vehicle traffic data in excel spreadsheet

IV. DISCUSSION OF RESULTS

Detailed results are shown in Figure 4.

During testing the trained MLP model was used to predict the likelihood of the occurrence of x in equation (1) for each instance of sample P when y has already occurred. Equation (1) reads as follows: what is the probability of observing x given that y has already occurred.

$$P(x|y)\,, \qquad (1)$$

where y is actual data and x is predicted data.

There are four parts shown on the testing results output in Figure 4 namely "prediction on test split", "evaluation on test set summary", "detailed accuracy by class" and the "confusion matrix".

The "confusion matrix" is used for post-processing the results. "Evaluation on test set summary" shows the summary of results achieved during testing by the trained MLP model. Prediction of each novel instance is shown under "predictions on test split". Towards the end of the results output in Figure 4 "detailed accuracy by class" section outlines the performance metric of the model.

Each of the 76 instances loaded from excel spreadsheet were predicted. The overall results summary is shown in Table 5. The first two lines in Table 5 show the accuracy and error rate of the model. All 76 instances were classified correctly.

A Kappa statistic value is 0 for the lack of any relation and 1 for very strong statistical relation. Kappa statistics of 1 shown in Table 5 show that the model has very strong statistical relations between the class label and the conjunction of constraints of attributes (instances). The mean absolute error, the root mean square error (RMSE), the relative absolute error and the root relative square error are most useful during numeric prediction. These errors depict the detailed error rate encountered by the MLP model during testing. The mean absolute error is the sum of errors from all incorrectly classified instances, i.e. is the sum of the difference between the predicted and the actual value for each incorrectly predicted instance.

The RMSE is the square of mean absolute errors. The mean absolute error and the RMSE are used to determine the learning rate of the model. The error rate close to zero in Table 5 (0.0131 and 0.0176 for mean absolute and root mean square respectively) shows that the knowledge a model obtained during training is good. The closer the error rate is to zero the more accurate is the model during prediction.

Relative absolute error and root relative square error give an idea of the scale of the error compared to how varied the actual values are i.e. the more varied the values, the harder the task of prediction. With 4 percent for both relative absolute error and root relative square error in Table 5 it is shown that the prediction task was not difficult due to strong statistical relations between the predicted classes and the actual instances.

```
Time taken to build model: 0.3 seconds

=== Predictions on test split ===

inst#,    actual, predicted, error, probability distribution
     1 1:into con 1:into con         *0.965  0.013  0.011  0.011
     2 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
     3 3:out cong 3:out cong          0.01   0.012 *0.965  0.013
     4   4:normal   4:normal          0.016  0.017  0.019 *0.948
     5 1:into con 1:into con         *0.965  0.013  0.011  0.011
     6 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
     7 3:out cong 3:out cong          0.01   0.012 *0.966  0.012
     8   4:normal   4:normal          0.016  0.018  0.019 *0.948
     9 1:into con 1:into con         *0.966  0.014  0.01   0.01
    10 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    11 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    12   4:normal   4:normal          0.017  0.018  0.018 *0.947
    13 1:into con 1:into con         *0.966  0.015  0.01   0.009
    14 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    15 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    16   4:normal   4:normal          0.017  0.018  0.018 *0.947
    17 1:into con 1:into con         *0.966  0.014  0.01   0.01
    18 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    19 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    20   4:normal   4:normal          0.017  0.018  0.018 *0.947
    21 1:into con 1:into con         *0.965  0.013  0.011  0.011
    22 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    23 3:out cong 3:out cong          0.01   0.012 *0.965  0.013
    24   4:normal   4:normal          0.016  0.018  0.019 *0.948
    25 1:into con 1:into con         *0.966  0.014  0.01   0.01
    26 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    27 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    28   4:normal   4:normal          0.017  0.018  0.018 *0.947
    29 1:into con 1:into con         *0.965  0.014  0.011  0.011
    30 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    31 3:out cong 3:out cong          0.01   0.012 *0.966  0.012
    32   4:normal   4:normal          0.016  0.018  0.019 *0.948
    33 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    34 1:into con 1:into con         *0.966  0.014  0.01   0.01
    35 3:out cong 3:out cong          0.01   0.012 *0.966  0.012
    36 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    37   4:normal   4:normal          0.016  0.017  0.019 *0.948
    38 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    39 1:into con 1:into con         *0.966  0.014  0.01   0.01
    40 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    41 1:into con 1:into con         *0.966  0.014  0.01   0.01
    42 3:out cong 3:out cong          0.01   0.012 *0.965  0.013
    43 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    44 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    45 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    46 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    47 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    48 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    49 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    50 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    51 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    52 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    53 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    54 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    55 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    56 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    57 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    58 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    59 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    60 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    61 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    62 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    63 2:congeste 2:congeste          0.004 *0.989  0.002  0.004
    64 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    65 2:congeste 2:congeste          0.005 *0.989  0.002  0.004
    66 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    67   4:normal   4:normal          0.017  0.018  0.018 *0.947
    68 3:out cong 3:out cong          0.01   0.012 *0.966  0.012
    69 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    70 1:into con 1:into con         *0.966  0.014  0.01   0.01
    71   4:normal   4:normal          0.017  0.018  0.018 *0.947
    72 1:into con 1:into con         *0.966  0.015  0.01   0.009
    73 2:congeste 2:congeste          0.004 *0.989  0.002  0.005
    74 3:out cong 3:out cong          0.011  0.012 *0.966  0.012
    75   4:normal   4:normal          0.017  0.018  0.018 *0.947
    76 3:out cong 3:out cong          0.01   0.012 *0.966  0.012

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances          76              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0.0131
Root mean squared error                  0.0176
Relative absolute error                  3.8212 %
Root relative squared error              4.2675 %
Total Number of Instances               76

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1          1         1          1        into congested
                 1         0          1          1         1          1        congested
                 1         0          1          1         1          1        out congested
                 1         0          1          1         1          1        normal
Weighted Avg.    1         0          1          1         1          1

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 13  0  0  0 |  a = into congested
  0 36  0  0 |  b = congested
  0  0 15  0 |  c = out congested
  0  0  0 12 |  d = normal
```

Figure 4: Testing output results

Table 5: Evaluation on test set summary

```
=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances          76              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0.0131
Root mean squared error                  0.0176
Relative absolute error                  3.8212 %
Root relative squared error              4.2675 %
Total Number of Instances               76
```

In Figure 4 under "predictions on test split" shows prediction results for each of the instances loaded from an excel spreadsheet. The column heads are identified by inst# (instance number), actual, predicted, error and probability distribution. Underneath inst# column are all the 76 instances to be predicted. Actual column shows actual discriticised traffic status based on traffic properties of each instance. The predicted column shows predicted traffic status based on the prior knowledge the model gained during training. The error column identifies instances incorrectly classified. The probability distribution has four columns representing four classes to be predicted (from left to right: into congested, congested, out congested and normal). A plus (+) sign inside the error column identifies instances incorrectly classified. A plus (+) sign is not visible and as a result none of the instances were misclassified. Figure 5 was generated from plotting the instances shown underneath "predictions on test split" that have asterisks. 36 of these instances (with a maroon square) show 99% average success rate. These maroon squares instances are in the majority at this level. This high success rate is due to high probability of each of these instances shown in Figure 5 in the 2nd column of "probability distribution" compared to others instances in the 1st, 3rd and 4th columns. This means that traffic in the next 5 minutes is likely to be congested.

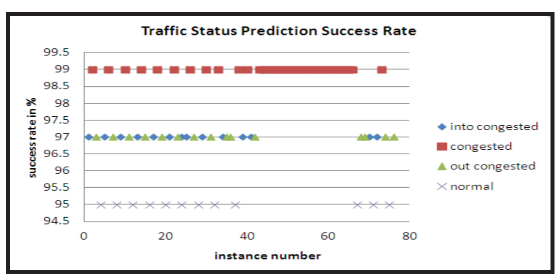


Figure 5: Instances with the highest success rate

In Figure 6 the margin curve shows the confidence level of the classifier in predicting all the instances. A margin curve is a plot illustrating the difference between probability predicted for the actual class and the highest probability predicted for the next most likely class for each instance. In Figure 6 instance was plotted against a margin value. Instances occupy margin values between -1 and 1. Any instances that lie on the negative region were misclassified by the model. The larger the margin (margin value closer to 1), the more confident the classifier in predicting the true class (traffic status: congestion, into congestion, out of congestion and normal). All instances in Figure 6 are occupying a value around 0.98 thus indicate more confidence in predicting the next most likely class.

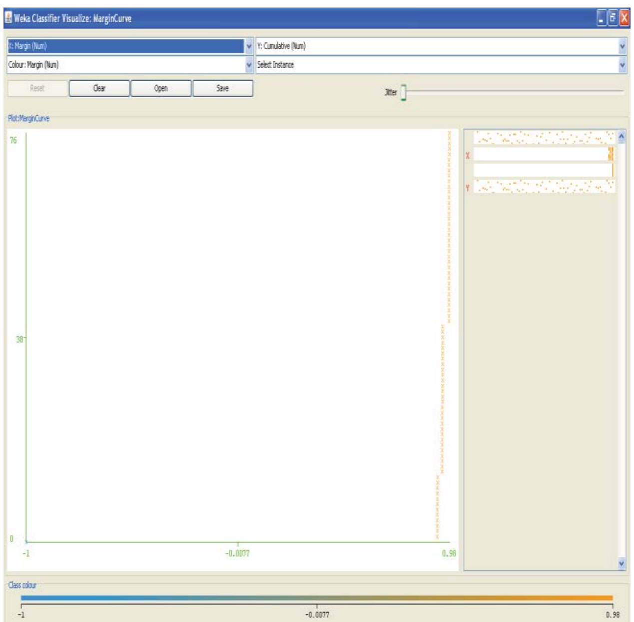


Figure 6: MLP margin curve

The accuracy of the algorithm in predicting each of the four classes is shown in Table 6. The four conditions are shown as classes in the "detailed accuracy by the class" table (Table 6). A "detailed accuracy by class" table makes use of a confusion matrix to determine the performance of the trained MLP model used in predicting each of the four classes (congested, into congested, out congested, normal). A confusion matrix shown in Figure 6 is a visualization tool typically used to present the results attained by the learner.

Table 6: Detailed accuracy of the model

```
=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1          1        1          1        into congested
                 1         0          1          1        1          1        congested
                 1         0          1          1        1          1        out congested
                 1         0          1          1        1          1        normal
Weighted Avg.    1         0          1          1        1          1

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 13  0  0  0 |  a = into congested
  0 36  0  0 |  b = congested
  0  0 15  0 |  c = out congested
  0  0  0 12 |  d = normal
```

Each row in a confusion matrix represents the instances in the actual class and each column represents the instances in a predicted class. A benefit of using confusion matrix is that it is easy to see if the system is confusing two classes i.e. mislabelling one class as the other class. Table 6 shows a 4x4 confusion matrix (a, b, c and d) representing into congested, congested, out congested and normal in actual and predicted classes respectively. The numbers from the top left corner down to the bottom right corner (13, 36, 15 and 12) represent a main (leading) diagonal. These numbers show all instances classified correctly for each predicted class. Any number(s) which appear above the main diagonal are called False Positive (FP) numbers i.e. it identifies the number of instances mislabelled as belonging to another different class. Any number(s) that appear below the main diagonal are called False Negative (FN) numbers. False negative numbers also identify incorrectly classified instances. The confusion matrix in Table 6 shows that the model made 76 correct predictions (13+ 36 + 15 + 12) as shown by the main diagonal numbers. There is no number above or below the main diagonal since the model predicted all 76 instances correctly.

The model accuracy is 100%, this value is confirmed by average True Positive (TP) rate of 1 outlined within "detailed accuracy by class" table in Table 6. Recall and precision rate in Table 6 measures the quality of the classification process. Recall refers to the portion of the positive examples retrieved during the classification process versus the total number of existing positive examples including positive examples not retrieved during classification. Recalls of 1 mean all positive (100%) examples were retrieved and classified as positive. As a result for all the four classes all positive examples were all retrieved. It is worth noting that the recall and TP rate values within “detailed accuracy by class” table in Table 6 are the same. This is due to the fact that the TP rate is a measure of positive examples which were classified as positive by the classifier.

Precision is defined as the portion of the positive examples that exist in the total number of examples retrieved i.e. out of all examples retrieved for each class how many were classified positive? 100% of examples were classified positive for congested, into congested, out congested and normal out of all examples retrieved as shown by a Precision value of 1 for each class.

Receiver Operating Characteristics (ROC) area shown inside “detailed accuracy by class” table in Table 6 is a metric used to gain insight into the decision making ability of the model i.e. how likely is it that the model can accurately predict the negative or the positive class? ROC measures the impact of changes in the probability threshold. The probability threshold is the decision point used by the model for classification. The default classification threshold for binary (positive and negative) classification is 0.5. When the probability of prediction is 50% or more, the model predicts the class. When the probability is less than 50% the other class is predicted. In a multiclass classification the predicted class is the one with the highest probability. Looking at the ROC column, a value of 1 shows that this trained MLP model decision making is 100% spot on in identifying positive and negative examples.

A wireless medium could then be used to display the predicted traffic status using already available LCD display boards installed along the highways.

V. CONCLUSION

This work successfully addresses the problem of traffic congestion along the Ben Schoeman highway. Having an intelligent system that employs the use of multilayer perceptron in predicting the future traffic status further contributes towards reduction in pollution and petrol usage normally experienced when cars idle for a very long times. Notifying the motorists about a traffic pattern to happen by using LCD billboards located on the highway ensures that the motorists spend less time on the road, the less time they spend means reduction in accidents due to fatigue and tiredness normally experienced when motorists take a long time on the road thus result in ineffectiveness at work places. Instead more time will be spent contributing towards companies’ production resulting in more profit for different organizations. The more profit different organizations obtain leads to high economic gain for the whole of South Africa.

REFERENCES

1. DEPARTMENT of Transport **see** REPUBLIC OF SOUTH AFRICA. Department of Transport.

2. REPUBLIC OF SOUTH AFRICA. Gautrain.2012. Annual report. Johannesburg: Government printer.

3. REPUBLIC OF SOUTH AFRICA. SANRAL.2011. Annual report. Johannesburg: Government printer.

4. Xu T, Sun X, He Y and Xie C. “Artificial Neural Network and Wavelet Analysis Application in Speed Forecast of Beijing Urban Freeway”, (IEEE 2009), P.R China.

5. Mao S, Chen Y and Xiao X. “Short-term Traffic Flow Prediction Based on GM (1,1,Exp) model”, (IEEE 2011), China.

6. Wang Y, Wang H and Xia L. “Highway Traffic Prediction with Neural Network and Genetic Algorithm”, (IEEE 2005), Beijing, China.

Using Modified Intelligent Water Drops Algorithm for QoS-driven Web Service Selection (MIWD-SS)

L. Mohammad Khanli[1], N. Danandeh[1], and M. Mohammad Khanli[1]
[1]Electrical and Computer Engineering Dept., University of Tabriz, Tabriz, Iran

Abstract - *A Modified Intelligent Water Drops algorithm based on local search and best result transferring for the next iteration is proposed for QoS-driven web Service Selection with global QoS constraints. A good web service compositions is a result of best available service selection method for composition. MIWD-SS is using IWD algorithm with two new modifications. One is transferring the best answer from the last iteration to the new one and the other one using local search based on simulated annealing algorithm. Simulations show that when used in web service selection, MIWD-SS is better than IDIPSO, SGA and QQDSGA in computing time and optimality with both small and large size of the composite web service problem. It even works better by increasing the number of classes or the number of available services in a class.*

Keywords: Web Service Selection, QoS, Water Drop Algorithm

1 Introduction

Web service is a software application identified by an URI. It is a new type of distributed computing model, with traits of self-contained, modular, etc. A single web service always provides some simple function and unable to meet the needs of complicated applications requirement. Therefore how to integrate simple web service to a new more powerful composited web service became an urgent.

In a web service composition issue [1] the question of how to construct a composite service scheme according to its QoS has become a research focus in recent years. Due to the growing number of candidate services that provide the same functionality but different QoSs, it became more complicated to select a combination of services with optimal QoS performance, while satisfying user's QoS constraints.

Different methods used in service composition like IP_QSC, MILP_QSC, MIPQCD_QSC, DQSC, etc. that some of them uses intelligent algorithms to solve the problem for example genetic algorithm GA. A metaheuristic algorithm can be classed as a constructive approach or a local search model. Evolutionary-based algorithms are local search algorithms whereas the ant colony-based algorithms are constructive algorithms.

Intelligent Water Drops algorithm [17] is a new population-based optimization algorithm that uses the constructive method to find the optimal solutions of a given problem. In this paper we use the IWD algorithm with a little modification to solve the QoS-aware service composition problem. We defined eight HUD heuristic functions to meet the global user QoS constraints in service selection. Also for increasing the power of global search in IWD algorithm and avoidance of local optimal problem, we used simulated annealing algorithm for local search in large scale problems by transferring the last best answer to the next iteration.

Experimental comparisons with three previous methods (IPIPSO, SGA, QQDSGA) shows that in terms of computation time and optimality our method is better than other methods.

The remainder of this paper is organized as follows. After a review of related works in section 2, section 3 presents the service selection problem and QoS parameters used. Section 4 describes how we used IWD algorithm to solve service selection problem. Section 5 show the simulations of proposed solution. Finally our conclusion and future works are given in section 6.

2 Related Works

The service selection problem can be divided into two categories. One is constructive approach or a local search model and the other is based on using constructive algorithms. In first methods [1-7] they are going to find accurate answer for problem, but by increasing the problem's size, they lost their efficiency. Zeng et al., [1] changed the service selection problem into the Integer Programing (IP). The problem of selecting an optimal execution plan is mapped into an IP problem. In continue Ardania et al., [2] considered the compound service's for each single service and changed this problem to Mixed integer linear programming (MILP). In [3], researchers, for solving the QoS-aware service selection problem, global QoS constraints of users were analyzed, then by using MILP method they disintegrate to single service QoS constraints. So by making the local limitations and increasing the discovered functional value for all duties, solving the limitations and also increasing the functional would be guaranteed. Then in [4], Alferiay et al., by using skyline concept, they presented a definition of skyline service. In base of skyline service is a service which any other services couldn't be better than it.

In [5], researchers solved the service selection problem by changing it to multi optional knapsack problem. In researches [6], the researchers solved the service selection problem by branch and limitation. Swan and Jaeo [7] presented a heuristic function by using mathematical formulas which it is possible to extract the single services efficiency from final composite service.
Approaches based on using constructive algorithms [8-16] usually seek optimum answer, and by attention to QoS-aware service selection problem is a NP hard problem these approaches can decrease the complexity and have a fine scalability. Generally to compound an efficient service, defining the scoring and suitability functions. In these approaches, the suitability function is made of two goal and fine functions, the goal function is used for evaluating the compound service efficiency and the fine function is used for fining the users' limitations' assault. Finally, a compound service is selected which the fine function value for it is 0 and the goal function value for it is optimum. The approaches based of constructive algorithm include the wide range of constructive algorithms like genetic algorithms [8], and its development [9], particle swarm optimization algorithm [10], Immune algorithm [11], ant colony optimization [12] and compound approaches like using tabu search and hybrid genetic [14], compounding the genetic algorithm and particle swarm optimization [15], particle swarm optimization and Artificial Immune System [16]. And usually are going to find the approximate answer close to optimum.
IWD algorithm [17] is a population-based optimization algorithm which the answers are made one by one and in detail. This algorithm has attracted the researchers' attention for optimizing the real world's problems which the important one is TSP [18], MKP [19], movement direction's planning of air-robot [20], Steiner tree [21], QoS based routing in network [22] and vehicle routing problem [23]. Therefore, in this article we present a modified IWD algorithm for solving the QoS-aware service selection problem and find the scalable solution in comparison with pervious works.

3 Qos Aware Service Selection Problem

In this part, at first we present the needed definitions for modeling the QoS aware service selection problem and finally define the composite service evaluation function.
Here we consider the replying time, cost, accessibility, reliability and executive success for the QoS parameters and by using these definitions, we model this problem.
Task: t_i task is the abstract show of user's request application which doesn't have executive mood, but its input, output and needed parameters are recognized.
Candidate service: the candidate service s_{ij} is a real web service of S_i service class which can do the t_i task.
Service class: a collection of candidate services with same function and different characteristics of QoS.

Service quality vector candidate: quality characteristics of candidate service S_{ij} would be shown in quality vector frame $Q_{s_{ij}}$ with equation (1):

$$Q_{s_{ij}} = \left\{ q_1\left(s_{ij}\right), q_2\left(s_{ij}\right), \ldots, q_t\left(s_{ij}\right) \right\} \quad \textbf{(1)}$$

Where t is the number of QoS parameters and $q_t(s_{ij})$ is the t th parameters of s_{ij} service quality.

Composite service: the composite service CS is a collection of task $T = \{t_1, t_2, \ldots, t_n\}$ which suitable service S_i for each task t_i from service class S_i is selected and these services are executed by executive engine (BPEL) to reply the complicated request of user which is made from T task collections.
Composite service quality vector: the quality characteristics of a compound service CS is shown as below.

$$Q_{CS} = \left\{ q_1'\left(CS\right), q_2'\left(CS\right), \ldots, q_t'\left(CS\right) \right\} \quad \textbf{(2)}$$

In that $q'_t(CS)$ is the t th value QoS of the composite service CS and by attention to structure of composite service we use AQF combining function [3] for the global quality with this equation.

$$q_k'\left(CS\right) = AQF_k\left(CS\right) \quad \textbf{(3)}$$

User's QoS constraints: the quality constraints of user is the lowest threshold of the composite service should be supplied.

$$\forall q_k \in QoS^+ : \; q_k'\left(CS\right) \geq C_k \quad \textbf{(4)}$$

$$\forall q_k \in QoS^- : \; q_k'\left(CS\right) \leq C_k \quad \textbf{(5)}$$

Where QoS^+ is the positive measures of QoS (availability, reliability and executive success), QoS^- negative service quality criteria (response time and cost) and C_k is the user constraints to measure the k th QoS in final composite service. More QoS^+ value increase the quality, more QoS^- value decrease the quality and vice versa.
Efficiency function: for measuring the efficiency of composite service CS, at first the quality of vector (6) and (7) equations would be normalized, then U(CS) function defines by using the normalized vector equation (8) and weighting technic [1] equation (9).

$$\forall q_k \in QoS^- : q_k'' = \frac{Q_k^{max}\left(CS\right) - q_k'\left(CS\right)}{Q_k^{max}\left(CS\right) - Q_k^{min}\left(CS\right)} \quad \textbf{(6)}$$

$$\forall q_k \in QoS^+ : q_k'' = \frac{q_k'\left(CS\right) - Q_k^{min}\left(CS\right)}{Q_k^{max}\left(CS\right) - Q_k^{min}\left(CS\right)} \quad \textbf{(7)}$$

$$NQ_{CS} = \left\{ q_1''\left(CS\right), q_2''\left(CS\right), \ldots, q_t''\left(CS\right) \right\} \quad \textbf{(8)}$$

$$U\left(CS\right) = \sum_{k=1}^{t} W_k \times NQ_{CS}\left(k\right) \quad \textbf{(9)}$$

QoS aware service selecting problem: The goal is to select QoS driven service, select suitable candidates from each class corresponding to the task t_i from S_i services available so that the sum of the final composite services to meet the constraints of C and maximize the performance of function U.

4 QoS aware service selection using modified IWD Algorithm

In this section the IWD algorithm generally described, then we will show steps of using modified IWD algorithm for QoS aware service selection.

4.1 IWD Algorithm

IWD algorithm based on the idea of water droplets that flow in rivers, lakes, and seas so that every water drop naturally find its way to the sea or ocean. In nature, countless water droplets together build the optimal route to reach their destination in other words, it is an intelligent mechanism based on population. IWD algorithm mechanisms using a lot of water droplets (IWD) to make different routes and finally find its optimal (or near optimal) path gradually over time.

The intelligent water drops algorithm, every drop of water (IWD), which represents a first null answer begins from starting point and selects next point from his environment and add them to his route and finally reach the destination and make his answer. In this algorithm, IWD prefers path with less soil to pass. Every IWD has an initial soil with it and while passing through the path based on his speed removes some of the path soil and add it to his own soil. More speed results more soil. After selecting its direction through the movement from one point of river to other point of it, amount of path's soil was washed and increased the transferring soil. This function calculates the inefficiency of movement from one point of direction to other point. Therefore, suitable replies for the special problem will be in less soil directions. And after a while, IWD had tendency to select the less soil direction which will be optimum direction.

4.2 Designing the function HUD

As mentioned, the HUD shows the degree of unsuitability of selected path for moving from one node to another node. According to the algorithm, IWD, this function plays an important role in the convergence of the algorithm, so that if properly designed, for any optimization problem, the nodes of high quality will be rated low degree.

For designing the suitable function designing HUD for service selection problem in base of QoS should consider the user's limitations and increasing the efficiency of final composite service.

Evaluating user's global limitations:

For designing the function HUD for evaluating the global limitations, we suppose that user's limitations are applied on composite services; that is, if a user's limitation, the time of answering less than 3 seconds and the final composite service made from 3 single services, the local applied limitations for each single services, the time of answering will be less than 3 seconds. Therefore, the local limitations can be accessed by equations of (10) and (11):

$$\forall q_k \in QoS^- : C'_k = \frac{C_k}{n} \quad \textbf{(10)}$$

$$\forall q_k \in QoS^+ : C'_k = \sqrt[n]{C_k} \quad \textbf{(11)}$$

While by equation (12), we defined the function which determines the distance value to reject the local limitations after selecting the single service Sj:

$$D\left(s_j\right) = \sum_{k \in QoS^-} W_k \times \frac{C'\left(k\right) - q_k\left(s_j\right)}{C\left(k\right)} + \sum_{k \in QoS^+} W_k \times \frac{q_k\left(s_j\right) - C'\left(k\right)}{C\left(k\right)} \quad \textbf{(12)}$$

While the value of D(sj) is negative, the worst possibility can be accessed for rejecting the local limitations for service class (Sj) sj by equation (13) and by using the equation (14) D (sj) can change it to positive values:

$$\forall s_j \in S_i : WD\left(S_i\right) = \sum_{k \in QoS^-} W_k \times \frac{C'\left(k\right) - Q_k^{max}\left(S_i\right)}{C'\left(k\right)} + \sum_{k \in QoS^+} W_k \times \frac{Q_k^{min}\left(S_i\right) - C'\left(k\right)}{C'\left(k\right)} \quad \textbf{(13)}$$

$$if\ WD\left(S_i\right) < 0 : D\left(s_j\right) = D\left(s_j\right) - WD\left(S_i\right) \quad \textbf{(14)}$$

Then the unsuitable discovery function H1 can be equation (15). The applied strategy in this function is made by selecting the local limitations with far distances which create the user's global limitations.

$$H_1 = \frac{1}{D\left(s_j\right)} \quad \textbf{(15)}$$

In equation (15) the undesirability of service selection has related inversely with the distance from the local constraints violation ($D\left(s_j\right)$); that is, the service with more distance to violate local restrictions, is most popular, and vice versa.

Maximize overall performance: to maximize the efficiency of the final composite service, we define equation (16) to calculate the efficiency of selection of single service s_j from class service S_i :

$$U\left(s_j\right)=\sum_{k\in QoS^-} W_k * \frac{Q_k^{max}\left(S_i\right)-q_k\left(s_j\right)}{Q_k^{max}\left(S_i\right)-Q_k^{min}\left(S_i\right)} + \sum_{k\in QoS^+} W_k * \frac{q_k\left(s_j\right)-Q_k^{min}\left(S_i\right)}{Q_k^{max}\left(S_i\right)-Q_k^{min}\left(S_i\right)} \quad \textbf{(16)}$$

As shown in equation (17) we can define H_2 heuristic function based on selecting of individual services with high-performance local constraints.

$$H_1=\frac{1}{U\left(s_j\right)} \quad \textbf{(17)}$$

HUD is the ultimate definition of equation (18) can be achieved QoS -based service selection problem targets:

$$HUD=H_1\times H_2 \quad \textbf{(18)}$$

4.3 Fitness Function

Fitness function F (equation (23)) for the solution of the problem of evaluating the service selection based on QoS, has the two parts of the objective function (equation (9)) to maximize the efficiency of service and final combined penalty function (equation (22)) to calculate penalties for each service user restrictions violate motion is made.

$$D\left(CS\right)=\sum_{k\in QoS^-} W_k \times \frac{q_k^{'}\left(CS\right)-C\left(k\right)}{Q_k^{max}\left(CS\right)-Q_k^{min}\left(CS\right)} + \sum_{k\in QoS^+} W_k \times \frac{C\left(k\right)-q_k^{'}\left(CS\right)}{Q_k^{max}\left(CS\right)-Q_k^{min}\left(CS\right)} \quad \textbf{(19)}$$

$$\mathrm{F}=\begin{cases} U & \text{if } D=0 \\ \left(\left(1-\alpha\right)\times U\right)-\left(\alpha\times D\right) & \text{else} \end{cases} \quad \textbf{(20)}$$

4.4 Proposed Algorithm

In the proposed solution, the search space is considered to lead to a graph $G=\langle N,E\rangle$ where N shows nodes and E is edges of the search space (Figure 1).

By using the HUD function presented in the section (4.2) in IWD algorithm, the steps of proposed methods are as below:

1. Initialization of static parameters (a_v ،b_v ،c_v ،a_s ،b_s ،c_s ،P_0 ، P_n ،Ps ،P_{IWD} ،$K(N_c)$ ،N_{IWD} ،InitSoil ،InitVel)

2. Initialization of dynamic parameters (empty list of IWD answer to V_c(IWD), setting the early pace and amount of each IWD soil in order to InitVel and InitSoil.)

3. For each IWD, randomly select a service s_i from the service class S_1 and add it to the list of selected services V_c(IWD).

4. For each unfinished IWD, perform the following steps:

4-1. Select the next service s_j from next service class with probability $P_i^{IWD}(j)$ defined in Equation (21) and add it to the list of services to meet.

$$P_i^{IWD}\left(j\right)=\frac{f\left(soil\left(i,j\right)\right)}{\sum_{k\notin V_c(IWD)} f\left(soil\left(i,k\right)\right)} \quad \textbf{(21)}$$

4.2. Calculate the speed of service IWD s_i to s_j service with equation (22)

$$vel^{IWD}\left(t+1\right)=vel^{IWD}\left(t\right)+\frac{a_v}{b_v+c_v\times soil^2\left(i,j\right)} \quad \textbf{(22)}$$

4-3. Calculate travel time from service s_i to service s_j by equation (23) and using the HUD introduced in section (4.2).

$$time\left(i,j;vel^{IWD}\left(t+1\right)\right)=\frac{HUD\left(i,j\right)}{\max\left(\varepsilon,vel^{IWD}\left(t+1\right)\right)} \quad \textbf{(23)}$$

4-4. To obtain the amount of soil washed from the edge (s_i, s_j) with equation (24) and add it to the soil that carries IWD by equation (25).

$$\Delta\mathrm{soil}\left(\mathrm{i},\mathrm{j}\right)=\frac{a_s}{b_s+c_s\times\left(time^2\left(i,j;vel^{IWD}\left(t+1\right)\right)\right)} \quad \textbf{(24)}$$

$$soil^{IWD}=soil^{IWD}+\Delta\mathrm{soil}\left(\mathrm{i},\mathrm{j}\right) \quad \textbf{(25)}$$

4-5 local updating the soil of (s_i , s_j) with equation (26)

$$soil\left(i,j\right)=\rho_0.soil\left(i,j\right)-\rho_n.\Delta\mathrm{soil}\left(\mathrm{i},\mathrm{j}\right) \quad \textbf{(26)}$$

5. Calculating the suitability of made answers by IWD and finding the best answers T^{IB} with equation (27)

$$T^{IB}=\arg\max_{\forall T^{IWD}} q\left(T^{IWD}\right) \quad \textbf{(27)}$$

6. Go to step 2 with the transition T^{TB} to the next iteration until the maximum number of iterations ($iter_{Max}$) is reached.

7. The algorithm terminates with a solution (final composite service) T^{TB}.

Finally the MIWD1_SS and MIWD3_SS are proposed. These two approaches utilize HUD function which was presented in section (4.2), by this difference, MIWD1_SS, just the first strategy of soil transfer and MIWD3_SS from both first and second strategies were used for transferring the soil.

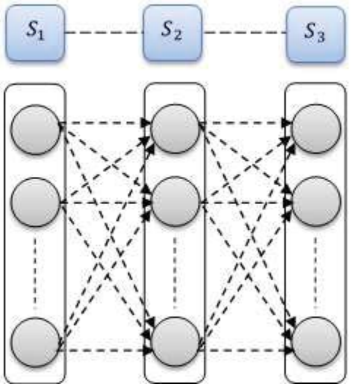


Figure 1: Disjunctive graph search space of a composite service consists of duty

5 Evaluation

For evaluating the proposed methods we use a randomly generated data set. This data set contains 5000 web services with 6 quality of service (as standard response time (Rt), price (Pr), accessibility (Av), a successful executive (Sc) and reliability (Re)). The parameters used in the proposed solution in Table 1 is obtained.

Table 1: proposed method's parameters

Value	Parameter
N_{IWD}	**20**
iterMax	**300**
InitSoil	**50**
InitVel	**500**
P_0	**5**
P_n	**3**
P_s	**1.9**
P_{iwd}	**1**
a_v / b_v / c_v	**1/0.01/1**
a_s / b_s / c_s	**1/0.01/1**
t	**5**
α	**0.5**
N_{Run}	**30**

The proposed strategies and solutions compared with using MATLAB software version 7.12.0 (R2011a) on a computer with a processor Intel Core2 Duo 2.53 GHz, 2GB memory and operating system Windows 8.1 x32 Enterprise.To compare the efficiency of the proposed solutions, three approaches SGA [8], QQDSGA [13] and IDIPSO [16] used.
In this experiment, we suppose fixed number of tasks (in this experiment, 10) and increase the number of services with each service class (in this test from 100 to 500).
As shown in Figure (2) and (3) we can see , the number of service - in candidate , MIWD1_SS proposed solution compared to other solutions (including MIWD3_SS), significantly less time to optimum (near optimal) do so.
IDIPSO strategy has significant increase in runtime and optimality reduced to about 70% by increasing the number of services. The reason for this increase in time to IDIPSO is using the first-best local strategy [16] which is dependent on the number of services per Class of Service.

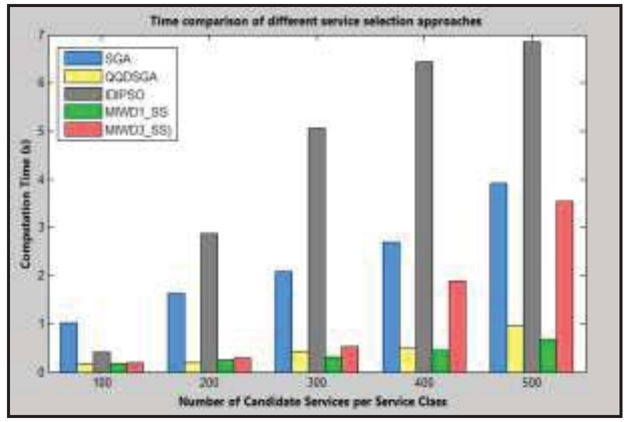


Figure 2: Scalability with computation time

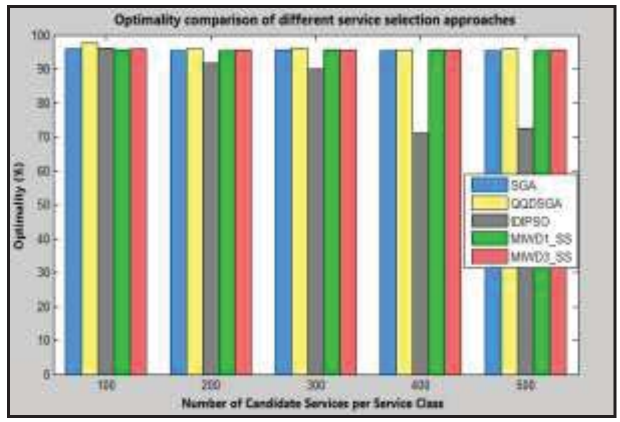


Figure 3: Scalability with optimality

6 Conclusion

Through the present article in base of the modified IWD for selection services problem in base of QoS was presented. The applied changes for dealing with IWD algorithm for selecting the service the HUD function included metal melting in compounding IWD which the changes lead to present the MIWD1_SS and MIWD3_SS. Considerably, the scalable is high in front of the others. As the future research, the proposed research will be proposed for super-localization by considering the cloud computational characteristics.

7 References

[1] Zeng, L., Benatallah, B., Ngu, A. H., Dumas, M., Kalagnanam, J., & Chang, H. (2004). QoS-aware middleware for web services composition. Software Engineering, IEEE Transactions on, 30(5), 311-327.

[2] Ardagna, D., & Pernici, B. (2007). Adaptive service composition in flexible processes. Software Engineering, IEEE Transactions on, 33(6), 369-384.

[3] Alrifai, M., & Risse, T. (2009, April). Combining global optimization with local selection for efficient QoS-aware

service composition. In Proceedings of the 18th international conference on World wide web (pp. 881-890). ACM.

[4] Alrifai, M., Skoutas, D., & Risse, T. (2010, April). Selecting skyline services for QoS-based web service composition. In Proceedings of the 19th international conference on World wide web (pp. 11-20). ACM.

[5] Yu, T., Zhang, Y., & Lin, K. J. (2007). Efficient algorithms for Web services selection with end-to-end QoS constraints. ACM Transactions on the Web (TWEB), 1(1), 6.

[6] Liu, M., Wang, M., Shen, W., Luo, N., & Yan, J. (2012). A quality of service (QoS)-aware execution plan selection approach for a service composition process. Future Generation Computer Systems, 28(7), 1080-1089.

[7] Sun, S. X., & Zhao, J. (2012). A decomposition-based approach for service composition with global QoS guarantees. Information Sciences, 199, 138-153.

[8] Canfora, G., Di Penta, M., Esposito, R., & Villani, M. L. (2005, June). An approach for QoS-aware service composition based on genetic algorithms. In Proceedings of the 7th annual conference on Genetic and evolutionary computation (pp. 1069-1075). ACM.

[9] Ma, Y., & Zhang, C. (2008). Quick convergence of genetic algorithm for QoS-driven web service selection. Computer Networks, 52(5), 1093-1104.

[10] Wang, W., Sun, Q., Zhao, X., & Yang, F. (2010). An improved particle swarm optimization algorithm for QoS-aware web service selection in service oriented communication. International Journal of Computational Intelligence Systems, 3(sup01), 18-30.

[11] Gao, Y., Na, J., Zhang, B., Yang, L., Gong, Q., & Dai, Y. (2006). Immune algorithm for selecting optimum services in web services composition. Wuhan University Journal of Natural Sciences, 11(1), 221-225.

[12] Zhang, W., Chang, C. K., Feng, T., & Jiang, H. Y. (2010, July). QoS-based dynamic web service composition with ant colony optimization. In Computer Software and Applications Conference (COMPSAC), 2010 IEEE 34th Annual (pp. 493-502). IEEE.

[13] Gao, Z. P., Chen, J., Qiu, X. S., & Meng, L. M. (2009). QoE/QoS driven simulated annealing-based genetic algorithm for Web services selection. The Journal of China Universities of Posts and Telecommunications, 16, 102-107.

[14] Parejo, J. A., Fernandez, P., & Cortés, A. R. (2008). Qos-aware services composition using tabu search and hybrid genetic algorithms. Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos, 2(1), 55-66.

[15] Hu, C. H., Chen, X. H., & Liang, X. M. (2009). Dynamic services selection algorithm in Web services composition supporting cross-enterprises collaboration. Journal of Central South University of Technology, 16, 269-274.

[16] Zhao, X., Song, B., Huang, P., Wen, Z., Weng, J., & Fan, Y. (2012). An improved discrete immune optimization algorithm based on PSO for QoS-driven web service composition. Applied Soft Computing, 12(8), 2208-2216.

[17] Shah-Hosseini, H. (2007, September). Problem solving by intelligent water drops. In Evolutionary Computation, 2007. CEC 2007. IEEE Congress on (pp. 3226-3231). IEEE.

[18] Shah-Hosseini, H. (2009). The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm. International Journal of Bio-Inspired Computation, 1(1), 71-79.

[19] Shah-Hosseini, H. (2008). Intelligent water drops algorithm: A new optimization method for solving the multiple knapsack problem. International Journal of Intelligent Computing and Cybernetics, 1(2), 193-212.

[20] Duan, H., Liu, S., & Lei, X. (2008, June). Air robot path planning based on intelligent water drops optimization. In Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on (pp. 1397-1401). IEEE.

[21] Noferesti, S., & Shah-Hosseini, H. (2012). A Hybrid Algorithm for Solving Steiner Tree Problem. International Journal of Computer Applications, 41(5), 14-20.

[22] Sensarma, D., & Majumder, K. (2014, January). IWDRA: An Intelligent Water Drop Based QoS-Aware Routing Algorithm for MANETs. In Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013 (pp. 329-336). Springer International Publishing.

[23] Kamkar, I., Akbarzadeh-T, M. R., & Yaghoobi, M. (2010, October). Intelligent water drops a new optimization algorithm for solving the vehicle routing problem. In Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on (pp. 4142-4146). IEEE.

[24] Cardoso, J., Sheth, A., Miller, J., Arnold, J., & Kochut, K. (2004). Quality of service for workflows and web service processes. Web Semantics: Science, Services and Agents on the World Wide Web, 1(3), 281-308.

Building Energy Modeling Using Non-Linear Auto Regression Neural Networks

N. Nassif[1], **S. Hamoush**[1], **D. Yoon**[2]

[1]Department of Civil, Architectural, Environmental Engineering, North Carolina A&T State University, Greensboro, NC, USA
[2]Department of Architectural Engineering, Gachon University, Seongnam-si, Korea

Abstract – *The paper discusses the modeling methodologies for building energy system using non-linear auto regression artificial neural networks. The model predicts whole building energy consumptions as a function of four input variables, dry bulb and wet bulb outdoor air temperatures, hour of day and type of day. To train and test the models, data from two existing buildings and from simulations are collected and used. The data are pre-processing using wavelet basis to remove the noise and anomalies. Different neural network structures are then tested along with various input delays to determine the one yielding the best results. The results show that the model can predict the energy consumptions accurately and it can be then used for various energy efficiency and saving estimation applications*.

Keywords: Building energy model, neural network, wavelet transfer, HVAC system, regression model.

1 Introduction

According to U.S. Energy Information Administration EIA [1], today buildings in the U.S. consume 72 percent of electricity produced, and use 55 percent of U.S. natural gas. Buildings account for about 40 percent of the energy consumed in the United States, more than industry and transportation. Of this energy, heating and cooling systems use about 55 percent. If energy-use trends continue, buildings will become the largest consumer of global energy by 2025. The development of building energy savings methods and models becomes apparently more necessary for a sustainable future. Although today modern buildings are mostly equipped with advanced building automation system (BAS) with the ability of collecting a large amount of data, those buildings still do not operate optimally due to the lack of embedded computational means and centralized solutions. Therefore, there is a significant need to investigate how modern computational techniques can help generate the analysis needed to gain full benefit from the available online data and at the same time perform many potential intelligent applications such as modeling, optimization, energy efficiency and energy assessment, and fault detection and diagnosis [2-5]. The author's overall objective is to develop energy efficiency solution tools for building energy systems through computational intelligence approaches and computational methods for data analysis. In the process of developing those tools, data-based models are developed. This paper presents the model using time series data available from building-level power meters. Although the aim of the model is to be integrated into the energy solution tools for building energy assessment and fault detection and diagnosis, the model can be also used for many other applications such as for establishing baselines and calculating retrofit savings.

Many researchers have investigated data-based models for building energy systems such as single variate and multivariate change point [6-7], and Fouries series [8]. A single or multivariate regression model is widely used as a means of monitoring building energy consumptions, identifying energy efficiency measures and O&M problems, and development of baseline model in energy conservation measurement and verification projects [9].
The steady state model based on artificial neural network are also used for modeling heating, ventilation, and air conditioning HVAC components [10] and estimation of energy savings for building retrofits [11]. However, in this paper, a dynamic non-linear model based on auto regression neural network is proposed. Various structures of ANN are investigated to determine the best one that could produce best accurate results. The paper also introduces data analysis methods using wavelet analysis for pre-processing, de-noising, and potential compressing the training data.

2 Methodology

To achieve our objectives, the following methodology is used: (1) data collecting and preprocessing, (2) developing, training, testing, and refining the models using artificial neural network ANN methods, and (3) integrating the developed model into the tools for building energy assessments and other intelligent functions. The whole building level modeling and data analysis are the main focus in this paper. The data are collected from two real buildings selected from a university campus, covering eight months. In addition, an existing building is also modeled using well-known energy simulation software eQuest [12] to generate hourly energy consumptions "simulated or synthetic data" for further analysis. The actual data and simulated data are used for model training and testing.

3 Data collection and preprocessing

The 25,000 ft^2 three-floor office and class room building is modeled by the energy simulation software eQuest. The hourly total energy consumptions are collected along with weather conditions for Greensboro, NC. The simulated data are divided into two sets for model training and testing. Other data collected are outdoor air dry bulb and web bulb temperatures, type of day (weekend/weekday/holidays), and the hour of day. Those data are the model inputs as discussed in model section below. Figure 1 shows the hourly energy consumptions as a function of outdoor air temperatures. Three different trends are noted, occupied period which is between 8 am and 5pm, unoccupied period (between 5 pm and 7 am) and holidays, and HVAC system start and stop period. HVAC system turns on at 7am, one hour before occupied period, and it turns off at 4pm, one hour before the end of occupied period, but the ventilation system stays on between 4pm to 5pm.

For further analysis, two buildings from a university campus are selected. The data are collected on a five-minute interval for eight months. To exclude the noise and anomalies, the data is de-noised using wavelet transfer. Comparing Fourier transfer, the wavelet is simply another expansion basis for representing a given signal or function. The power of the wavelet basis is its ability to take a function or signal and express it as a limit of successive approximations, each of which is a finer and finer version of the function in time. These successive approximations correspond to different resolution levels. Figure 2 shows the energy consumption data for one building and for 30 days using 'HAAR' transfer function and with five levels of resolution. The data are smoothed by removing high frequency signals (using threshold settings of 2.4 for high frequency signals d_1 and d_2 in Figure 2). Figure 3 shows the de-noised and actual energy data for two buildings, called building#1 (Blg#1) and building #2 (Blg#2). The smoothed signals are used for the model training. Other advantage of wavelet analysis is that the data can be compressed by retaining high percentage of energy (e.g. 99% of energy).

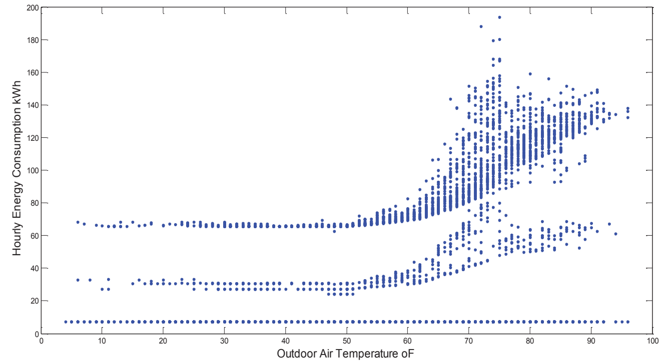


Figure 1. Hourly energy consumptions as a function of outdoor air temperatures for the simulated building

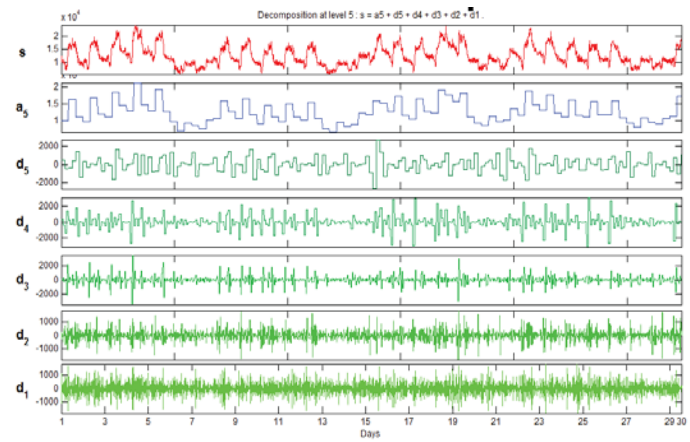


Figure 2. Data analysis for one building and 30 days (five min intervals) using 'HAAR' transfer function and with five levels of resolution

4 Modeling Using ANN

The most important factor in developing the energy solution tools is the need for accurate dynamic models. Depending on the type of functions and the accuracy required, the models can vary from simple to more detailed and sophisticated calculations [13]. However, it is of practical importance to develop simple, yet accurate and reliable models to better capture the real dynamic behavior of the subsystems and overall system over the entire operating range. In this paper, non-linear time series auto regression artificial neural networks are used. A large set of various neuron networks structures with different time delays are tested in order to determine the best and simple structures yielding adequate accuracy in term of mean square errors or coefficient of variances COV.

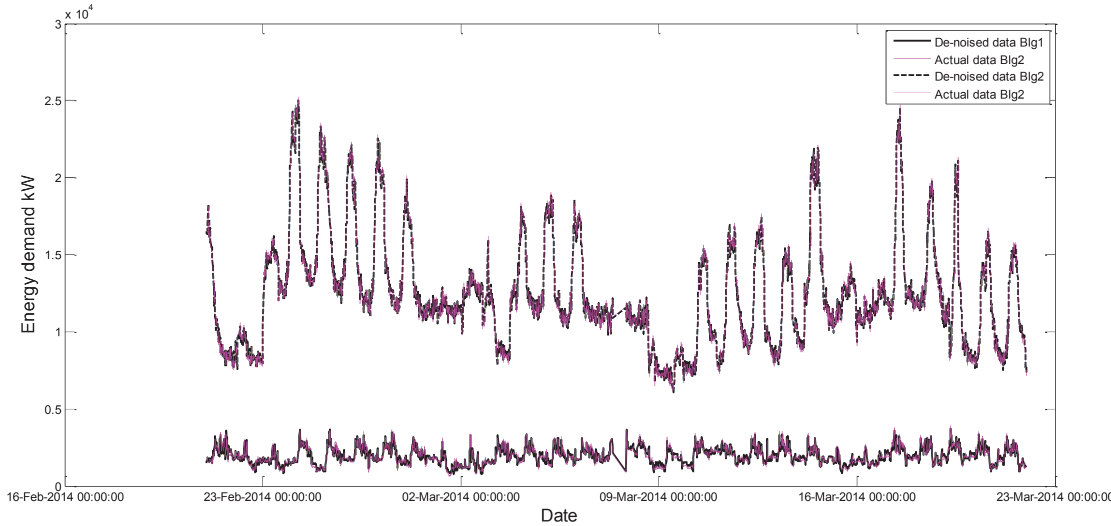


Figure 3. Actual and smoothed energy consumptions for two buildings

Artificial neural networks are computational models that are inspired by the natural neurons of the brain. As showing in Figure 4, natural neurons receive signals through synapses located on the membrane of a neuron. When the signals that are received are strong enough, the neuron is activated and emits a signal through to the axon. That signal may be sent to another synapse, and also might trigger other neurons for activation. The strength of the neurons' interconnections is called the adaptive weights. These are numerical parameters that are tuned by a learning algorithm. The higher a weight of an artificial neuron, the stronger the input which multiplied by it will be. Weights can also be a negative value, so signals may inhibited by a negative weight. Depending on the weights, the computation of the neuron in a network will be different. Adjusting the weights can produce the outputs needed for specific inputs.

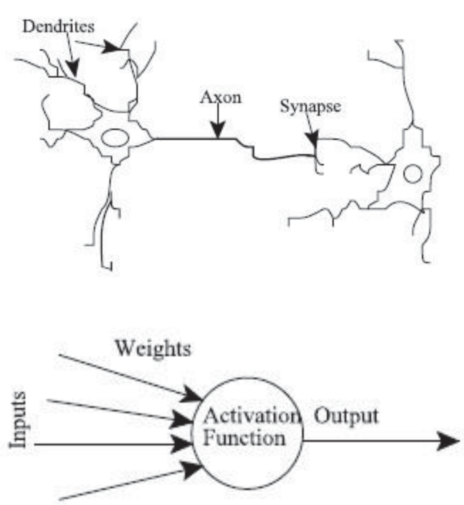


Figure 4. An artist conception of a natural neuron (top) and an artificial neuron (bottom)

Neural networks require appropriate learning algorithms to be trained. When a neural network is trained with a given set of data, it builds a predictive model based on the data. This model reflects a minimization in error when the network's out prediction is compared with the known outcome. Each learning algorithm has its own set of error-correction rules for reducing this error.

Figure 5 shows the schematic of non-linear time series auto regression artificial neural netwrok used in this paper. The model inputs are outdoor air dry bulb temperature, web bulb temperature, type of day (weekend/weekday), and the hour of day. The model output is building level energy consumption. The type of day variable is either "zero" or "one" corresponding to weekdays or weekend and holidays. The hour of day varies from 1 to 24 hrs. The best input delays n will be determined in this study based on the minimum mean square error during the testing period.

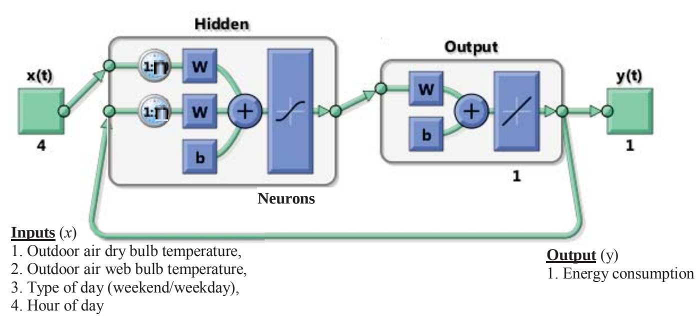


Figure 5. A schematic of non-linear time series auto regression artificial neural netwroks

5 Results

The synthetic data are divided into two sets (1) training set, from Jan 1st to August 31st and (2) testing set from

September 1st to December 31st. Various ANN structures are tested with various input delays in order to find the best structure. Figure 6 shows the resulted mean square errors for different number of neurons and input delays for testing and training period.

The test has been done for input time delays varying from 1 to 6 and number of neurons varying from 5 to 120 with an increment of five. Only some of those results (best results) are depicted in Figure 6.

It is generally noted that the performance in term of mean square error of each network improves with an increased number of hidden neurons in the training data set but this does not necessary produce better results in the testing data set. Indeed, the performance increase will be diminished sometime after 65 neurons. For instance, for the time input delay of one, the MSE for the training data drops from 29.93 to 10.16 when the number of neurons increases from 5 to 115. However, the MSE for testing data drops to minimum (i.e. 13.5) at 65 and then starts to increase with higher number of neuron. In addition, increasing the input delay leads to improve the results on the training data but not on testing data. As a result, for this investigated building, the delay input of one (one hour time step) holds best results on the testing data.

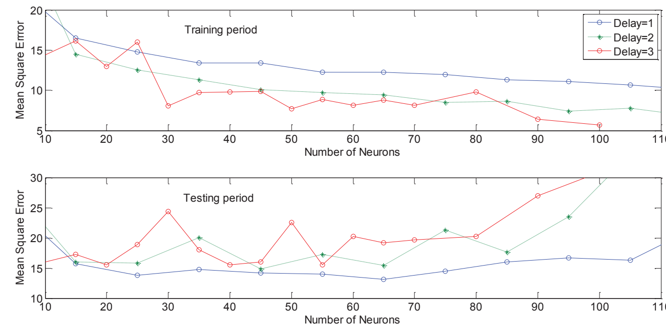


Figure 6. Mean square errors for different number of neurons and input delays for testing and training data

Figure 7 shows actual and predicted energy consumptions for training and testing sets. Two weeks for training and two weeks for testing data are shown in Figure 7. The results shown in this figure are corresponding to best neural network structure and input delay which are 65 neurons and the input delay of one. The days are counted from Jan 1st (day =1) to December 31st (day=365). For instance, the number of 236 in figure 8 represents Monday, September 25th, 2014. The model produces accurate results. The coefficient of variance (CV) is used as a statistical index for the model accuracy. The CV is 0.22% for training data and 0.24% for testing data.

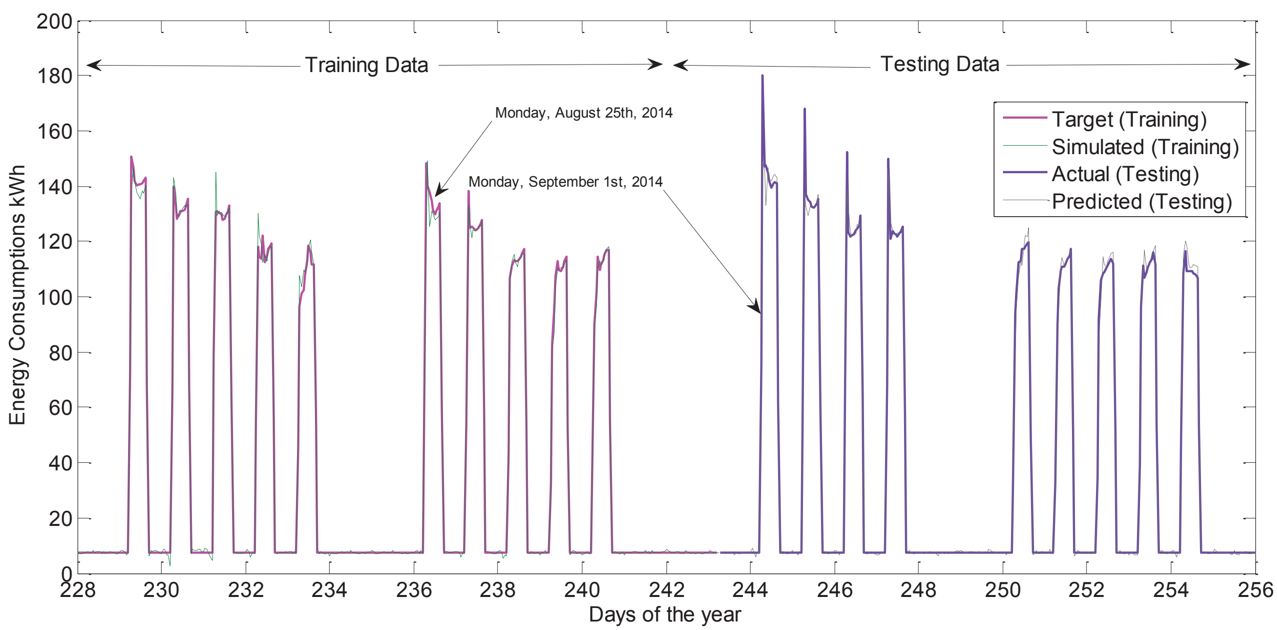


Figure 7. Actual and predicted energy consumptions for testing and training sets

The works are repeated, covering the two existing buildings (Blg#1 and Blg#2 as shown in Figure 3). The actual data are divided into training set of five months (from Feb to June 2014) and testing set of three months (July, August, and September 2014). The coefficient of variance for the training and testing data are calculated. The COV for Building 1 is 0.24% for training and 0.34% for testing data. The COV for Building 2 is 0.23% for training and 0.29% for testing data. These results are for the best ANN structures, which are 55 neurons for building#1 and 60 neurons for building#2. Time delay of one hold best results for both buildings. The results, from simulations and actual data on two buildings, show that the best ANN structures are within 50-65 neurons and time delay of one hour. The model can predict energy consumptions quite accurate with COV of less than 0.34% and it can be used for many energy efficiency and saving estimation applications.

6 Conclusions

This paper presents modeling methodologies for building energy systems using non-linear auto regression artificial neural networks. The model predicts whole building energy consumptions as a function of four input variables, dry bulb and wet bulb outdoor air temperatures, hour of day and type of day. Data from simulations and actual buildings are used for model training and testing. To exclude the noise and anomalies, the data are de-noised using wavelet transfer. Different neural network structures are tested along with various input delays to determine the one yielding the best results in term of mean square errors. The model with neurons of between 50 and 65 and with a time delay of one hour holds the best results. The testing results show that the coefficient of variance is 0.24% for the simulated building, 0.34% for Building #1, 0.29% for Building#2. The model can provide accurate prediction of building energy consumptions and that can be used for many applications such as energy building assessment, fault detection and diagnosis, energy saving estimation, and saving measurement and verifications.

7 Acknowledgments

This work was performed under the following financial assistant award 70NANB14D1319 from the U.S. Department of Commerce, National Institute of Standards and Technology

8 References

[1] U.S. Energy Information Administration EIA, www.eia.gov

[2] ASHRAE.. ASHRAE Handbook-Applications. Chapter 41. Atlanta: American Society of Heating Refrigeration and Air Conditioning Engineers, Inc. 2011.

[3] Seem, J.E. "Using intelligent data analysis to detect abnormal energy consumption in buildings". Energy and Buildings 39 52–58, 2007.

[4] Wang, S.W., Zhou Q. and Xiao F. "A System-level Fault Detection and Diagnosis Strategy for HVAC Systems Involving Sensor Faults". Energy and Buildings, V42(4), pp. 477-490, 2010.

[5] Nassif, N. "Single and Multivariate Regression Models for Estimating Monthly Energy Consumptions in Schools in Hot and Humid Climates". Energy Engineering 110 (5), 33-54, 2013.

[6] Katipamula, S., T.A. Reddy, and D.E. Claridge. "Multivariate regression modeling". *Journal of Solar Energy Engineering* 120:177-184, 1998.

[7] Kissock, J.K., T.A. Reddy, and D.E. Claridge. "Ambient-temperature regression analysis for estimating retrofit savings in commercial buildings". *Journal of Solar Energy Engineering* 120:168-176, 1998.

[8] Dhar, A., T.A. Reddy, and D.E. Claridge. "Modeling hourly energy use in commercial buildings with Fourier series functional forms". *Journal of Solar Energy Engineering* 120:217-223, 1998.

[9] ASHRAE. ASHRAE Handbook-Fundamentals, SI Edition,. Atlanta: American Society of Heating, Refrigerating, and Air-Conditioning Engineers ASHRAE, Chapter 19, 2013.

[10] Nassif, N. *"Modeling and Optimization of HVAC systems using Artificial Neural Network and Genetic Algorithm."* International Journal of building Simulation 7 (3):237.245, 2014

[11] Krarti, M., J.F. Kreider, D. Cohen, and P. Curtiss.. "Estimation of energy savings for building retrofits using neural networks". *Journal of Solar Energy Engineering* 120:211-216, 1998.

[12] eQuest. **QU**ick **E**nergy **S**imulation **T**ool, eQUEST Version 3.65. http://www.doe2.com/equest/

[13] Nassif, N. "Self-Tuning Dynamic Models of HVAC System Components". *Energy and Buildings 40 (2008):* 1709–1720, 2008.

A NOVEL METAHEURISTIC APPROACH FOR TRAVELLING TOURIST PROBLEM WITH TIME WINDOWS: A SINGLE DAY IN ISTANBUL

Timur Keskintürk[1], Eyüp Çetin[2], and Bahadır Fatih Yıldırım[2]
[1]School of Business, Istanbul University, Istanbul, Turkey
[2] School of Transportation & Logistics, Istanbul University, Istanbul, Turkey

Abstract - *In this study, we aim to suggest and solve employing a novel metaheuristic approach a tourist trip design program within a day covering 34 POIs in Istanbul where the continents Europe and Asia meet, as an attractive tourism center. We set two objectives with some constraints and two time windows in the problem. In order to solve the problem, we use a genetic algorithm based metaheuristic.*

Keywords: Travelling Salesman Problem, Travelling Tourist Problem, Metaheuristic, Istanbul.

1 Introduction

Tourists that visit one city, region or destination during a trip of a limited time (one or several days), facing the problem to visit all point of interests (POIs) that exist in that tourism destination, thus it is not possible to visit every POIs during such a limited period The problem includes a number of constraints such as the required visiting time for each point, time for travelling and the traveling distance among POIs. Thus, tourists need to plan their trip rationally. Doing the plan of a trip that includes most POIs to visit, for the available time or using time rationally is usually a complex task. In Operations Research / Management Science literature, this problem is named as the Tourist Trip Design Problem (TTDP). The TTDP refers to a route-planning problem for tourists interested in visiting multiple POIs.

2 Related literature

There are some related approaches and models in the literature. Some representative studies are as follows. Souffriau et al. [1], solve the problem involves a set from the city of Ghent of possible locations having a score and the objective is to maximize the total score of the visited locations, below the available time budget. Vansteenwegen et al. [2] designed a personalized tourist trip and modelled as a Team Orienteering Problem (TOP) with time windows (TOPTW). Guided local search (GLS) and variable neighborhood search (VNS) algorithms are applied to efficiently solve the TOP. Iterated local search (ILS) is implemented to solve the TOPTW. The GLS and VNS algorithms are compared with the best known heuristics and applied on large problem sets [2]. Also, Garcia et al. [3] model the tourist planning problem, integrating public transportation, as the Time Dependent Team Orienteering Problem with Time Windows (TDTOPTW) and they present an heuristic able to solve it on real-time. As a conclusion, they show the prototype which generates and customizes routes in real-time [3]. Hasuike et al. [4], proposed a new personal tour planning problem with time-dependent satisfactions, traveling and activity duration times for point of interests and formulated as a 0 - 1 integer programming problem [5].

Sylejmani and Dika [6], introduce an algorithm that plans a touristic trip with 40 instances of point of interests of the city of Vienna by considering some hard and soft constrains. Also, Rodríguez et al. [7], purposed a tool that provides each tourist with the itinerary best suited to their needs and wishes along with the characteristics of the area. Tool developed by using a mathematical model and interactive multi-criteria techniques.

In 2012 studies, Sylejmani et al. [8] use Tabu Search Algorithm solving the Multi Constrained Team Orienteering Problem with Time Windows whereas [5], modeled a tourist trip design problem as an orienteering problem and formulate in a time-dependent network. In paper, label correcting algorithm (LCA) is used to solve the problem based on the idea of network planning and dynamic programming.

3 A real world application: A single day in Istanbul

In this effort, we model the TTPTW problem for a single day in Istanbul as an illustrative example. We assume that a specified hotel is a starting point and it should be leaved at a specified hour after having the breakfast. The POI(s) given should be visited between the hotel departure hour and the lunch. After the lunch break, the visit begins again up to the dinner as a second part of the daily tour. The arrival to the hotel should be provided as soon as after having the dinner at a chosen place.

We group POIs in three categories; the first one is dining (lunch and dinner) places, the next is historical and cultural places and the last group is touristic outdoors. These places are selected from the most popular touristic sites in Istanbul. The number of nodes is given in Table 1.

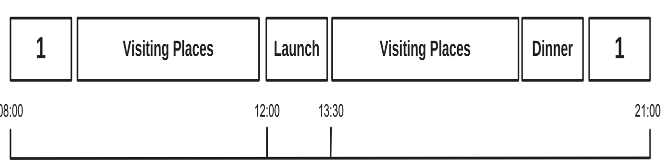
Table 1. Number of Nodes

Group	Name	# of Nodes
1	Dining	18
2	Cultural and Historical	16
3	Outdoors	7

We select two nodes from the first group of dining among all 41 nodes. These two nodes do not have to be distinct places. Historical and cultural and outdoor places may be visited in morning or afternoon and for those, we employ two time windows in which all specified visits should be realized between these time intervals. More concretely, the first time window is 330 minutes length departing form the hotel at 8 am to the end of the lunch at 1.30 pm. The last 90 minutes of the first time window is for the lunch and the assignment in this part of the tour should be chosen from the first group places only.

The second time window is 450 minutes length from the end of the lunch at 1.30 pm to the arrival to the hotel at 9 pm. Again the last part of the window is for the dinner, which may only be selected from the first group of dining. Both time windows are illustrated in Figure 1.

Figure 1. The Time Windows



For the illustrative application, the real world data is obtained from the Google Maps at https://maps.google.com/ [9]. We arrange the transportation smart way distance data between each of 42 nodes (41 POIs and the hotel). Also, the visit times at the places are estimated and assumed to be the same for the similar places.

We introduce and define the Problem I and Problem II, which differ in objective function building within the application. The objective in the Problem I is the minimization of the slack times out of the transportation and visit times whereas in the Problem II the maximization of the number of nodes - POIs- visited. In the second approach, the goal is the visiting as much as POIs within the time windows limitations in the morning and afternoon.

4 A novel metaheuristic for TTPTW

To solve the problem we have used genetic algorithm based metaheuristic. Genetic algorithm is one of the most known metaheuristic techniques. It is population based random search procedure that starts with random initial population. In our problem we also use random initial population. For substrings that represent visited nodes algorithm chooses from nodes depends on location of genes. After population is created fitness values evaluate. There are two different type of fitness function named as Type I and Type II. Then population send to the selection operator to get better population according to their fitness values. Mutation has an important role to search solution space. According to chromosome it has 4 different part. Two parts of the chromosome represents visited nodes. With a mutation probability the nodes that visited may withdraw from chromosome or nodes may add to the chromosome that they are not in the chromosome (Figure 2). The other two substrings represent launch and dinner. Again, with a probability of mutation they exchange with using the first list that includes restaurants (Figure 2). This process runs until end condition terminated. The pseudo-code of the metaheuristic is presented in Fig. 3.

Figure 2. The Proposed Chromosome and Operators

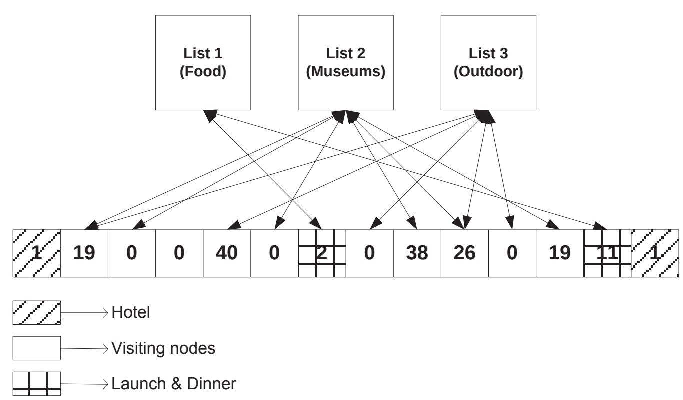


Figure 3. The pseudo-code of the metaheuristic

METAHEURISTIC

STEP 0: Generate an initial population

STEP 1: Evaluate the fitness value of the chromosomes

STEP 2: Perform selection operation and give those individual that have better fitness values a more chance to survive in the next generation.

STEP 3: Perform mutation operation according to probabilities.

STEP 4: Repeat steps 1, 2, 3 until the Metaheuristic is run for the predetermined number of generations

STEP 5: Select the best chromosome.

5 Implementation of the application

The problem on which we study is the TTPTW. A total number of nodes that we may choose for a tour is 41. A metaheuristic approach based on GA is applied to the problem. Initial population is created randomly. The number of chromosomes is 20. Roulette well operator is selected as a selection. Mutation rate is 0.01 and number of iteration is 50,000. For two different types of the problem, algorithm runs 100 times. The results of the problems are presented in Table 2.

Table 2. The Results for Both Problems

Problem Type	Min	Mean	Max	Standard Deviation	Min	Mean	Max	Standard Deviation
I	Slack time (min)				Tour duration (minutes)			
	0.12	0.16	0.33	0.05	779.66	779.83	779.88	0.05
II	Number of visited nodes				Tour duration (minutes)			
	8	8.57	9	0.50	626.00	736.29	776.87	37.69

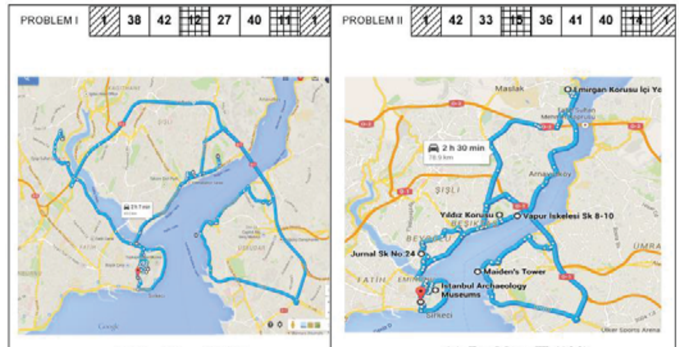
As it is clearly seen in Table 2, the findings of our proposed metaheuristics are extremely satisfactory for both types of the problem.

More specifically, regarding the Problem I, the time length is 780 min and the slack time is less than 1 min that is 0.16 min on average, very close to 0. The POIs are perfectly chosen with almost no idle time for the proposed tour. The total tour duration that our approach suggests is 779.83 min including visit durations at selected sites.

Regarding the Problem II, in which the objective function is the maximization of the number of places visited, yields 8.57 as the objective function value. The total tour duration is 736.29 on average. In comparison with the Problem I, the slack time is expectedly higher since the objective function is the only maximization of the nodes, which ignores the elapsed time, in the Problem II.

However, the deviation results for the both Problems are very small. In other words, the proposed metaheuristics generates outputs, which are very close to each other for the replication number of 100. The suggested routes that the Problem I and Problem II generate are given on the map in Figure 4, respectively.

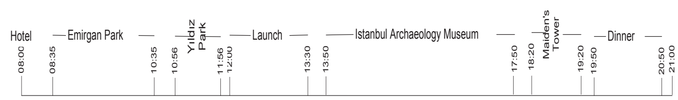
Figure 4. The Proposed Tours and Routes of the Problem I and Problem II



(a) Problem I (100) (a) Problem II (100)

The suggested tour of the Problem I is given in Figure 5.

Figure 5. The Proposed Tour of the Problem I



6 Conclusion and future research

In this effort, we propose a metaheuristics for the solution of the traveling tourist problem with time window and employ for a real-world data; a single great day in Istanbul. The problem includes two time windows of morning and afternoon in a single day. The first covers from the departure of the hotel in the morning to the beginning of the lunch whereas the second covers from the end of the lunch to the arrival to the hotel in the evening. Our two models, the first is with the objective function of the minimization of the total slack times during the tour when the second is with that of the maximization of the number of the visited sites, investigate the best visit points and routes within those time windows. The outputs of the models are extremely good and reasonably applicable.

As a future work, our basic heuristics approach may easily be extended for a multi-day scheduling. In this case, the solution difficulty coming from the multi period nature of the extended problem may be handled with adding local search algorithm to our developed metaheuristics.

Acknowledgment

This paper was partially supported by the Istanbul University. Research Foundation Project (BAP) No: 49254.

7 References

[1] W. Souffriau, P. Vansteenwegen, J. Vertommen, G. V. Berghe and D. V. Oudheusden. “A Personalized Tourist Trip

Design Algorithm for Mobile Tourist Guides," Applied Artificial Intelligence, Vol. 22, No. 10, pp. 964-985, 2008.

[2] Pieter Vansteenwegen, Wouter Souffriau, Greet Vanden Berghe, Dirk Van Oudheusden. "Metaheuristics for Tourist Trip Planning", Metaheuristics in the Service Industry Lecture Notes in Economics and Mathematical Systems, Springer Berlin Heidelberg, Volume 624, pp 15-31, 2009.

[3] Ander Garcia, Olatz Arbelaitz, Maria Teresa Linaza, Pieter Vansteenwegen, Wouter Souffriau. "Personalized Tourist Route Generation", 10th International Conference on Web Engineering ICWE 2010 Workshops, Vienna, Austria, July 2010.

[4] T. Hasuike, H. Katagiri, H. Tsubaki and H. Tsuda. "Tour Planning for Sightseeing with Time-Dependent Satisfactions of Activities and Traveling Times," American Journal of Operations Research, Vol. 3 No. 3, pp. 369-379, 2013.

[5] Li, J., & Fu, P. "A label correcting algorithm for dynamic tourist trip planning". Journal of Software, 7(12), 2899-2905, 2012.

[6] Sylejmani, K., Dika, A. Solving touristic trip planning problem by using taboo search approach. International Journal of Computer Science Issues (IJCSI), 8(5), 2011.

[7] Rodríguez, B., Molina, J., Pérez, F., Caballero, R. "Interactive design of personalized tourism routes. Tourism Management", 33(4), 926-940, 2012.

[8] Sylejmani, K., Dorn, J., & Musliu, N. "A Tabu Search approach for Multi Constrained Team Orienteering Problem and its application in touristic trip planning". In HIS (pp. 300-305), 2012.

[9] Google Maps, https://maps.google.com/, 25.12.2014.

Analysis of the allocation of Surge Arresters in Distribution Lines against Induced Voltages by Indirect Lightning

M. A. Araújo, R. A. Flauzino, O. E. Batista, L. A. Moraes {IEEE member}
Department of Electrical and Computer Engineering
University of São Paulo (USP)
São Carlos, Brazil.
marcel.araujo@usp.br, raflauzino@sc.usp.br, oureste.batista@usp.br, lucas.moraes@usp.br

Abstract— *Induced voltages by indirect lightning cause faults, damage to equipment and malfunctions to electric power systems, compromising the quality, continuity and reliability of the electrical energy supply. In this context, research development is crucial to improve lightning protection procedures, especially for electrical systems located in areas with high ground flash densities. This article presents the results of the application of a practical procedure for protection of the IEEE 13 Node Test Feeder against induced voltages by indirect lightning simulated with the Rusck model employing metal-oxide surge arresters and fuzzy systems for location of points with greater probability of occurrence of flashovers. The results demonstrate the efficiency and practicality of the presented procedure regarding the technical specification and definition of the best allocation and amount of surge arresters for feeder protection.*

Index Terms— fuzzy systems, indirect lightning, induced voltages, lightning protection, surge arrester.

1 Introduction

The incisive regulation, the increasing use of electronic devices and the deployment of new technologies associated with smart grid (SG) and distributed generation (DG) have promoted efforts to improve the levels of quality, continuity and reliability of the supply of electrical energy [1]-[2]. However, transmission lines (TLs) and distribution lines (DLs) are often located in areas with high densities of lightning, being, therefore, subject to unscheduled interruptions caused by temporary overvoltages, compromising these levels [3]-[4].

Lightning is a random phenomenon, which is difficult to be analyzed. This phenomenon causes several electromagnetic disturbances in electric power systems (EPSs). Due to the heating process and climate change through which the planet passes, the occurrence of lightning is intensifying and becoming increasingly severe [5]-[6].

The direct and indirect incidence of lightning originate temporary overvoltages and induced voltages of high amplitude in TLs and DLs. The lightning overvoltages are a major cause of momentary and sustained interruptions on the distribution networks, and in Brazil it is estimated that about one-third of the shutdowns in these networks are caused by lightning [7]-[8].

Moreover, according to related literature, the EPSs may be protected against lightning overvoltages by shield wire, surge arresters, insulators and grounding systems [9]-[10]. However, the application of metal-oxide surge arresters is one of the best alternatives under technical and economical aspects to improve the system's performance under lightning conditions [11]-[13].

In this context, this paper exposes a practical procedure based on the installation of surge arresters for protection against induced voltages by indirect lightning. To do so, using the toolbox SimPowerSystems™, of the Simulink® graphical environment of the MATLAB software, simulations of the induced voltages over IEEE 13 Node Test Feeder and the protection promoted by the allocation of metal–oxide surge arresters at different points and quantities along the feeder will be evaluated. Also be used a fuzzy system (FS) to locate automatically and quickly the points most sensitive and susceptible to flashovers of the network, and in set with the analysis of the accounting of the number of active surge arresters facing induced voltage define the best location and amount of the same to be installed.

Having said that, then, the section II presents the main aspects related to induced voltages by indirect lightning. Section III exposes details about the computational implementation of the feeder, the induced voltages and the metal-oxide surge arresters. In section IV are shown particularities of the FS implemented. The following are exhibit the results of the computer simulations, of the application of FS, and of the analysis of the acting of surge arresters in section V. Finally the study's findings are presented in the section VI.

This work was supported by the University of São Paulo under Research Grant CAPES, CNPq and FAPESP (process 2011/17610-0).

2 Induced voltages by indirect lightning

Indirect lightning are discharges that fall in regions adjacent to the LTs and DLs and that can cause damage to EPSs through the appearing of induced voltages. These voltages are more common in DLs since these networks are usually naturally protected against direct lightning by buildings, towers, trees and high hurdles around their structures [14]-[15]. Although the magnitude of these voltages is smaller than those caused by direct lightning, they are more frequent and severe to distribution systems with nominal voltage of less than 15 kV. This is because the induced voltages can exceed supportability levels of the DLs causing disruptive discharge in isolation, or flashover, reaching high values around 300 kV, since the basic level of isolation of the DLs is low, ranging between approximately 95 kV and 170 kV [7], [14]-[15].

For the calculation of induced voltages, electromagnetic coupling models were developed, among which stands out the Rusck Model [16], with notorious acceptance, adopted by renowned associations such as CIGRÉ and IEEE [17]-[18]. This model is characterized by being an excellent engineering tool, due to its effectiveness, simplicity, and speed of calculation.

Thus, in the analysis of this article, it will be implemented the Rusck Model [16], which has easy computational implementation by providing an analytical expression for the calculation of the induced voltage as shown in (1).

$$U(x,t) = Z_0 I h \beta \frac{ct-x}{y^2+\beta^2(ct-x)^2}\left(1+\frac{x+\beta^2(ct-x)}{\sqrt{(\beta ct)^2+(1-\beta^2)(x^2+y^2)}}\right) \quad (1)$$

Where $U(x,t)$ is the induced voltage in the time domain into a conductor of a line, x is a point along the section of the line, t is the time interval, I is the peak value of the discharge current, h is the height of the line in relation to the ground and y is the distance between the point of incidence of the lightning and the line. Z_0 is the characteristic impedance of the line defined by $Z_0=(1/4\pi)\sqrt{\mu_0/\varepsilon_0}$, where μ_0 is the permeability of vacuum and ε_0 is the electric permittivity of vacuum. β is the ratio between the discharge wave propagation speed (v) and the speed of light in vacuum (c), i.e., $\beta = v/c$.

3 Computational implementations

In order to perform the analysis required by this article, below it will be implemented and simulated computationally IEEE 13 Node Test Feeder, a subsystem of estimation to induced voltages by indirect lightning by Rusck Model and the metal–oxide surge arresters with technical specifications most appropriate for protection of the evaluated system.

3.1 IEEE 13 Node Test Feeder

This feeder was selected due to the varied characteristics of your loads and transformers, and their data are open access, which allows the realization of other related research for purposes of comparison and evaluation.

The implementation of this feeder distribution was carried out on the basis in the data presented in [19], the voltage regulator between the nodes 650 and 632 was suppressed in modeling so that improvements in the voltage profile for its actuation during the simulations do not occur, and it was admitted the switch between the nodes 671 and 692 always closed. In order to validate it was executed the simulation of this system in steady state and through the voltage and current data obtained were created fuzzy rules from which it will be possible notice the influence of the induced voltages in each one of the nodes of the feeder, and then examine the performance of the surge arresters.

To facilitate the reader's understanding in the course of the paper the substation transformer will always be identified by the abbreviations TRF-S, the in-line transformer for TRF-L, loads by L and the number of its respective node, and the distributed load distributed between the nodes 632 and 671 by L-632/671. In addition, following the specifications contained in [19], in this article will always be taken the base voltage of 4.16 kV for the elements of the feeder, with the exception of the load L-634, located after of the in-line transformer, whose base voltage is 480 V.

3.2 Subsystem for estimation of induced voltages

To simulate the action of the induced voltages, it was implemented a subsystem for estimation of induced voltages (SEIV) switched based on (1). For that, voltages were estimated always in relation to the height of the section of the feeder where the SEIV is located and the center of the line, so x=0, and for the period of time corresponding to two cycles at 60 Hz frequency, i.e., t=0.034s. For I was adopted the value of 10 kA, defined from [20], v was taken equal to 120 m/μs from the references [17] and [21], c is 299 792 458 m/s, μ_0 is $4\pi \times 10^{-7}$ Tm/A and ε_0 is 8.85×10^{-12} F/m [22], and thereby, β is equal to 0.4 and Z_0 is equal to 30 Ω.

For the distance between the line and the point of incidence of the lightning (y), 3 groups with distinct values were formed: Group-1: y=50m; Group-2: y=150m; Group–3: y=250m. It wasn't assumed y greater than 250 m due to the reduced length of the excerpts from the line and less than 50 m due to the occurrence of direct lightning for distances lower than this, by virtue of radius of attraction of the elements of the distribution system.

Thus, making use of the SEIV with Rusck Model and the parameters defined above, the induced voltages presented in Fig. 1 were simulated, in which it can be seen the peak amplitude of the same for each of the 3 groups defined.

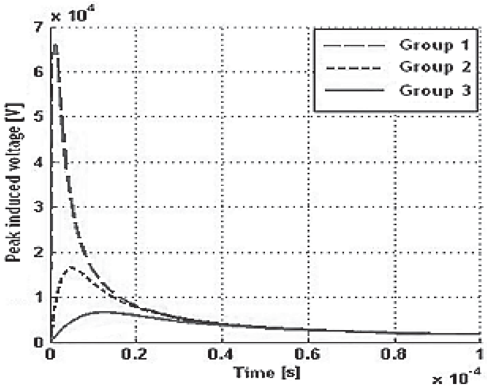


Figure 1: Induced voltages for 3 cases to be simulated.

3.3 Selection and implementation of the metal–oxide surge arresters

After a research about the commercially accessible metal-oxide surge arresters, it was decided to use the silicone housed, medium voltage, type 3EK4, surge arresters from Siemens. This choice occurred due to constructive robustness, mechanical strength, optimization between performance and weight, and suppression of partial discharges and moisture that ingress in its structure, ensuring long service life to the same [23].

The selection of the electrical characteristics of the surge arresters to be implemented was substantiated on exposed in [24]. In this way, knowing that the nominal voltage of feeder test is 4.16 kV and that it has solidly grounded neutral system, its maximum phase voltage (U_s) is 3.39 kV and continuous operation voltage (U_c), calculated from (2), must be $U_c \geq 2.09$ kV.

$$U_c \geq \frac{1{,}05\, U_S}{3} \quad (2)$$

Following it is calculated by means of (3) and (4) the first nominal voltage of the surge arrester (U_{r1}), whose value must be $U_{r1} \geq 2.61$ kV, and the second (U_{r2}), which must be $U_{r2} \geq 2.92$ kV. Thus, it is necessary to determine the final voltage of the surge arrester taking U_r as the greatest value between U_{r1} and U_{r2}, i.e., $U_r \geq 2.92$ kV, and then it is reset U_c by means of equation (5), from which gets $U_c \geq 2.34$ kV.

$$U_{r1} \geq 1{,}25 \times \frac{1{,}05\, U_S}{3} \quad (3)$$

$$U_{r2} \geq 1{,}4\, U_S \quad (4)$$

$$U_c \geq \frac{U_{r2}}{1{,}25} \quad (5)$$

Resuming [24], analyzing the Table 1 and remembering that it is recommended the selection of the surge arrester from a category slightly higher than the minimum indicated by calculated data, it should be assumed $U_r = 6$ kV and $U_c = 4.8$ kV and defined the 3EK4 060-1CB3 model as the surge arrester to be used. Table 1 also shows the maximum values of the residual voltages ($U_{max, residual}$) for different forms of waves and discharge currents, lightning impulse withstanding voltage of 1.2/50 µs ($U_{1,2/50\mu s}$) and power frequency withstanding voltage 1 minute, wet ($U_{freq,industrial}$).

Table 1: Models and electrical characteristics of the surge arrester type 3EK4, adapted from [24].

U_r [kV]	U_c [kV]	Model	$U_{max,\ residual}$ [kV] 8/20µs 10 kA	8/20µs 20 kA	30/60µs 500 A	Housing insulation $U_{1.2/50\mu s}$ [kV]	$U_{freq,\ industrial}$ [kV]
3	2.4	3EK4 030-1CB3	8.0	9.1	6.1	57	24
6	4.8	3EK4 060-1CB3	15.9	18.3	12,2	57	24

Using the model of the element Surge Arrester of the toolbox SimPowerSystems™ and data obtained above, it was simulated the behavior of the surge arresters facing lightning with waveform of 1.2/50 µs and lightning current of 10 kA, reaching the result of the Fig. 2. Analyzing this figure it was verified that the model of surge arresters implemented is operating correctly, because from the moment in which they are subjected to a voltage higher than their nominal voltage, they passed to act with low impedance dissipating the atmospheric impulse current, limiting the value of the resulting overvoltage and recovering its high impedance value after the end of the solicitation of voltage.

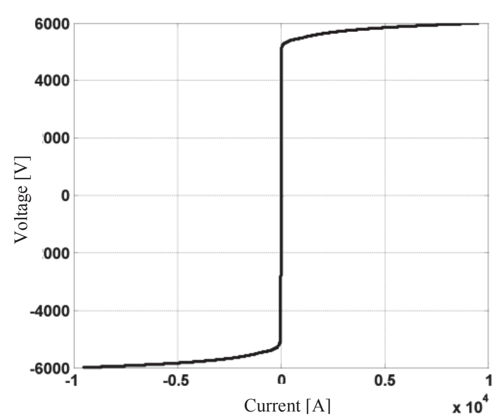


Figure 2: Performance of the model of surge arresters selected to a waveform 1.2/50 µs and a lightning current of 10 kA.

4 Fuzzy system

In order to map the relationship between the voltages and currents measured before and after of the emergence of the induced voltages by indirect lightning was used a multilayer fuzzy system as shown in Fig. 3, with automatic adjustment structural and parametric presented in [25] and [26]. With the application this intelligent tool it is intended to localize automatically and quickly the most sensitive points, and therefore susceptible to flashovers in the network, for more efficient protection of loads and transformers.

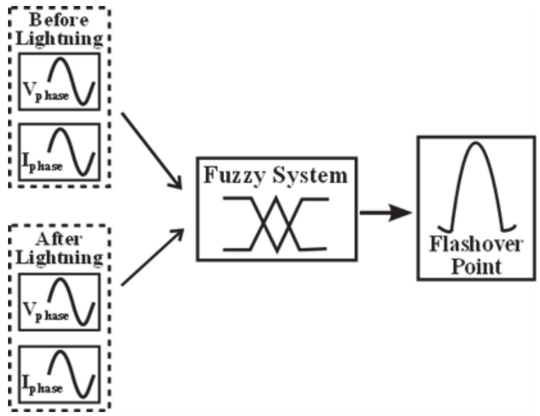


Figure 3: Fuzzy system for locating the point of occurrence of flashover.

The multilayer fuzzy system applied in this paper is compound of two entries, and each entry has three gaussian relevance functions associated. Already the output variable has two gaussian relevance functions associated and a rule base composed of five rules. Assuming that each gaussian function has two free parameters of tuning, the mean and the variance, then the input layer will have 12 free parameters of tuning, in the same way that the output layer will have four tuning parameters.

The tuning of the fuzzy inference system implemented consists of two distinct steps of operation. In the first step, the rule base is generated so as to minimize the cost function associated with the system. In the second step, the fine tuning of the system is performed through the adaptation of the free parameters of system, i.e., at this step the parameters of the relevance functions of the inputs and outputs, as well as the weighting of each rule is amended to optimize the FS.

In addition, it is used as a method of aggregation of the inference rules the operator "maximum" and to defuzzification method "center area", which will produce the only output of the system. Fig. 4 shows a schematic diagram of the multilayer fuzzy model for a system composed of two inputs and three inference rules activated.

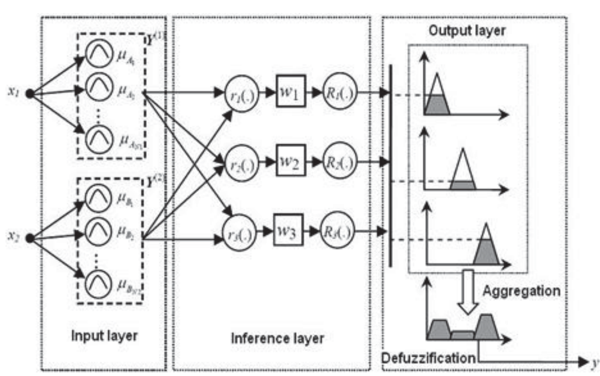


Figure 4: Multilayers fuzzy system.

Thus, the FS implemented can provide qualitative information on the inter-relationship between the voltages and currents before and after the occurrence of indirect lightning, helping to identify the points of greatest sensitivity flashovers event.

5 Results

From the analysis of the length of the sections of the feeder and of the existence or not of connection of loads to its nodes, the location of SEIVs and surge arresters (SAs) were defined. The SEIVs were allocated in the center of the sections along the branches of the feeder to cover the greatest length of line by the action of its induced voltages, as illustrated by the Fig. 5, accounting a total of 7 subsystems. On the other hand, the SAs were allocated in the structures of the terminations of the sections of line, as shown in Fig. 5, in order to protect the loads and the transformers. It may be noted that in Fig. 5 the abbreviations SA symbolizes the surge arresters connected in all phases of determined node of the feeder.

Moreover, by means of combining the switching of the 7 SEIVs 128 simulations were held for each of the 3 groups defined in section III-B of this paper, and voltage and current meters were allocated to all nodes of the feeder. Thus, analyzing the data of 384 simulations with DL operating without any surge arresters allocated, it is obtained the probability of occurrence of flashover in each node of the feeder through the use of the FS implemented, as shown in Fig. 6.

By means of Fig. 6, it is noted that in the simulations of the Group-1 higher overvoltage peaks were observed, and therefore are higher the probability of occurrence of flashovers on transformers and loads compared to simulations of Groups 2 and 3, since the smaller the distance between the line and the point of impact of the lightning, larger are induced voltages.

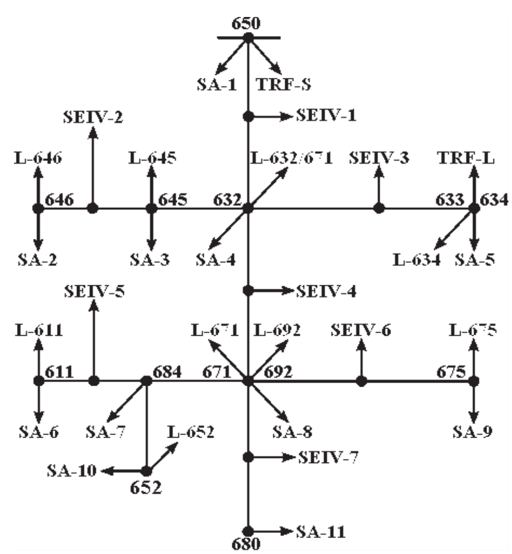


Figure 5: Allocation points of the surge arresters (SAs), subsystems estimation induced voltages (SEIVs), loads (Ls) and transformers (TRFs) in the test feeder.

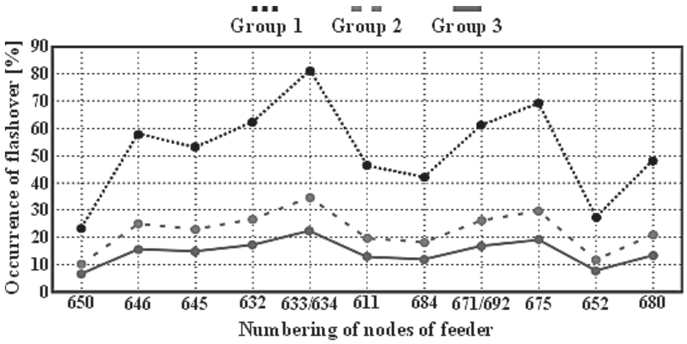


Figure 6: Probability of occurrence of flashover in each one nodes of the test feeder.

Furthermore, analyzing Fig. 6 and making the correspondence between the node number and the surge arrester connected to it, it is possible to define the most relevant surge arrester for continuous operation test feeder, since the higher the probability of occurrence of flashover in a node, the higher the probability of voltages exceed the supportability of the devices connected to it. Thus, following this criterion the most important surge arresters for protection of this feeder in descending order according to the first ordering are: SA-5, SA-9, SA-4, SA-8, SA-2, SA-3, SA-11, SA-6, SA-7, SA-10 and SA-1.

For the purpose of evaluating the performance of the proposed method, a set of test samples was applied to the FS. The estimation showed average error of 0.0049%, standard deviation of 0.0122% and maximum error of 0.0387%. Fig. 7 shows the results of this test, in which each level represents a point of event flashover, and where it is possible to note the precision of the method developed. Already Fig. 8 contains the histogram of errors, from which it can be seen that the vast majority of the errors occur when its percentage value is very low, again showing the accuracy of the procedure implemented.

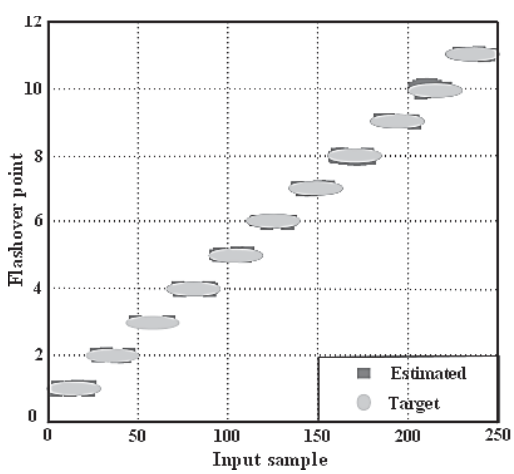


Figure 7: Test of the locating the point of flashover happening.

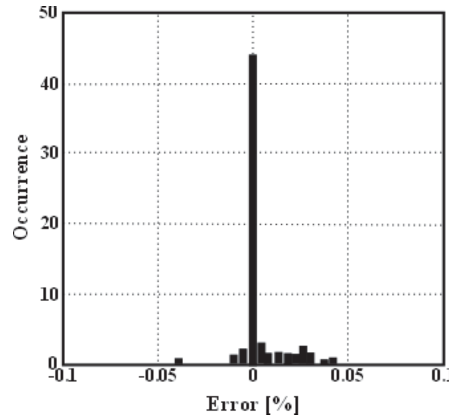


Figure 8: Histogram of the error of point location of the event flashovers.

The following, 384 new simulations were performed with the surge arresters in operation in order to quantify the number of performances by phase for each one of the nodes in that were allocated, as exposed in Fig. 9. Considering the percentage number of operations of the surge arresters it is possible to define the second descending ordering of importance of the surge arresters for protection of the feeder as follows: SA-9, SA-8, SA-11, SA-7, SA-4, SA-5, SA-3, SA-6, SA-2, SA-10 and SA-1.

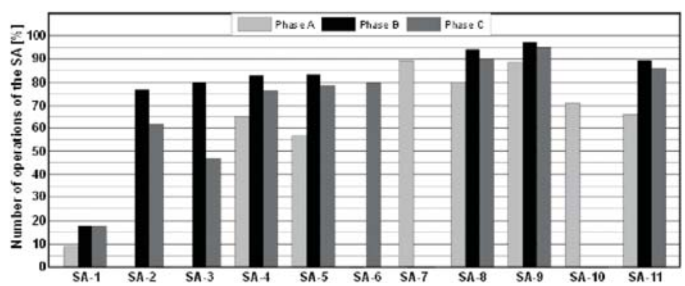


Figure 9: Percentage number of operations of the surge arresters in relation to 384 simulations performed.

Considering the technical/economic viability and the protection provided by the installation of the surge arresters, assigning weights of 1 to 11 according the position of the surge arresters in both previous ordering and adding them, it is possible to obtain a third ordering. In this new ordering, shown in Table 2, the surge arresters with less sum of the weights are the most important, since the lower the weights in the first two ordering the greater the importance them for system protection. Thus, so practice it is determined the best ordering of installation of the surge arresters in the test feeder for more effective protection against induced voltages by indirect lightning.

Table 2: Ordering of the surge arresters according their installation priority to test feeder protection.

First ordination	Weight 1	Second ordination	Weight 1	Normal ordination	Total weight	Third ordination
SA - 5	1	SA - 9	1	SA - 1	22	SA - 9
SA - 9	2	SA - 8	2	SA - 2	14	SA - 8
SA - 4	3	SA - 11	3	SA - 3	13	SA - 5
SA - 8	4	SA - 7	4	SA - 4	8	SA - 4
SA - 2	5	SA - 4	5	SA - 5	7	SA - 11
SA - 3	6	SA - 5	6	SA - 6	16	SA - 7
SA - 11	7	SA - 3	7	SA - 7	11	SA - 3
SA - 6	8	SA - 6	8	SA - 8	6	SA - 2
SA - 7	9	SA - 2	9	SA - 9	3	SA - 6
SA - 10	10	SA - 10	10	SA - 10	20	SA - 10
SA - 1	11	SA - 1	11	SA - 11	10	SA - 1

6 Conclusions

With the results obtained it is concluded that the implemented FS for to identify the point of occurrence of flashovers was efficient and robust, as exposed by the responses to the test performed and found by the measured errors. Furthermore, it is observed that the FS allows greater agility for analysis of lightning overvoltages under the test feeder analyzed since it determines the percentage of occurrence of flashover on each node of feeder, contributing to determination of the most critical point that must be protected in the DL.

Associating the ordinations derived from the possibility of occurrence of flashovers and the number of operations of the surge arresters by connection phase, it was possible to establish a third ordinations of necessity of installation of the surge arresters, which overlooks the needs of utilities with respect to technical/economic viability and to the customer service of the reliability, compliance and quality requirements of energy can be applied in defining the surge arresters to be installed by phase along the feeder.

In this context, it was noted the efficiency and practicality of the proposed procedure for protection of the IEEE 13 Node Test Feeder against induced voltages by indirect lightning, glimpsing not only protect EPSs and its components, but also covered the consumers serviced by them.

7 References

[1] H. B. Putten, P. R. MacGregor and F. C. Lamber, “Distributed generation: Semantic hype or the dawn of a new era?”, IEEE Power and Energy Magazine, vol. 1, nº 1, pp. 22 – 29, 2003.

[2] M. H. Bollen and F. Hassan, Integration of Distributed Generation in the Power System. New York: Wiley-IEEE Press, 2011.

[3] “IEEE Guide for Improving the Lightning Performance of Electric Power Overhead Distribution Lines”, IEEE Std 1410-2010 (Revision of IEEE Std 1410-2004), pages 1-73, 2011.

[4] M. Marzinotto, F. Fiamingo, C. Mazzetti and G.B. Lo Piparo, Quality of the supplied electric service: a tool to evaluate the need of protection against lightning surges, Electric Power Systems Research, Vol. 85, pages 75-81, 2012.

[5] A. Piantini. Lightning Protection. Journal of Lightning Research, 4, (Suppl 1: M1), pp. 1-2, 2012.

[6] O. Pinto. Lightning in the Tropics: From a Source of Fire to a Monitoring System of Climatic Changes. Nova Science Publishers, New York, 2009.

[7] A. Piantini and J. M. Janiszewski. “Lightning-Induced Voltages on Overhead Lines—Application of the Extended Rusck Model”. IEEE Transactions on Electromagnetic Compatibility, vol. 51, nº 3, pp. 548 - 558, 2009.

[8] O. Pinto, Pinto, I. R. A. Storms and lightning in Brazil (in Portuguese). São José dos Campos, Ed. Independente, 194 p., 2000.

[9] P. M. Anderson. Power System Protection. IEEE Press power engineering series. McGraw-Hill, pages 1307, 1999.

[10] Y. G. Paithankar and S. R. Bhide. Fundamentals of Power System Protection. PHI Learning; 2nd edition, pages 384, 2010.

[11] IEEE, Power & Energy Society. IEEE Guide for the Application of Metal-Oxide Surge Arresters for Alternating Current-Systems. Surge Protective Devices Committee, IEEE Standards Board, American National Standards Institute, New York, 2009.

[12] K. Munukutla, V. Vittal, G. T. Heydt, D. Chipman and B. Keel. “Practical Evaluation of Surge Arrester Placement for Transmission Line Lightning Protection”. IEEE Transaction on Power Delivery, Vol. 25, nº 3, July, 2010.

[13] M. A. Araújo, R. A. Flauzino, R. A. C. Altafim, O. E. Batista and L. A. Moraes. “Practical methodology for modeling and simulation of a lightning protection system using metal-oxide surge arresters for distribution lines”. Electric Power Systems Research, v. 118, pp. 47-54, 2015.

[14] A. J. Eriksson; M. F. Stringfellow and D. V. Meal. Lightning–Induced Overvoltages on Overhead Distribution Lines. IEEE Transactions on Power Apparatus and Systems, Vol. PAS-101, nº 4, p. 960-968, April, 1982.

[15] A. Borghetti; C. A. Nucci; M. Palone. An Improved Procedure for the Assessment of Overhead Line Indirect Lightning Performance and Its Comparison with the IEEE Std. 1410 Method. IEEE Transactions on Power Delivery, Vol. 22, nº 1, January, 2007.

[16] S. Rusck. Induced Lightning Overvoltages on Power Transmission Lines With Special Reference to the Overvoltage Protection of Low Voltage Networks. Transactions of the Royal Institute of Technology, Stockholm, Sweden, nº 120, 1958.

[17] CIGRÉ, Working Group 33.01 (Lightning) of Study Committee 33 (Overvoltages and Insulation Coordination). Guide to Procedures for Estimating the Lightning Performance of Transmission Lines. CIGRÉ Brochure 63, Paris, October, 1991.

[18] IEEE, Working Group on the Lightning Performance of Distribution Lines. IEEE Guide for Improving the Lightning Performance of Electric Power Overhead Distribution Lines. IEEE Std 1410, 2004.

[19] W. H. Kersting. Radial distribution test feeders, IEEE Power Engineering Society Winter Meeting, IEEE Conference Proceedings, Vol. 2, pp. 908–912, 2001.

[20] W. Diesendorf, Insulation Coordination in High Voltage Electric Power Systems. London, Butterworth, 1974.

[21] IEEE Working Group on Estimating Lightning Performance of Transmission lines, "IEEE guide for improving the lightning performance of transmission lines," Proposed IEEE P1243 draft, 1996.

[22] D. Halliday; R. Resnick; J. Walker. Fundamentals of physics 1 – Electromagnetism (in Portuguese). Editora: Ltc, 9ª Edição, 2012.

[23] Siemens, Guide to surge arresters. Surge arresters of medium voltage 3EK4 type with silicone casing. Available at: http://www.energy.siemens.com/br/pt/transmissao-de-energia/produtos-de-alta-tensao/descarregadores-de-sobretensao-limitadores/tensao-alta-e-media/, Accessed at 2/28/2015.

[24] V. Hinrichsen. Metal-oxide Surge Arrester in High-Voltage Power Systems, Fundamentals. 1 st Edition, Siemens AG, Berlin, Germany, 2012.

[25] R. A. Flauzino and I. N. Silva. "Tuning of Fuzzy Inference Systems Through Unconstrained Optimization Techniques," International Journal of Computer Research, vol. 12, pp. 243–252, 2003.

[26] R. A. Flauzino and I. N. Silva. "Efficient parametric adjustment of fuzzy inference system using unconstrained optimization," Lecture Notes in Computer Science, vol. 4507, pp. 399–406, 2007.

A Comparative Study of Metaheuristics Techniques for Portfolio Selection Problem

Adebiyi Ayodele A. and Ayo Charles K.
Department of Computer and Information Sciences, Covenant University, PMB1023, Ota, Nigeria.
ayo.adebiyi@covenantuniversity.edu.ng, charles.ayo@covenantuniversity.edu.ng

Abstract - *Portfolio selection problem (PSP) is one of the major interesting research areas in finance which have drawn interest of several researchers over the years. Over time, the different approaches had been engaged in solving the PSP ranging from computational techniques to metaheuristics techniques with varying results. In this paper, we engaged three different metaheuristics techniques under this same condition to solve extended Markowitz mean-variance portfolio selection model. The three metaheuristics techniques are Non-dominated Sorting Genetic Algorithm II (NSGAII), Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO) and Generalized Differential Evolution 3 (GDE3). A comparative analysis was carried out with results obtained with existing benchmark data available in literature. The outcome of the findings reveals that SMPSO shows superior performance, followed by NSGAII in many different instances; however, the mean execution time of GDE3 was the fastest among the three techniques considered.*

Keyword: Portfolio Selection, Speed-constrained Multi-objective Particle Swarm Optimization, Non-dominated Sorting Genetic Algorithm, Generalized Differential Evolution 3, Metaheuristic.

1 INTRODUCTION

There are quite a number of methods researcher had engaged to tackle the portfolio selection problem with one short coming or the other. The following are the prevalent methods that have been applied to PSP in literature. Fuzzy set theory had been immensely engaged in portfolio selection; among the few works reported in literature are the works of [1, 2, 3, 4, 5, 6]. Genetic Algorithm (GA) has also been extensively used to solve PSP as reported in work of [7, 8, 9, 10, 11]. In the work of [12] engaged a heuristic technique of particle swarm optimization (PSO) to extend Markwitz mean variance portfolio selection problem. Their findings compared with GA revealed a superior performance over GA model. Also, in a similar work by [13], developed PSO model for PSP and compared their results with GA model. Their finding showed that PSO model demonstrated high computational efficiency in building optimal risky portfolios. Others related works that engaged PSO for PSP are [14, 15, 16]. Few related works that engaged GDE for portfolio selection problem are as follows [17, 18].

This paper presents an empirical comparative study of three different metaheuristics techniques to portfolio selection model and relates the findings obtained to what has been reported in extant literature. The finding reveals that SMPSO shows superior performance, followed by NSGAII in many different instances; however, the mean execution time of GDE3 was the fastest among the three techniques considered.

The rest of the paper is organized as follows. Section 2 presents the portfolio selection problem used in this study. The methodology used is explained in section 3. Selection 4 contained the computational results obtained in this work and the paper concluded in section 5.

2 PORTFOLIO SELECTION PROBLEM

This section describes the PSP model used in this work as formulated in the work of [19]. The model is an extension of Markowitz's mean variance portfolio selection model in the work of [20]. To explain the PSP model the definition of following variables are of importance. Therefore:

N	is the number of available assets
M	is the number of assets to be selected from N available assets
B	is the total available budget
R	is the investor's expected rate of return
σ_p^2	is the return variance of the portfolio.
σ_{ij}	is the covariance of returns of asset i and j ;
B_{lower_i}	is the minimum amount of budget that can be invested in asset i
B_{upper_i}	is the maximum amount of budget that can be invested in asset i
c_i	is the minimum transaction lots for asset i

x_i is the number of c_i 's that is purchased

w_i is the decision variable that represents the weight of the budget to be invested in asset i .

w_j is the decision variable that represents the weight of budget to be invested in asset j ;

z_i is a binary variable $\{0,1\}$ if 1 asset i is in the portfolio and otherwise 0

e_i is the expert opinion, a random variable of equal or greater than 0.5 if the asset i is selected and otherwise 0

i is the index of securities

Investors are always desire to minimize risk of investment and maximize possible return. The extended Markowitz model for the portfolio selection problem used in this study is as formulated as follows:

$$\min \quad \sigma_p^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \sigma_{ij} \tag{1}$$

where

$$w_j = \frac{x_i c_i z_i}{\sum_{i=1}^{N} x_i c_i z_i}, j = 1,..., N \tag{2}$$

and

$$\sum_{i=1}^{N} z_i = M \leq N; \quad M, N \in \mathrm{N} \tag{3}$$

subject to

$$\sum_{i=1}^{N} x_i c_i z_i e_i r_i \geq BR \tag{4}$$

$$\sum_{i=1}^{N} x_i c_i z_i e_i \leq B \tag{5}$$

$$0 \leq B_{lower_i} \leq x_i c_i \leq B_{upper_i} \leq B, i = 1,..., N \tag{6}$$

$$\sum_{i=1}^{N} w_i = 1 \tag{7}$$

$$w_i \geq 0, \quad \forall_i \in \{1,2,..., N\} \tag{8}$$

$$e_i \in \{0,1\} \tag{9}$$

where

$$z_i = \begin{cases} 1 & if \quad e_i \geq 0.5 \\ 0 & if \quad e_i < 0.5 \end{cases} \tag{10}$$

$x_i c_1$ represents the number of units of asset i in the selected portfolio. z_i is the decision variable in which it is equal to 1 if the asset i is upheld in the portfolio and otherwise 0. The inequality in equation (3) denotes cardinality constraint. Equation (5) represents the budget constraint. Equation (6) indicates the bounds on holdings constraint. The equations (7) and (8) ensure that the total budgets are invested in the portfolio. The equations (9) and (10) represent the expert opinion constraint. The expert opinion constraint is a practicable and useful constraint in a real life scenario of portfolio selection because the expert has detailed information about sector capitalization where each asset i to be selected in the portfolio belong in order to minimize investment risk. Beyond sector capitalization the expert or financial analyst can access other information regarding each asset i to be selected in the portfolio such as price/annual earning, management calibre, dividend rate, book value and so on. An in-depth analysis of these information can guide the expert upon which an opinion is formed whether asset i should be included in the portfolio or not. This paper considered four different set of important constraints to the portfolio selection problem. This extended Markowitz model was solved with three efficient Metaheuristics to find the optimum solution and compared the results with one another.

3 METHODOLOGY

This section describes data set used and experimental details. The extended Markowitz model used in this work was implemented with efficient Metaheuristics method of NSGAII, SMPSO, and GDE3 with each set of data of 31 and 85 stocks from the stock markets of Hong Kong Hang Seng and the German DAX 100 respectively. The data were obtained from test data from OR-Library [21]. Each data set contains the number of assets (N). The mean return and standard deviation of return for each asset i and correlation between asset i and j for all possible pairs of assets. In order to evaluate the performance of the algorithm on the portfolio model used. It was run on a PC with Intel Pentium 4.3 GHz with 2GB RAM. The parameter settings for each of the data set is as follows: expert opinion was set to greater than 0.5 if the asset is selected in the portfolio, the value of the budget was set to 2800, expected rate of returns was set to 0.004, 0.005, and 0.006 respectively. A predetermined upper and lower bound was set for each of the selected assets. The size of portfolio was set to 15, 20, 25 for each of the data set.

Five criteria were used to compare the performance of the results obtained by the NSGAII, SMPSO and GDE3 algorithms used for the portfolio model. The criteria are as follows:

- Best variance; depict lowest risk from algorithm runs, showing the best solution found.
- Mean variance; the average of the objective function found by the algorithm.

- Worst variance, depicts the highest risk from algorithm runs, showing the worst solution.
- Standard deviation of variance, depicts how close the solution found by the algorithms are close to each other and,
- Mean execution time, depicts the amount of time needed to arrive to a solution.

4 COMPUTATIONAL RESULT AND DISCUSSION

The results of GDE3, NSGAII and SMPSO algorithms for data set of 31 stocks are tabulated in table 1 over 50 independent iterations. Similarly, the results obtained for data set of 85 stocks with GDE3, NSGAII, SMPSO are contained in table 2 accordingly.

The computational experiment as indicated in table 1 when the size of data set is 31 shows that SMPSO have the best results in all the instances when the size of the portfolio is 15, 20, and 25 respectively. This is followed by the results of NSGAII. However, it was observed that computational time of GDE3 is lesser than NSGAII and SMPSO metaheuristics.

Similarly, to further evaluate the performance of the extended Markowitz portfolio model in a complex scenario of larger dataset of 85 stocks. Table 2 shows the results obtained with 85 stock data set and comparison with metaheuristics used depicts similar trend as SMPSO gave superior performance over the other metaheuristics of NSGAII and GDE3. However, GDE3 metaheuristics have less computation time to generate solutions with the portfolio model in comparison to other metaheuristics.

The results in this study corroborate others finding in extant literature that SMPSO metaheuristics provide alternative promising method in solving portfolio selection problem. It can be used as a guide to investors to minimize their risks of investment.

Table 1: Results of applying GDE3, NSGAII and SMPSO algorithms to 31 stocks data set across 50 independent executions

Size of Portfolio		Expected rate of return	0.004			0.005			0.006		
	Variance	Algorithms	GDE3	NSGAII	SMPSO	GDE3	NSGAII	SMPSO	GDE3	NSGAII	SMPSO
15		Best	0.45123666	0.35203899	0.15107619	0.45057284	0.34717306	0.214028489	0.54322836	0.35378268	0.25157078
		Mean	0.83230867	0.67218507	0.61424974	0.79284597	0.64333504	0.566691078	0.71944886	0.68509601	0.60348172
		Worst	1.10275477	0.98431799	0.77090457	1.09839860	0.95442150	0.735944859	0.96206750	1.07633423	0.69708422
		Std. Dev.	0.34063198	0.13738563	0.13578541	0.22195648	0.15035748	0.129776277	0.19958594	0.17494801	0.14831623
		Mean exe. time (s)	22.81358	34.74636	38.32675	24.20438	29.6409	31.0872	32.41838	28.78512	35.6251
20		Best	0.73169554	0.52981593	0.32947806	0.90755939	0.51186707	0.373316492	0.536085601	0.315823093	0.287602147
		Mean	1.46968870	0.93358625	0.55694377	1.49795920	0.91360268	0.800158982	1.472028375	0.908991852	0.705545178
		Worst	2.31710848	1.56117885	0.95614994	2.07168664	1.38676291	1.234916183	2.403294162	1.448040121	0.943375093
		Std. Dev.	0.35321293	0.21277851	0.17553432	0.31429033	0.19925772	0.114052924	0.356558781	0.232814076	0.188732521
		Mean exe. time (s)	37.39024	43.38638	45.93512	28.52338	30.10292	32.5496	36.85026	33.25678	39.98751
25		Best	0.88679723	0.65977877	0.45408498	1.14151308	0.63744981	0.510079919	0.845219905	0.635221067	0.434594241
		Mean	1.65110704	1.08509956	0.68236685	1.81419318	1.10107861	0.946176642	1.703029315	1.058870543	0.721922588
		Worst	2.57558983	2.09389079	0.86684947	2.55817477	1.71801513	1.368720214	2.705876324	1.723429881	1.239178664
		Std. Dev.	0.39776689	0.28499070	0.16394986	0.35767066	0.26451288	0.214568917	0.383476778	0.223461266	0.172342988
		Mean exe. time (s)	25.19022	36.53728	40.18954	21.07522	25.48038	29.50243	25.05926	20.6254	28.9627

Table 2: Results of applying GDE3, NSGAII and SMPSO algorithms to 85 stock data set across 50 independent executions

Size of Portfolio		Expected rate of return	0.004			0.005			0.006		
		Algorithms	GDE3	NSGAII	SMPSO	GDE3	NSGAII	SMPSO	GDE3	NSGAII	SMPSO
15	Variance	Best	0.22518776	0.14043814	0.13438406	0.17278134	0.09654804	0.079737446	0.235832895	0.198249716	0.075848570
		Mean	0.60096916	0.29943517	0.26523937	0.54470193	0.28107119	0.254470528	0.547518040	0.322653323	0.278438959
		Worst	0.96524521	0.55895829	0.48650518	0.94764824	0.43885621	0.394134793	1.023240679	0.598769274	0.356018608
		Std. Dev.	0.16767265	0.08599229	0.07007661	0.18088869	0.07421649	0.067178084	0.168622109	0.083048395	0.064038095
		Mean exe. time (s)	33.55548	39.26028	42.63432	37.81104	45.05456	47.11035	42.08742	44.13692	46.98563
20		Best	0.35715149	0.20805088	0.17834219	0.41142129	0.24077569	0.158127063	0.432725211	0.217592437	0.137872202
		Mean	0.81708831	0.41541530	0.35715149	0.78676979	0.40527862	0.326921371	0.832714277	0.427922810	0.283161073
		Worst	1.23085633	0.70692483	0.59088845	1.35634723	0.70969466	0.527231052	1.351143703	0.743734672	0.487801885
		Std. Dev.	0.19587954	0.10504707	0.09101289	0.19614638	0.11433733	0.083268498	0.230234543	0.124465355	0.074373463
		Mean exe. time (s)	37.95714	42.44526	43.56093	32.60874	38.62946	41.78632	35.80182	37.14916	39.9572
25		Best	0.39090568	0.27617457	0.19016071	0.4885082	0.30835802	0.232138784	0.356562971	0.325238055	0.304761661
		Mean	0.83595081	0.52020104	0.35529302	0.85784000	0.52260675	0.488508212	0.877025262	0.529653841	0.410432784
		Worst	1.37399055	0.78384814	0.68452987	1.13836791	0.95231780	0.720440502	1.336416098	0.789133804	0.633696336
		Std. Dev.	0.21105804	0.11005574	0.08231027	0.17048585	0.13102051	1.028592817	0.211977567	0.113633008	0.10946686
		Mean exe. time (s)	38.01702	44.0845	45.94935	36.69042	41.6266	44.889324	42.45796	50.19838	53.33126

5. CONCLUSION

In this research work, a comparative study of three metaheuristics algorithms are engaged to solve portfolio selection problem. The three metaheuristics techniques are Non-dominated Sorting Genetic Algorithm II (NSGAII), Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO) and Generalized Differential Evolution 3 (GDE3). The outcome of the findings reveals that SMPSO shows superior performance, followed by NSGAII in many different instances; however, the computational time of GDE3 was the fastest among the three techniques considered. The future studies are to engage hybrid of swarm intelligence techniques to solve PSP model for optimum performance.

REFERENCES

[1] Tanaka H., Guo P., Turksen B. “Portfolio selection based on fuzzy probabilities and possibility distributions”, Fuzzy Sets and Systems Vol. 111, pp. 387–397, 2000.

[2] Leon T., Liern V., and Vercher E. “Viability of infeasible portfolio selection problems: a fuzzy approach”, European Journal of Operational Research, Vol. 139 pp.178–189, 2002.

[3] Huang X. “Fuzzy chance-constrained portfolio selection”, Applied Mathematics and Computation Vol. 177 pp. 500–507, 2006.

[4] Gupta P., Mehlawat M.K., and Saxena A. “Asset portfolio optimization using fuzzy mathematical programming”, Information Science Vol. 178 pp.1734–1755, 2008.

[5] Li X., Zhang Y., Wong H.S., and Qin Z. “A hybrid intelligent algorithm for portfolio selection problem with fuzzy returns”, Journal of Computational and Applied Mathematics Vol. 233 pp. 264–278, 2009.

[6] Jun L. and Jiuping X. “Multi-objective portfolio selection model with fuzzy random returns and a compromise approach-based genetic algorithm”, Journal of Information Sciences, Vol.220, pp.507-521, 2013.

[7] Loraschi A., Tettamanzi A., Tomassini M., Svizzero C., Scientifico C., and Verda P. “Distributed genetic algorithms with an application to portfolio selection”, D.W. Pearson, N.C. Steele, R.F. Albrecht (Eds.), Proceedings of the international conference on artificial neural networks and genetic algorithms (ICANNGA95), Springer-Verlag, Berlin, pp. 384–387, 1995.

[8] Lin D., and Wang S. “A genetic algorithm for portfolio selection problems”, Advanced Modeling and Optimization, Vol. 4 No.1, pp. 13–27, 2002.

[9] Oh K.J., Kim T.Y., Min S.-H., and Lee H.Y. “Portfolio algorithm based on portfolio beta using genetic algorithm”, Expert Systems with Applications, Vol. 30, pp. 527–534, 2006.

[10] Zhang W.-G., Chen W., and Wang Y.-L. “The adaptive genetic algorithms for portfolio selection problem”, International Journal of Computer Science and Network Security, Vol. 6, No. 1 , pp. 196–200, 2006.

[11] Lin C.C., and Liu Y.T. “Genetic algorithms for portfolio selection problems with minimum transaction lots”, European Journal of Operational Research, Vol. 185, pp. 393–404, 2008.

[12] Hamid R. G. , Mehrshad F., “Constrained Portfolio Selection using Particle Swarm Optimization”, Expert System with Applications, vol.38, no.7, pp.8327-8335, 2011.

[13] Hanhong Z., Yi W., Kesheng W., Yun C. Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem”, Expert System with Applications, Vol.38, pp.10161-10169, 2011.

[14] Pulido G.T. and Coello Coello C.A. “A constraint-handling mechanism for Particle Swarm Optimization”, IEEE (2004), pp. 1396–1403, 2004.

[15] Xu F., Chen W., Yang L. “Improved Particle Swarm Optimization for realistic portfolio selection”, Eighth ACIS international conference on software engineering, artificial intelligence, networking, and parallel/distributed computing, IEEE Computer Society, pp. 185–190, 2007.

[16] Zheng J., Wu Q., and Song W. (2007) “An improved particle swarm algorithm for solving nonlinear constrained optimization problems”, Third International Conference on Natural Computation (ICNC 2007), IEEE Computer Society.

[17] Ardia D., Boudt K., Carl P., Mullen K.M. and Peterson B.G. “Differential Evolution with DEoptim: An Application to Non-Convex Portfolio Optimization”, The R Journal vol. 3, no.1, pp. 27-34, 2011.

[18] Xiaohua M., Yuelin G., and Bo W. “Portfolio Optimization with Cardinality Constraints Based on Hybrid Differential Evolution”, 2012 AASRI Conference on Computational Intelligence and Bioinformatics, vol.1, pp. 311-317, 2012.

[19] Adebiyi A.A and Ayo C.K. “Portfolio Selection using Generalised Differential Evolution 3”, Journal of Applied Mathematical Sciences, Vol. 9, No. 42, pp. 2069-2082, 2015.

[20] Hanhong Z., Yi W., Kesheng W., and Yun C. “Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem”, Expert System with Applications, vol.38, pp.10161-10169, 2011.

[21] OR-Library https://files.nyu.edu/jeb21/public/jeb/orlib/portinfo.html 2004. [Access online March, 2014]

Text Optimization using Interactive Evolutionary Computation Techniques

Q. Madera, M. Garcia and O. Castillo
Division of Graduate Studies and Research, Tijuana Institute of Technology, Tijuana, Mexico

Abstract - *The description of a product or an ad's text can be rewritten in many ways if other text fragments which are similar in meaning substitute different words or phrases. A good selection of words or phrases, composing an ad, is very important for the creation of a good advertisement text, as the meaning of the text depends on this and it affects in a positive or a negative way the interest of the possible consumers towards the advertised product. In this paper we present a method for the optimization of advertisement texts through the use of interactive evolutionary computing techniques and fuzzy logic. The use of Fuzzy Logic enables to manage the uncertainty in the process of evolution. The EvoSpace platform is used to perform the evolution of a text, resulting in an optimized text, which should have a better impact on its readers in terms of persuasion.*

Keywords: Evolutionary Computing, Fuzzy Logic, Text Optimization

1 Introduction

Text content is plays a very important role in e-commerce applications, as this is one of the most common ways of giving information about a commercial product to the consumers [12]. When the author of an advertisement text is an expert ad writer, the product should have a better chance of receiving a positive response from the consumers. The combination of words or phrases (blocks of text) that the experts decide to use when writing the text of an ad is important, because this particular combination could be the one that persuades the consumer into buying the product. If an inexperienced writer decides to write an advertisement text, it would be very difficult for him to choose a correct combination of the blocks of text that is successful into persuading the majority of the consumers. In this work, we propose that a writer of any level of experience can create an ad with different interchangeable blocks of text carrying the same meaning, and a third party could optimize it.

Evolutionary algorithms are commonly used to solve optimization problems [9] and that's why we decided to use this kind of techniques to optimize the advertisement texts. We believe that if a group of people can evaluate, in terms of persuasion, different combinations of the same ad, after many generations of evolution we can find an optimal ad, which will have a better impact on the majority of the consumers.

2 Basic Concepts

In this section we present some basic concepts for understanding this work, and the essential parts that compose the proposed method. One of the most important components is EvoSpace, which turned to be a crucial tool for the implementation of the genetic algorithm used in this work [5]. This tool was developed by our research group [6] to perform experiments of interactive evolutionary approaches and for that reason was used to test the proposed method in this paper.

2.1 Evolutionary algorithms

Evolutionary algorithms are a subfield of artificial intelligence. They are mainly used in optimization problems where the search space is very large and aren't linear. These algorithms search for solutions based on the theory of the Darwinian evolution.

The methods of this kind generate a set of individuals that represent possible solutions. These solutions are usually generated randomly at the beginning of the evolution process. After each generation, the best solutions share part of their information to create other possible, better, solutions. All of the individuals compete to be the more fit solutions; the better solutions are conserved, while the worse are destroyed, according to a fitness function that evaluates their performance [1].

2.2 Genetic algorithms

Genetic algorithms (see Fig.1) are inspired in biological evolution; they evolve a population of individuals by performing genetic recombination and mutation. A selection of the best solutions is made by the use of certain criterion and a fitness function, and based on their performances, the more fit individuals survive and the less fit are discarded. Optimization based on genetic algorithms is a search method based on probability [15].

This is an elitist algorithm as it always conserves the best individual of the population unchanged. As the number of generations or iterations increases, the probability of finding the optimum solution tends to increase.

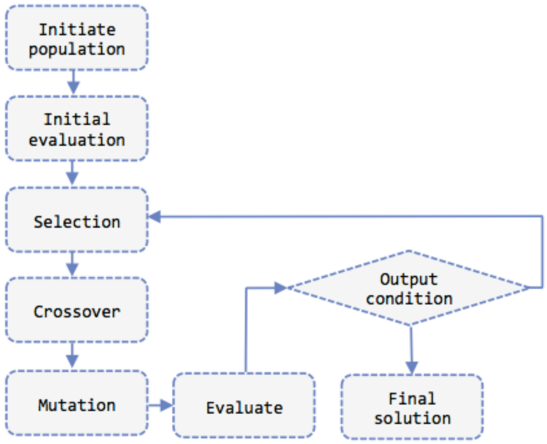


Fig.1. Genetic algorithm diagram.

2.3 Interactive evolutionary computation

Interactive evolutionary computing is a variation of evolutionary computing where the fitness of an individual is determined through the subjective evaluation performed by a human being. In traditional evolutionary computing, a human being requires a computational process to solve a problem. To do this, a person gives a problem's description as input to a solution model, and this model returns a result that has to be interpreted by a human being. But in interactive evolutionary computing the roles are inverted: there's an algorithm that asks a human being or a group of human beings to solve a problem, and then it gathers this information to interpret it later [14].

2.4 Article spinning

Article spinning is a method used to create multiple versions of a text article without creating versions considered as plagiarism, due to the uniqueness achieved of the generated content. Duplicated content is not accepted by several search engines like Google, Yahoo and Bing, so this method is used to generate many different versions of a single article that have a higher probability of being considered as unique content by these search engines. Words and phrases are randomly changed by other text blocks that have the same meaning, resulting in another version of the article with the same meaning, but different text content [11].

2.5 EvoSpace

EvoSpace is a cloud's space or habitat where evolutionary algorithms can be stored, developed, tested and be put into production. EvoSpace is very versatile, as the population is independent to the evolutionary model being used, and this allows us to make modifications to the evolutionary algorithm at any time. The client processes, called EvoWorkers, interact dynamically and asynchronously, and they can be displayed in remote clients like in the platform storing the server [6].

2.6 Online Advertisement

Online advertisement is performed based on the content of a website. For the creation of this type of advertisement, which has the objective of giving information about a product on Internet, it must contain different media elements such as text, links, images, videos, animations, etc. There are companies like Google that have created systems for the creation of online advertisement campaigns, like AdSense and AdWords [4]. AdSense positions ads in websites related to the textual content being displayed in the web pages. Users owning these ads pay a certain amount of money for each click on their ads

2.7 Google AdWords

Google has an online advertising program called "Google AdWords". With this advertising program the user can find new customers and grow the business (see Fig.2), the user chooses where the advertisement appears, set a budget and measure the impact of the advertisement.

There is no minimum spending commitment. Can pause or stop the campaign anytime. The advertisement reach people online as they search for words or phrases called "keywords" or when the users browse websites with themes related to the business. The advertisement can appear on Google and its partner websites. With cost per click called "CPC" bidding, charged only when someone clicks the advertisement.

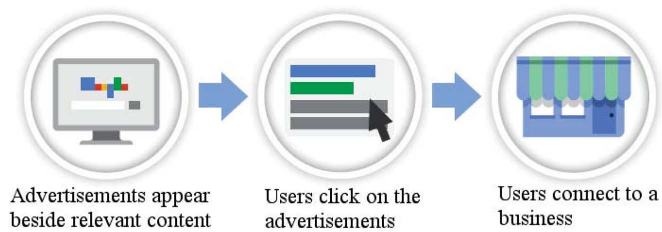


Fig.2. Google AdWords

2.8 Fuzzy logic

Fuzzy sets provide an excellent method for representing some forms of uncertainty; a statement can be neither false nor true and also can be both false and true.

Examples of famous paradoxes cannot be solved using classical logic.

Russell's paradox:

“A barber who himself has a beard shaves all men who do not shave themselves. He does not shave men who shave themselves. “

Answer to question: “Who shaves the barber?” Is contradictory, he shaves himself and he does not shave himself.

Another paradox:

“All Cretans are liars”, said the Cretan if the Cretan is liar then his claim cannot be believed and so is not a liar. If he is not liar then he is telling truth. But because he is Cretan, he must therefore a liar [16].

The main idea in Fuzzy systems is that the membership values are indicated by in the range 0 - 1 where 0 is absolutely false and 1 is absolutely true. The membership grades are not probabilities on a finite universal set must add to 1 while there is no such requirement for membership grades.

Fuzzy set theory is different to the conventional set theory, allows each one element of a given set to belong to that set to some degree but in classical set theory according to which each element either is fully excluded from the set or completely belongs to the set. The Grade of Membership is the degree to which an element belongs to given set.

3 Related Work

In this section we will explain the platform EvoSpace-Interactive, which is the platform that we used to build the graphical interfaced used by the users to choose what ads they considered were more persuasive.

3.1 EvoSpace-Interactive

EvoSpace-interactive was initially tested by implementing an interactive evolutionary computation program called Shapes. This software evolved images formed by equilateral triangles that could have one of twelve possible colors. Currently, EvoSpace has made modifications to the images displayed by changing the shape to more attractive animations that last for a short period of time. For this work, we modified EvoSpace to be capable of displaying text ads instead of images [5].

4 Problem Description

If an inexperienced article writer decided to create many different versions of a text ad, where he changed some words and phrases to other text blocks of similar meaning, and he showed them to some friends and asked them to vote for the version they think is the best one, he would be performing a small optimization to his text. This is because his text isn't based only on his opinion, but on the opinion of all of his friends and his own, and now the winning text could be considered more attractive to a higher percentage of readers.

Based on this example, we believe that a more effective optimization of the text for an ad should take into consideration thousands of possible versions, and dozens of people should give their opinion on what blocks of text are better to increase the persuasion of the text.

5 Methodology

5.1 Article Format

The text (see Fig.3) is contains different sections enclosed by curly braces, which contain different blocks of text (the blocks of text can be of any size, from a single word to whole sentences and paragraphs). These blocks of text are separated by bars. The text blocks that are outside of the curly braces (represented by grey text in Fig.3) won't have any modifications when the texts evolve.

{Diseñamos | Creamos | Fabricamos | Construimos | Desarrollamos} {este auto | este carro | esta pieza de arte | este impresionante transporte | este auto único | este carro único}, {para ser | para convertirse en | para que sea | con el fin de ser} la mejor forma {de viajar | de transportarte | para ir de un lugar a otro | de moverte | de trasladarte} {con tu familia | con tus amigos | a donde quieras | románticamente | cómodamente | de forma divertida | silenciosamente | de forma segura}. Por eso está {equipado | preparado | construido | fabricado | diseñado | creado | desarrollado} con {barras laterales | protecciones a los costados | puertas protegidas | protecciones laterales} contra {impacto | golpes | accidentes | choques}. Puede incluir {frenos ABS | un sistema antibloqueo de ruedas | frenos antibloqueo | frenos reforzados | frenos inteligentes | frenos antiderrapantes} y bolsas de aire {frontales | al frente | delanteras | grandes | suaves | seguras}. {Y por si fuera poco, es maniobrable, estable y eficiente | Cuenta con dirección hidráulica en todas sus versiones | Tiene un motor de 4 cilindros de 1.6L | Tiene transmisión manual de 5 velocidades | Tiene transmisión automática de 4 velocidades | Tiene un espacio interior y cajuela amplia | Por eso este transporte es tu mejor opción | Compra seguro, compra inteligentemente}.

Fig.3. Text format.

A vector generates each version of the text, where each number in it represents the corresponding option. In this example (see Fig.4), if we want to know what number is the word "Fabricamos", we have to look in the options to find out it's the third option. So, in the first position of the vector we'll have the number 3. The rest of the vector represents the rest of the options that should be printed in the generated text.

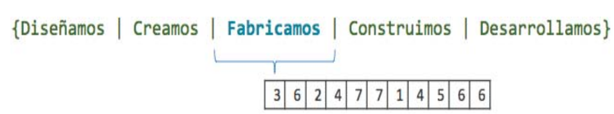


Fig.4. Representation of the vector.

5.2 Implementation of EvoSpace-Interactive

EvoSpace-interactive was initially tested by EvoSpace was modified to be able to evolve advertisement texts. The text, using the format explained before, must be analyzed by the program to determine how many text segments will be changing, and how many and what are their options. This way a vector can be generated, which would represent the chromosome of an individual. EvoSpace creates 100 random individuals when it initializes its population.

Changes were also made to the graphical user interface (see Fig.5). The amount of likes an individual currently has was removed so it wouldn't act as a bias during the decision process of the user. By default, in EvoSpace users can create a collection of their favorite individuals, and this feature was also removed as we consider this isn't necessary for our experiments. The general design was modified so the texts stand out from the rest of the elements of the interface.

5.3 System configuration

The population is initialized with 100 randomly generated individuals. For the evaluation of the individuals, the users are presented with two texts. A user can choose what text (or texts) he considers would be more persuasive. If no text is considered as a good choice, the user can also decide not to choose any of the options. When a selection is made, the selected texts are sent to EvoSpace and stored in its database, and the user is immediately presented with two more options, so he can continue with the selection process.

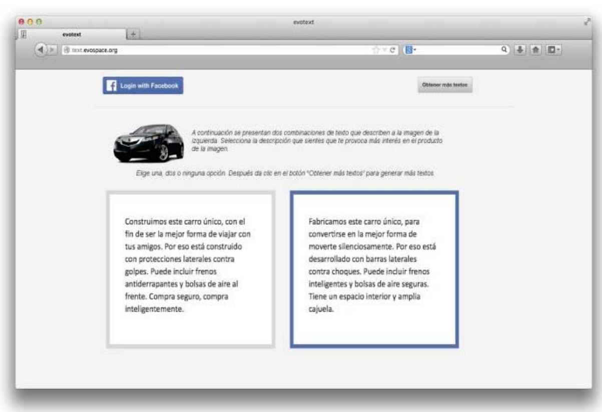


Fig.5.User Interface.

5.4 Implementation of fuzzy control system on Google AdWords

A fuzzy system was implemented on a Google AdWords advertising campaign in order to control the price, get as many impressions and the most clicks using at the lowest possible price. The overall work is designed in individual modules for easier testing and development (see Fig.7). For this reason in the process of testing on Google AdWords with the fuzzy system implementation, it does not involve the text optimization algorithm.

A Mamdani type fuzzy controller was implemented and the input variables are: time, CTR and price. The output variable is the price with a triangular membership functions for each value. The linguistic values were: low, medium and high (see Fig.6).

The fuzzy rules implemented:

```
if time is low and ctr is low and price is low then nprice is low
if time is low and ctr is low and price is medium then nprice is low
if time is low and ctr is low and price is high then nprice is low
if time is low and ctr is medium and price is low then nprice is medium
if time is low and ctr is medium and price is medium then nprice is medium
if time is low and ctr is medium and price is high then nprice is medium
if time is low and ctr is high and price is low then nprice is medium
if time is low and ctr is high and price is medium then nprice is high
if time is low and ctr is high and price is high then nprice is high
if time is medium and ctr is low and price is low then nprice is medium
if time is medium and ctr is low and price is medium then nprice is medium
if time is medium and ctr is low and price is high then nprice is medium
if time is medium and ctr is medium and price is low then nprice is medium
if time is medium and ctr is medium and price is medium then nprice is medium
if time is medium and ctr is medium and price is high then nprice is medium
if time is medium and ctr is high and price is low then nprice is medium
if time is medium and ctr is high and price is medium then nprice is high
if time is medium and ctr is high and price is high then nprice is high
if time is high and ctr is low and price is low then nprice is low
if time is high and ctr is low and price is medium then nprice is low
if time is high and ctr is low and price is high then nprice is low
if time is high and ctr is medium and price is low then nprice is low
if time is high and ctr is medium and price is medium then nprice is low
if time is high and ctr is medium and price is high then nprice is medium
if time is high and ctr is high and price is low then nprice is low
if time is high and ctr is high and price is medium then nprice is medium
if time is high and ctr is high and price is high then nprice is medium
```

Fig.6. Fuzzy rules.

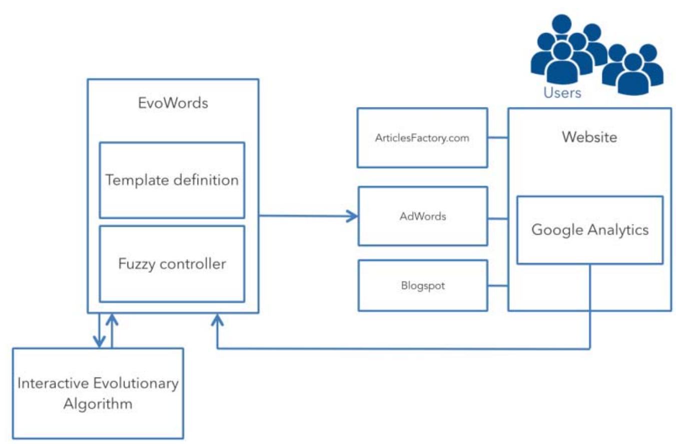


Fig.7. General Architecture.

6 Simulation Results

6.1 Text optimization algorithm simulation results

The system started operation at http://text.evospace.org since December 4th 2013, and a total of 75 users have participated in the selection process, generating more than 180 samples.

The best chromosome was extracted from the database and it was compared against the worst generated chromosome. This best chromosome was also compared against an actual ad that advertises a Chevrolet car created by an expert. This ad, similar in size to the evolved text, was taken from Chevrolet's website [2].

A total of 30 people were surveyed, showing them two paper sheets containing two texts (see Fig.8). On the first paper sheet, the text representing our best individual and the text created by the expert were shown. The second paper sheet showed the texts generated by our best and worst individuals. The texts were shown in different positions to discard the possibility of a person choosing an option due to its position (for example, if a person likes more a text because it's on the right side).

Fig.8.Survey texts.

In our first case, 60% of the people chose the text generated by our genetic algorithm, and 40% chose the text generated by the expert. In the second case, 63.3% of the people surveyed chose the text represented by the best individual, while 36.7% chose the text represented by the worst individual.

6.2 Advertising campaign simulation results

In the first experiment using the fuzzy algorithm we obtained 22 clicks, 760 impressions for a total price of \$32.77MXN (see Fig.8). However it using the default method of Google AdWords we obtained 18 clicks, 559 impressions for a total price of \$28.93MXN (see Fig.9).

In the second experiment using the fuzzy algorithm we obtained 95 clicks, 2087 impressions for a total price of \$58.89MXN (see Fig.10). However it using the default method of Google AdWords we obtained 30 clicks, 752 impressions for a total price of \$50.66MXN (see Fig.11).

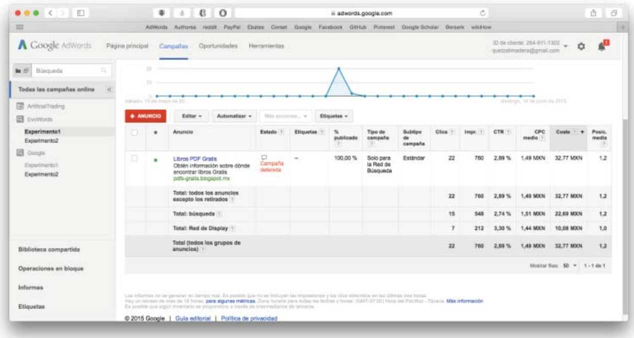


Fig.8. First experiment using a fuzzy algorithm.

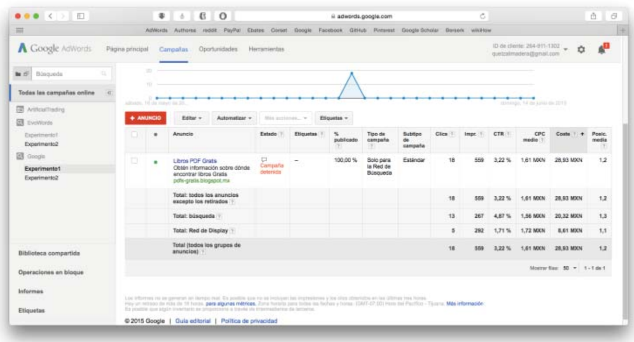


Fig.9. First experiment without a fuzzy algorithm.

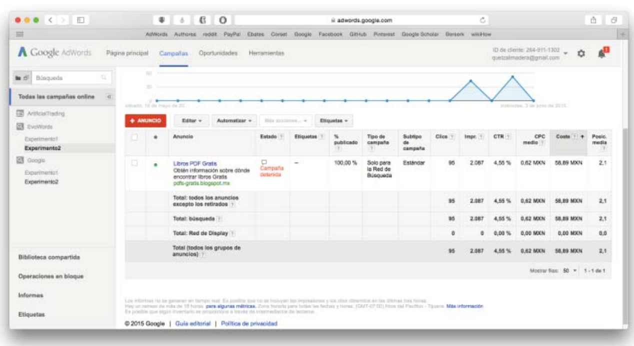


Fig.10. Second experiment using a fuzzy algorithm.

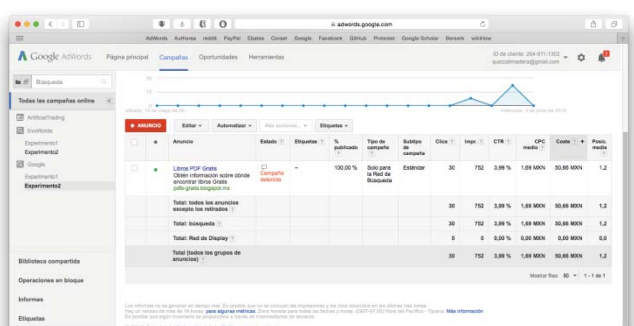


Fig.11. Second experiment without fuzzy algorithm.

7 Conclusions

The evolution of advertisement texts written by an inexperienced person in the field of marketing, through the use of interactive evolutionary techniques, is a viable alternative for the creation of texts with a higher probability of persuading consumers into buying the advertised product. We are currently working on the implementation of a clustering algorithm for grouping [8] users according to their profiles, and perform interactive evolution to generate optimal advertisement texts for each cluster of people. Using a fuzzy system to regulate prices of the Google AdWords campaign seems to have a promising future, but we still need to perform more experiments. We also plan to improve the design of the fuzzy system as in other recent works [17].

Acknowledgment

We thank the Doctoral Computer Science program of the Division of Graduate Studies and Research, of Tijuana Institute of Technology, and the financial support provided by our sponsor CONACYT contract grant number: 189350.

References

[1] Bäck T. Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Vol. 996. Oxford: Oxford university press, 1996.

[2] Chevrolet. "Chevrolet Mexico Ad description ", [online]. December, 12 2013, Website: http://www.chevrolet.com.mx/spark-2014.html

[3] Choi S., Nora J. "Antecedents and consequences of web advertising credibility: A study of consumer response to banner ads." Journal of Interactive Advertising 3.1 (2002): 12-24.

[4] Davis H. Google advertising tools: Cashing in with AdSense, AdWords, and the Google APIs. O'reilly, 2006.

[5] Fernández F., et al. "EvoSpace-interactivo: una herramienta para el arte y diseño Interactivo y colaborativo". IX Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados, (2013): 220-228.

[6] García M., et al. "EvoSpace: a distributed evolutionary platform based on the tuple space model." Applications of Evolutionary Computation. Springer Berlin Heidelberg, 2013. 499-508.

[7] García M., et al. "EvoSpace-i: a framework for interactive evolutionary algorithms." Proceeding of the fifteenth annual conference companion on Genetic and evolutionary computation conference companion. ACM, (2013): 1301-1308.

[8] Hartigan J. Clustering algorithms. John Wiley & Sons, Inc., 1975.

[9] Jong. Evolutionary computation - a unified approach. MIT Press, 2006.

[10] Kowaliw T., Dorin A., McCormack J. Promoting creative design in interactive evolutionary computation. Evolutionary Computation, IEEE Transactions on, 16(4):523 –536, 2012.

[11] Malcolm, James A., Peter L. "An approach to detecting article spinning." Proceedings of the Third International Conference on Plagiarism. 2008.

[12] McQuarrie, Edward F., David M. "Visual rhetoric in advertising: Text-interpretive, experimental, and reader-response analyses." Journal of consumer research 26.1 (1999): 37-54.

[13] Saridakis C., et al. "Advances in Market Segmentation through Nature-Inspired Intelligence Methods: An Empirical Evaluation." Management Intelligent Systems. Springer International Publishing, (2013): 59-66

[14] Takagi H. "Interactive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation." Proceedings of the IEEE 89.9 (2001): 1275.

[15] Whitley D. "A genetic algorithm tutorial." Statistics and computing, Kluwer Academic Publishers 4.2 (1994): 65.

[16] Klir, G. J., & Folger, T. A. "Fuzzy sets, uncertainty, and information" (1988).

[17] Valdez F., Melin P., Castillo O. A survey on nature-inspired optimization algorithms with fuzzy logic for dynamic parameter adaptation. Expert Syst. Appl. 41(14): 6459-6466 (2014)

Realization of Task Intelligence Based on the Intelligence Operating Architecture for Assistive Robots

Bum-Soo Yoo, Yong-Ho Yoo, Woo-Ri Ko, Seung-Jae Lee, Seung-Hwan Baek, Se-Hyoung Cho, and Jong-Hwan Kim
School of Electrical Engineering, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea

Abstract— *Various types of robots have been built and used in our daily life for various purposes. One of them is an assistive robot that can autonomously perform a proper task in a given situation. The robot is required to integrate constituent algorithms into one controlled architecture for the autonomous operation. In this paper, we categorize the constituent algorithms that are developed for functional purposes of assistive robots and integrate them into the intelligence operating architecture to realize task intelligence for the robots. The effectiveness and applicability of the integrated architecture is demonstrated through the experiments with the humanoid robot, Mybot-KSR, developed in the Robot Intelligence Technology Lab., at KAIST.*

Keywords: Adaptive resonance theory (ART), humanoid robot, intelligence operating architecture (iOA), knowledge architecture, mechanism of thought, task intelligence.

1. Introduction

Robots have been used in our daily life for various purposes, such as entertainment, human-aid, personal assistance, etc. Even though it has not been long since robots' introduction in our daily life, robots are already considered capable of making human life more productive and delightful. Among them, personal assistive robots are designed to support humans in their homes and offices. To stay close with humans, they should look friendly to humans and also think and behave as the humans do. Thinking mechanism of the personal assistive robots needs to be engineered similar to human reasoning process since the human intelligence is yet known to be the most efficient for reasoning to generate a sequence of procedures.

The crux of building personal assistive robots is to integrate several constituent algorithms into one controlled architecture, which makes robots autonomously generate a proper behavior in a given situation. Considerable research has been performed in this area. An ontology-based unified robot knowledge framework for integration of low-level data with high-level knowledge was introduced [1]. A robot system producing behaviors from human activities was suggested [2]. It learned human activities and their interactions with objects in the form of associated affordances. Also, an autonomous robot system to learn manipulation action plans was introduced [3]. It used videos from the World Wide Web with convolutional neural network (CNN) and probabilistic manipulation action grammar based parsing module. However, none of these studies presented an integration of memorizing feature as in human brain with other features under high-level architecture controlled with intelligence in more human-like manner.

In this paper, we realize an integrated task intelligence using the intelligence operating architecture (iOA) for the humanoid robot, Mybot-KSR [4]. The iOA is based on the key functions of human brain to develop robot intelligence by governing the five constituent parts: perception, internal state, memory, reasoning, and execution. The robot uses the perception part to perceive environment. The perception module recognizes task-related objects by the CNN [5]. The memory part retrieves the most adequate task from the recognized objects using the adaptive resonance theory (ART), which is followed by retrieving a sequence of procedures for the task [6]-[8]. The robot constructs the ART structure by learning and memorizing the inter-connectivity between situations and tasks, including the sequence of procedures. The reasoning part generates behaviors for performing a task. Each procedure of the task in the memory part can retrieve a proper behavior with objects, which is sub-divided into executable forms, called primitive behaviors. The behavior selection module generates a sequence of primitive behaviors using the degree of consideration-based mechanism of thought (DoC-MoT) [9]. The behavior generation module produces trajectories for performing primitive behaviors with the rapidly-exploring random tree star (RRT*) [10]. Not only the trajectories from the RRT*, the trajectories learned from human behaviors are also used if trajectories are hard to be expressed with points and angles [11]. Besides, the attention module makes the perception more efficiently by controlling the gaze of the robot with a state machine [12]. Finally, the execution part controls motors to generate behaviors of the robot. Results of behaviors are perceived again to update robots behaviors continuously. Note that since the internal state part is not highly related with tasks, it is not used in this paper.

This paper is organized as follows. Section II presents realization of the iOA with constituent algorithms. Section III describes the experimental results with the Mybot-KSR.

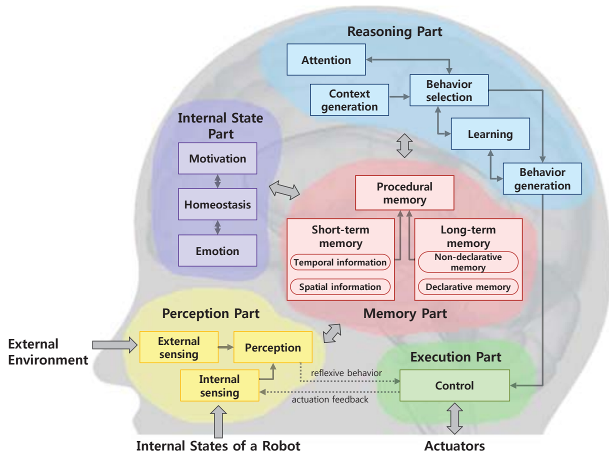


Fig. 1: The intelligence operating architecture.

Finally, the conclusion remarks follow in Section IV.

2. Intelligence Operating Architecture

Fig. 1 shows the intelligence operating architecture to integrate the constituent algorithms for realizing task intelligence of robots. It consists of five parts with fifteen modules. The perception part perceives environment and recognizes task-related objects. The memory part retrieves the most adequate task from the recognized objects. It also retrieves a sequence of procedures for performing tasks. The reasoning part subdivides a sequence of procedures into executable forms. Finally, the execution part controls motors to perform the proper task.

2.1 Perception Part

The robot can perceive external environment with the RGB-D camera through the external sensing module. The perceived visual inputs are delivered to the perception module. Task-related objects are recognized and their information is delivered to the memory part and reasoning part to perform tasks. Note that since the internal state part is not highly related to tasks, it is excluded along with the internal sensing module from the perception part.

In the perception module, a plane in the visual input is found by the random sample consensus (RANSAC) algorithm [13]. Since objects are usually placed on a plane the pixel points in and under the plane are filtered, while the rest points are regarded as objects. Along with it, the CNN for object recognition is actualized by the Convolutional Architecture for Fast Feature Embedding (CAFFE) library. The CNN is a series of layers consisting of neurons. Each neuron in hidden layers is assigned to a part of input neurons, called receptive field, and the CNN recognizes objects based on the receptive field only. As the information is fed forward, the outputs of neurons at each layer are gradually pooled. Also, the receptive field becomes larger, and the recognition becomes more accurate. Unlike the conventional recognition algorithm that tried to design good feature descriptors to extract important information, the CNN has a strong point in learning feature extraction by itself using perceived visual inputs only [5].

2.2 Memory Part

Robots are expected to autonomously infer the most adequate task from its memory for a perceived situation. Also, they need to retrieve a proper sequence of procedures from the memory to perform the task, which is called procedural memory. The procedural memory has been developed based on the ART [7]. The ART structure can memorize interconnectivity between situations and tasks, including the sequence of procedures.

The structure of the adaptive resonance system is shown in Fig. 2. It consists of an input field (F_1), procedure field (F_2), and task field (F_3). The F_1 field receives perceived input from the perception part. In this paper, it encodes the perceived input into five channels: a moving object (F_1^1), an aimed object (F_1^2), a relative position between the moving object and the aimed object (F_1^3), an inclination of the moving object (F_1^4), and an action performed by the moving object (F_1^5). The moving objects are $\{$*bread, toy, bottle, watering pot*$\}$, the aimed objects are $\{$*toaster, dish, box, bowl, flowerpot*$\}$, and the behaviors are $\{$*grasp, move, tilt, put down, push down, pour*$\}$.

The k-th channel is represented as the input vector $\mathbf{Z}^k = [z_1^k \; z_2^k \; \ldots \; z_{n_k}^k]^T$, where n_k is the number of nodes in the k-th channel and $z_i^k \in [0,1]$, i=$1, 2, \ldots, n_k$, indicates a value of the i-th node. Each node in F_1^1, F_1^2, and F_1^5 represents whether objects and action are perceived or not, respectively. Each node in F_1^3 represents the relative position between the moving object and the aimed object along the horizontal, vertical axes, normalized by 10.0 m. A relative depth position is also included. Each node in F_1^4 represents the inclination of the moving object along the roll, pitch, and yaw axes, normalized by π. The input vector $\mathbf{Z}^k$ is converted to an activity vector $\mathbf{X}^k$ by concatenation of the input vector $\mathbf{Z}^k$ and its complements $\bar{\mathbf{Z}}^k$. The F_2 field has one channel, which is represented as $\mathbf{Y} = [y_1 \; y_2 \; \ldots \; y_m]^T$, where m is the number of nodes in F_2 field and $y_j \in [0,1]$, j=$1, 2, \ldots, m$, indicates whether the j-th procedure occurs or not. The j-th node in F_2 field is evaluated with respect to each channel as follows:

$$T_j = \sum_{k=1}^{5} \gamma^k \frac{(\mathbf{X}^k \wedge \mathbf{W}_j^k)}{\alpha^k + |\mathbf{W}_j^k|} \tag{1}$$

where $\alpha^k \geq 0$ is a user-defined bias parameter, $\gamma^k \in [0,1]$ is a user-defined contribution parameter for k-th channel, and $\mathbf{W}_j^k = [w_{j_1}^k \; w_{j_2}^k \; \ldots \; w_{j_{2n_k}}^k]^T$ is a weight vector between the channel k of the F_1 field and the channel of the F_2 field. The $P \wedge Q$ for $P = [p_1 \; p_2 \; \ldots \; p_u]^T$ and $Q = [q_1 \; q_2 \; \ldots \; q_u]^T$ is defined as $\sqrt{\sum_{i=1}^{u} (\min(p_u, q_u))^2}$. The node of the largest

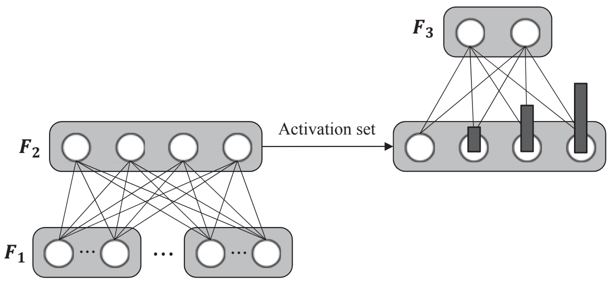


Fig. 2: The adaptive resonance system consists of three fields: the input field F_1, procedure field F_2, and task field F_3.

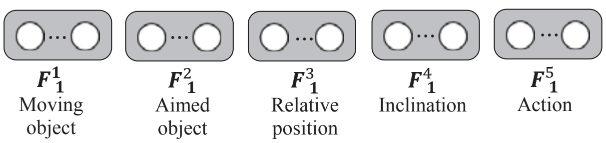


Fig. 3: A set of input vector in each channel.

T_j value is selected, and the resonance is evaluated as follows:

$$M^k = \frac{(\mathbf{X}^k \wedge \mathbf{W}^k_{j_{max}})}{\left|\mathbf{X}^k\right|} \tag{2}$$

where j_{max} is the selected node of the largest T_j value.

When the M^k is greater than the user defined vigilance parameter ρ^k for all k, the resonance occurs and $\mathbf{W}^k_{j_{max}}$ is updated as follows:

$$\mathbf{W}^{k(t+1)}_{j_{max}} = (1-\beta^k)\mathbf{W}^{k(t)}_{j_{max}} + \beta^k \min(\mathbf{x}^k, \mathbf{W}^{k(t)}_{j_{max}}) \tag{3}$$

where $\beta^k \in [0,1]$ is the user-defined learning rate of the k-th channel. If the resonance does not occur for all k, a new node is committed in the F_2 field.

In the learning process, the committed nodes in the F_2 field produce an activation set in the same F_2 field. The activation set is input to produce nodes in the F_3 field, as the F_1 field produces nodes in the F_2 field. The robot memorizes procedures of tasks by the nodes and weights, and uses them to recognize which task is adequate in a given situation.

2.3 Reasoning Part

2.3.1 Behavior Selection Module

After retrieving a sequence of procedures from the memory part, the behavior selection module provides a series of primitive behaviors to complete each procedure and select one of them considering situational information along with the degree of consideration for the information. For example, to accomplish the "grasping a bottle" procedure, the robot should "look around" to find out where the target object is positioned. Then, if a target object is positioned out of reachable range of hands of the robot, firstly the robot should "approach" the target object to grasp. Four situational information and seven primitive behaviors can be defined respectively as shown in Table 1 and Table 2.

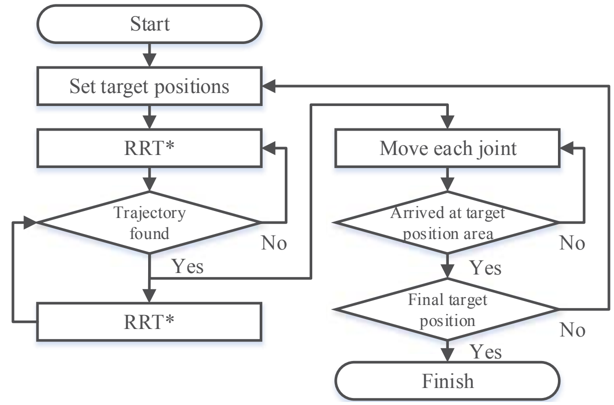


Fig. 4: The flowchart of an arm trajectory generation.

Table 1: Example of situational information.

	Situational information	Unit	Range
1	The distance between the robot's upper body and the target object (s_1)	m	[0,5]
2	The distance between the robot's hand and the target object (s_2)	m	[0,5]
3	The distance between the current and target positions of the grasping object (s_3)	m	[0,5]
4	The difference between the current and target orientation of the grasping object (s_4)	°	[0, 360]

Table 2: Example of primitive behaviors.

	Contents
Primitive behaviors	Look around (b_1), approach (b_2), bend body (b_3), grasp (b_4), move (b_5), tilt (b_6), pour (b_7)

The most appropriate sequence of primitive behaviors in a given situation is selected by the DoC-MoT [9]. The DoC is assigned to the power set of situational information with the λ-fuzzy measure [15]. Having the DoC applied, each primitive behavior, b_i, $i = 1, 2, \ldots, n$, is evaluated by the Choquet fuzzy integral as follows:

$$E(b_i) = \sum_{j=1}^{m} \{l_{ij} \cdot h^t_{ij} - l_{i(j-1)} \cdot h^t_{i(j-1)}\} g(S_j) \tag{4}$$

where $h_{ij} \in [0,1]$ is the user-defined partial evaluation value of b_i over j-th situational information s_j, m is the number of situational information, $L = (l_{ij})_{m \times n}$ is the knowledge link strength matrix in the learning module, and $g(S_j)$ is the DoC applied to S_j with assumptions that $h_{ij} \le h_{i(j+1)}$ and $S_j = \{s_j, s_{j+1}, \ldots, s_n\}$. After evaluating all primitive behaviors, the primitive behavior of the largest evaluation value is selected and is transferred to the behavior generation module.

2.3.2 Learning Module

The robot requires a proper link strength matrix in (4) to perform correct primitive behaviors in a given situation. Since the robot does not know a suitability of its behaviors, it needs to use a user-recommended behavior to evaluate the suitability between behaviors and situations. All l_{ij} are initially assigned as 1.0 and is updated by the user-recommended behavior, b_r, as follows:

$$l_{rj}^{t+1} = l_{rj}^{t} + \eta h_{rj}^{t} \quad (5)$$

where $\eta \in [0, 1]$ is the learning rate. The trained knowledge link strength matrix is normalized to satisfy $\sum_{i,j} l_{ij} = 1$ and used in (4).

2.3.3 Behavior Generation Module

The robot needs to generate a trajectory of its arms to manipulate objects such as tilting, grasping, and pouring. The RRT* can be used to generate a trajectory [16]. The RRT* is one of sampling-based algorithms that can generate optimal collision-free trajectories [10]. Fig. 4 shows the flowchart of the arm trajectory generating algorithm. From the behavior selection module, a primitive behavior that requires movement of an arm is transferred. Then, the behavior generation module chooses a closer arm to the target position, and the RRT* algorithm searches for an optimal trajectory from the hand of the chosen arm to the target position. The robot starts to move its arm when the initial trajectory is produced, and it stops movement when the hand arrives at the target position. During the movement, the robot continuously calculates the current position of the hand to monitor whether it has arrived at the target position or not. At the same time, the RRT* algorithm updates continuously to generate the optimal trajectory. Note that, if a sequence of target positions is produced, the robot generates trajectory to each target position sequentially.

After arriving the final target position, the robot needs to perform the behaviors of not only grasping objects, but also performing tasks on the grasped objects. Unlike the trajectories generated by the RRT*, generating trajectories for the grasping object is sometimes as inconvenient to be expressed by a series of coordinates and angles as generating trajectories for shaking. The robot can learn behaviors for the task from human's demonstration. The Gaussian mixture model (GMM) extracts dynamic characteristics of the multiple trajectories from the human's demonstration, and the motion trajectory morphing (MTM) algorithm generates a modified behavior trajectory for current environment by maintaining the dynamic characteristics [11].

2.3.4 Attention Module

The robot should acquire information of its surrounding environment to successfully perform tasks. Paying attention to proper targets at the right time can be a good solution to acquire the information. Since the gaze represents where the robot is paying attention to, the attention module controls the gaze of the robot. The attention module allows the robot to acquire situational information to perform tasks with its limited field of view and to indicate that the robot is concentrating on current tasks [12], [17].

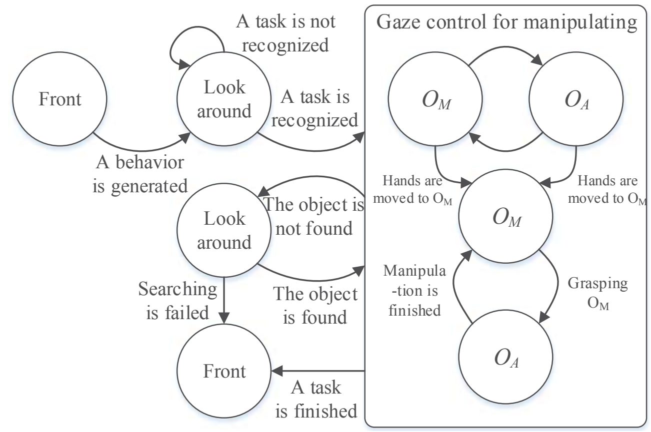


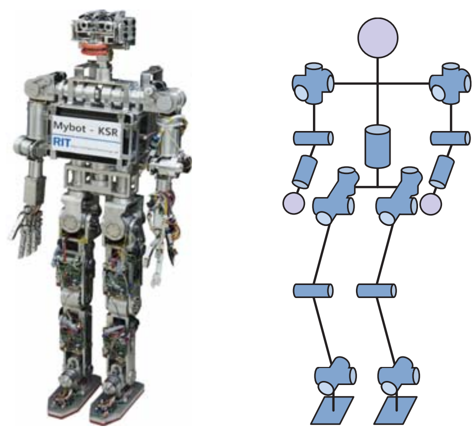
Fig. 5: A state machine in the attention module.

Most of people, who pursue the same goal, usually require similar information for the given task. It produces a large similarity in human gaze. Therefore, a state machine is used to designate targets for gazing in a sequence of procedures. Fig. 5 shows the state machine for the gaze control used in the experiments, where O_M and O_A represent the moving object and aimed object, respectively. At first, the robot gazes at the front. The behavior generation module in the reasoning part selects the "look around" behavior, and the robot starts looking around to perceive environment. From the perceived environment, the robot retrieves a sequence of procedures to perform a given task. Then, the robot performs its gaze control according to the state machine, until it finishes the task or the behavior generation module selects the "look around" behavior. The "look around" behavior is selected, when the objects disappear. The robot resumes the gaze control after detecting the objects, and it stops the gaze control when it fails to detect objects for a certain time period.

3. Experiments

3.1 Mybot-KSR

The Mybot-KSR is the humanoid robot developed in the Robot Intelligence Technology (RIT) Lab., at KAIST. Fig. 6 shows the Mybot-KSR and its configuration. The height of the robot is 142.0 cm and the weight is 58.0 kg. Degrees of freedom (DoFs) of each part is listed in Table 3. The robot is equipped with one RGB-D camera and four microphones on its head. The arms are built using integrated joint modules: a position sensor, a torque sensor, a motor and a harmonic



(a) (b)

Fig. 6: (a) The Mybot-KSR used in the experiment. (b) Configuration of the Mybot-KSR.

Table 3: Degrees of freedom of the Mybot-KSR.

Part	Arms	Hands	Head	Legs	Waist	Total
DoF	5×2	5×2	17	6×2	1	50

drive gear. The feet are mounted with a force-torque sensor to measure foot pressure.

Two PCs are used to control the robot. One with the i7-2670QM 2.2GHz CPU operates the iOA to decide behaviors from perceptive input, and the other with the Intel Atom N570 CPU controls posture by the 17 digital signal processors (DSPs), the TMS320C2846. The controller area network (CAN) are used to transfer control signals to the DSPs at every 10 ms, and each DSP controls motors at every 0.5 ms.

3.2 Experimental Results

The experiment was carried out in the lab environment. The robot had the yellow and red bottles, a mustard, and a bowl on the table in front of it. The robot inferred which task would be best fit in the given situation, mainly by identifying objects. Fig. 8 shows the snapshots of the experiments. On the first row are the images of the appearance of the robot with the table, and the second and third rows respectively show the RGB and depth images taken by the robot. In the RGB images, the detected objects, tracked object, and gaze direction are respectively marked with boxes, ellipses, and a circle. In the depth images, the plane is filtered out by the RANSAC, and a pair of images in each column were taken at the same time step. Note that in the RGB images, the bowl is supposed to be caught in the ellipse, but not due to its white color.

When the robot experienced difficulty in finding objects in its field of view it autonomously selected one of the primitive behaviors, “look around”, to find objects. After looking around, robot had a set of objects in its field of view: the yellow and red bottles, mustard, and a bowl, as shown in Fig. 8(a). From the recognized objects, it retrieved a specific memory by the ART on pouring the liquid inside the red bottle into the bowl from the procedural memory. Fig. 7 describes the retrieved sequence of procedures and the selected primitive behaviors to accomplish each procedure. The robot utilized the four situational information in Table 1 to evaluate each primitive behavior in Table 2. Reaching out its hand toward the target object to realize it was far to grasp, the robot selected the “approach” primitive behavior. Fig. 8(b) shows the robot approaching the target object using the modifiable walking pattern generator [18]. The robot stopped approaching when it reached the area in which the target object is close enough to grasp.

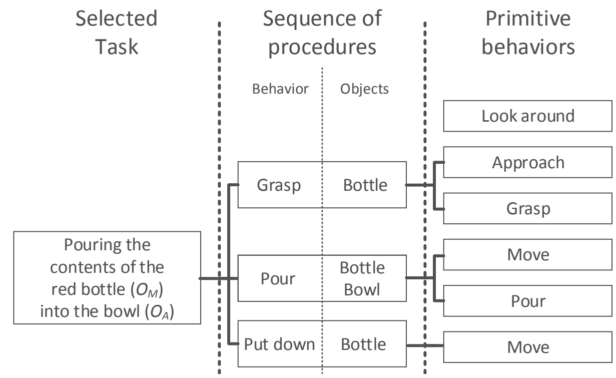
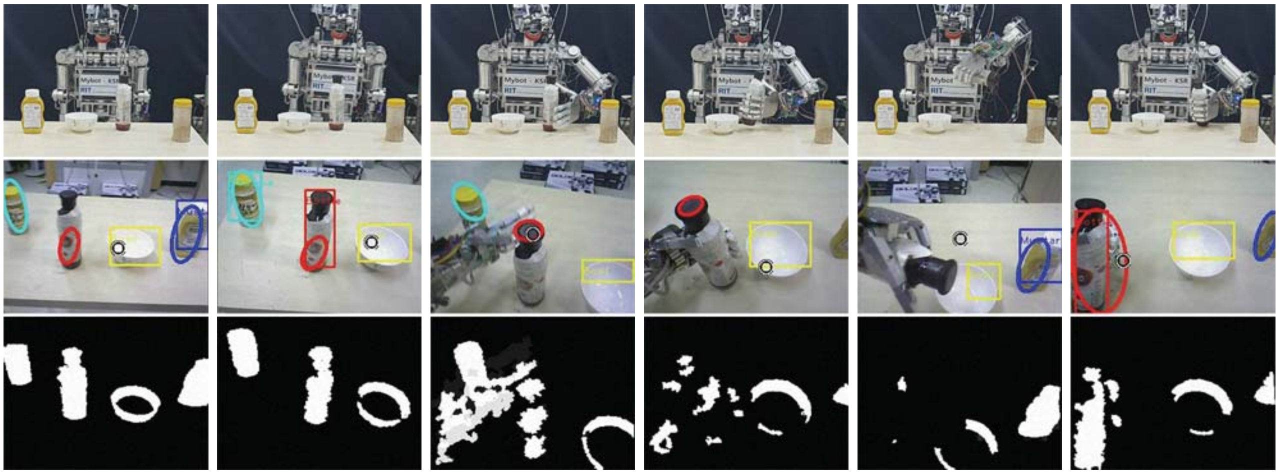


Fig. 7: The retrieved procedural memory with the sequence of procedures and the provided primitive behaviors.

To grasp the object, the robot chose the arm closer to the object and shifted the arm to the object using the RRT*, as shown in Fig. 8(c). For this movement, the robot used the “grasp” primitive behavior. With the help of the RRT*, the robot produced collision-free trajectories to avoid the table and other objects such as the yellow bottle. After grasping, the robot generated the “move” primitive behavior to take its arm above the bowl, followed by the “pour” primitive behavior, as shown in Figs. 8(d) and (e). Since the trajectories for the “pour” primitive behavior are inconvenient to be expressed by a series of coordinates and angles, they had previously been memorized through learning from human demonstration. The memorized behavior was modified suitable to the environment by the MTM algorithm. Finally, the robot replaced the red bottle to its original place and completed the task, as shown in Fig. 8(f). During the experiments, the robot produced appropriate gaze directions by the state machine.

4. Conclusion

In this paper, we developed and presented the iOA embedded robot system to carry out a given task with task intelligence. The iOA integrated constituent parts for key functions of human brain. This integrated iOA enabled to



(a) (b) (c) (d) (e) (f)

Fig. 8: Snapshots of the experiments. (a) The Mybot-KSR looked around and found the objects. (b) The robot walked to the target object. (c) The robot moved the arm to the object. (d) The robot moved the grasped red bottle to the bowl. (e) The robot poured the contents of the red bottle into the bowl. (f) The robot finished its task by taking the red bottle back to its original place.

memorize detected objects and related human behaviors to pertain current situation with the corresponding sequence of procedures, and to retrieve a relevant set of primitive behaviors. The experimental results showed that the robot embedded with the integrated iOA autonomously retrieve a sequence of procedures for a task in a given situation and successfully completed it through the reasoning process.

Acknowledgement

This work was supported by the Technology Innovation Program, 10045252, supported by the Ministry of Trade, Industry, and Energy (MOTIE, Korea).

References

[1] G. H. Lim *et al.,* "Ontology-based unified robot knowledge for service robots in indoor environments," *IEEE Trans. Syst. Man Cybern. A., Syst. Humans,* vol. 41, no. 3, pp. 492-509, May 2011.

[2] H. S. Koppula, *et al.,* "Learning human activities and object affordances from rgb-d videos" *Int. J. Robot. Research*, vol. 32, no. 8, pp. 951-970, Jul. 2013.

[3] Y. Yang *et al.,* "Robot learning manipulation action plans by "watching"unconstrained videos from the world wide web," submitted for publication.

[4] J.-H. Kim, *et al.,* "Intelligence technology for robots that think," *IEEE Comput. Intell. Mag.,* vol. 8, no. 5, pp. 70-84, Aug. 2013.

[5] Y. LeCun *et al.*, "Gradient-based learning applied to document recognition," *Proc. IEEE,* vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

[6] W. Wang, *et al.,* "Neural modeling of episodic memory: Encoding, retrieval, and forgetting," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 10, pp. 1574-1586, Aug. 2012.

[7] Y.-H. Yoo and J.-H. Kim, "Procedural memory learning from demonstration for task performance" in *Proc. Int. Conf. Systems, Man, and Cybernetics,* Hong Kong, China, 2015, to be published.

[8] G.-M Park *et al.,* "REM-ART: Reward-based electromagnetic adaptive resonance theory" in *Proc. Int. Conf. Artificial Intelligence,* Las Vegas, NV, USA, 2015, to be published.

[9] J.-H. Kim *et al.*, "The degree of consideration-based mechanism of thought and its application to artificial creatures for behavior selection," *IEEE Comput. Intell. Mag.,* vol. 7, no. 1, pp. 49-63, Jan. 2012.

[10] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Research*, vol. 30, no. 7, pp. 846-894, Jun. 2011.

[11] S. Cho and S. Jo, "Incremental online learning of robot behaviors from selected multiple kinesthetic teaching trials," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 43, no. 3, pp. 730-740, Sep. 2013.

[12] B.-S. Yoo and J.-H. Kim, "Fuzzy integral-based gaze control of a robotic head for human robot interaction," *IEEE Trans. Cybern.,* to be published.

[13] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for modeling fitting with applications to image analysis and automated cartography," *Commun. ACM,* vol. 24, no. 6, pp. 381-395, Jun. 1981.

[14] W.-R. Ko and J.-H. Kim, "Behavior selection method of humanoid robots to perform complex tasks," in *Proc. Int. Conf. Robot Intelligence Technology and Applications*, Beijing, China, 2014, pp. 127-133.

[15] M. Sugeno, "Fuzzy measures and fuzzy integrals: A survey," in *Proc. Fuzzy Automata and Decision Processes*, vol. 78, no. 33, M.M. Gupta, G.N. Saridis and B.R. Gaines, eds. Amsterdam, The Netherlands: North-Holland, 1977, pp. 89-102.

[16] S.-J. Lee *et al.,* "Arm trajectory generation based on RRT* for humanoid robot," in *Proc. Int. Conf. Robot Intelligence Technology and Applications,* Beijing, China, 2014, pp. 373-383.

[17] Y. Ye and J. Tsotsos, "A complexity-level analysis of the sensor planning task for object search," *Comput. Intell.,* vol. 17, no. 4, pp. 605-620, Nov. 2001.

[18] B.-J. Lee *et al.,* "Modifiable walking pattern of a humanoid robot by using allowable ZMP variation" *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 917-925, Jul. 2008.

Automated optimization for web page design based on user interaction analysis

A. Bruna Santos[1], B. Adriano Cruz[2], and C. Rodrigo de Toledo[2]
[1]PPGI, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Rio de Janeiro, Brazil
[2]PPGI, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Rio de Janeiro, Brazil

Abstract— *The evolution of web technologies allows us to create unlimited designs for web graphical user interfaces. Finding the best design is not an easy task, since combinations of web elements and changes in their style generate multiple possibilities. This paper intends to validate the possibility of optimizing (without human intervention) designs of web pages in real-time through: i) the development of a tool that generates different versions of web pages, collects and analyzes data about user behavior in real-time and improves those versions so that the best design - the one that is closest to the business goal - can be reached; ii) the usage of this tool in real web systems; iii) analysis of results.*

Keywords: design, website optimization, genetic algorithm, web analytics, controlled experiments.

1. Introduction

Advances in web technologies allow us to design highly sophisticated web graphic user interfaces to meet requirements of increasingly demanding users. HTML (Hypertext Markup Language) describes those interfaces through elements (e.g. texts, buttons and images) [12] whose properties (e.g. color, size and position) can also be changed [13]. Changes in an element's style may completely modify its appearance (fig. 1).

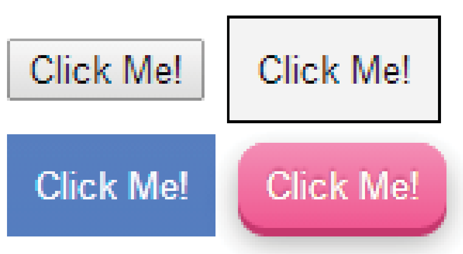


Fig. 1: Different designs for a button element. From the left to the right, from top to bottom: a) original button; b) button with a modified border and background color; c) button without border, with modified background color and font; d) button with modified border, background color and font.

Combining elements and modifying their properties leads to a wide range of interface designs. However, is there a way for us to choose the best one?

There are a few ways to decide the best design: through intuition about what a good design is or by means of analytics data. Human sensitivity provides qualitative data that may not properly understand user needs. This may lead to misconceptions about the relation between the users and the system [5]. On the other hand, data collected from users' behavior through the website and their analysis can help improving customer experience and also lead to successful business decisions [5], [3], [10]. These type of data can be collected through web analytics tools, like Google Analytics [1] and Piwik [9].

Previous papers used artificial intelligence algorithms to create designs for web graphic user interfaces although selection of candidates were based on intuition [8], [7], [14], [11].

This paper aims at optimizing web interface designs focusing on business without any human intervention during the evaluation process. To achieve our goal, we combined techniques to develop an application that creates, manages and evaluates different designs for a web page intending to find the best design. Techniques that serve as foundations to this work are: genetic algorithms [2], [6], web analytics [5], [3] and controlled experiments [3].

Seeing that we have a wide range of possible designs - and considering we can generate many combinations between HTML elements and styles - genetic algorithms offer a technique for design optimization. It still finds a solution, even if it may not be optimal [6].

Data collected through web analytics tools provide means to: a) avoid human intervention when deciding the best design; b) keep focus on business. They provide important feedback to decide if both system and business goals are in agreement [5].

Controlled experiments provide a methodology to regulate the real-time testing of designs and enable the validation of changes based on metrics of interest for business [3], [10].

This work differs from others because: a) It chooses the best design without human intervention; b) It evaluates designs through quantitative data and achieves business goals; c) It enables that the testing process occurs indefinitely.

2. Experimentation tool

To validate web pages design optimization in real-time, we developed a tool - named UIDesign - that generates versions for web pages, collects and analyses data about user behavior and improves those versions to find the best design. This section will discuss techniques to create UIDesign, its architecture and how it can be used in web applications.

2.1 Architecture

2.1.1 Design generation

When a user requests a web address, he receives a web page as a response. After that, he can finally interact with it, as shown in fig 2.

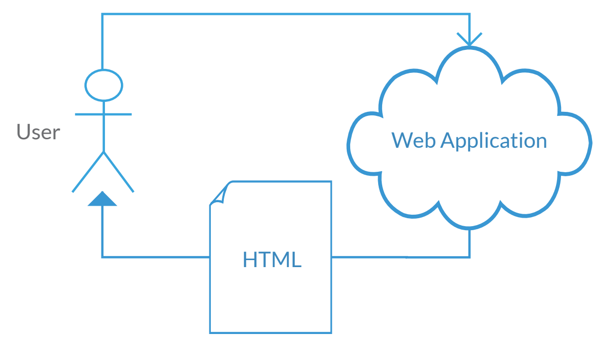


Fig. 2: User receives a web page as a response to his request

During a UIDesign experiment, as soon as the web application sends a web page back - that means, before user can interact with the page itself - it requests to UIDesign changes that should be applied to its elements. After receiving and carrying out these changes, the web page will have a new design. Only then, the user will be able to interact with the web page (fig 3).

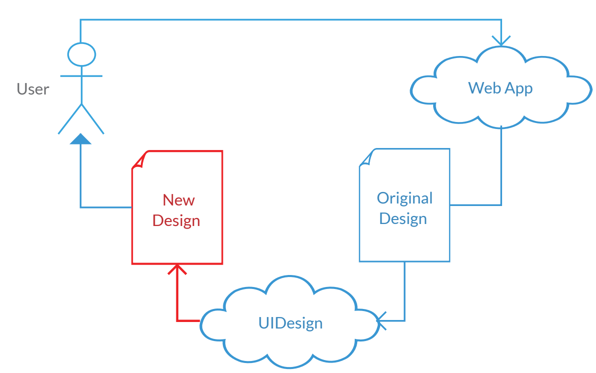


Fig. 3: Web application sends a web page back after a user's request. UIDesign will carry out changes to the web page's design. The User will interact with the modified web page.

This approach has two advantages: 1) UIDesign architecture can be completely different from web application architecture. This allows reusing UIDesign in many systems and also contributes to simultaneous experiments to different web applications; 2) UIDesign requires small changes in web application to test new designs because implementing these changes to the new design generation is a UIDesign responsibility.

2.1.2 Design evaluation

As soon as the user may interact with the web page, a web analytics tool collects information, that is relevant to business, about user behavior (fig 4). Each information is associated to a design (by its ID). This allows identifying, for example, how many clicks were made in a button when it was red or blue and thus, describing which design is the best.

Fig. 4: After receiving the new design, user interacts with it and a configured web analytics tool collects information about user behavior within the website.

2.2 Techniques

UIDesing combines three techniques to optimize web interfaces: genetic algorithms, web analytics and controlled experiments. Next sections will discuss their usage in UIDesign.

2.2.1 Genetic Algorithms

Since elements have a variety of modifiable properties, it is possible to create lots of designs for a single web page. Each design is a candidate solution (individual) for the optimization problem that this paper proposes. For this kind of problem, in which the search space is huge, genetic algorithms offer a technique to resolve it [2], [6].

An individual is represented by a string of bits (zeros and ones) and its length varies according to possible changes in elements. A genome describes which changes should be applied to web page elements to generate a design for testing (fig 5).

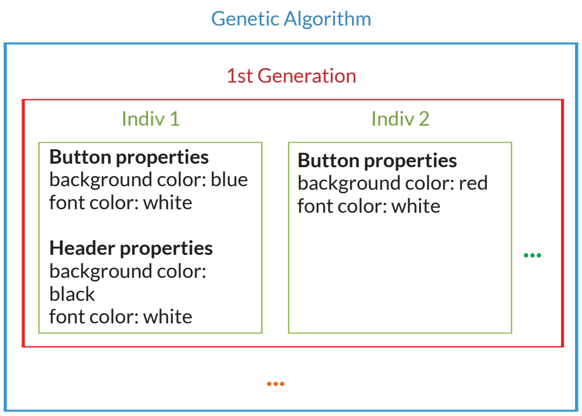


Fig. 5: An individual represents changes that should be applied to elements of the original design.

For example, suppose that there are three possible changes for a button element: a) set background color to blue and font color to white; b) set background color to red and font color to white; c) set background color to pink and font color to white. Also two possible changes for an header element: i) set background color to white and font color to black; ii) set background color to black and font color to white.

In this example, individuals are represented by four bits. The first two correspond to changes that will be applied to button elements, the last two to the header element. Tables 1 and 2 show all possible genomes and which changes they represent. As an example, the individual 0110 represents a design in which the button will have background color set to blue, font color set to white and the header will have background color set to black and font color set to white.

Table 1: Individuals that will change buttons design

Individual	button
00XX	applies nothing (keep original)
01XX	applies (a)
10XX	applies (b)
11XX	applies (c)

Table 2: Individuals that will change headers design

Individual	Header
XX00	applies nothing (keep original)
XX01	applies (i)
XX10	applies (ii)
XX11	applies nothing (keep original)

Fitness function takes data collected by a web analytics tool to evaluate individuals. Those data are related with business goals and they will be optimized by genetic algorithms. For example, if the business needs to increase the amount of clicks in a certain button, the individual with the highest value is closer to be considered the best one. The Roulette method is used to select individuals to crossover. Crossover occurs with probability of 0.8 and mutation with probability of 0.01.

2.2.2 Web analytics

Web analytics provides methods to collect and analyze data from interactions between users and a web system [5]. There are tools available with this purpose, like Google Analytics [1] and Piwik [9].

UIDesign uses data collected by a web analytics tool to evaluate designs. Only data that is useful for business is used in this process. So, design will be optimized focusing on business interests.

2.2.3 Controlled Experiments

Controlled experiments offer a methodology to validate ideas in real-time and they are used to both design and feature evaluation [4]. The advantage lies in validating changes based on collected and analyzed data from users' behavior within a web page instead of personal opinions. The main idea behind controlled experiments is that they expose different versions of an interface to different user groups simultaneously, collect and analyze data that are related to business interests.

UIDesign explores the concepts of controlled experiments as it shows different versions of the original design to many different user groups and also assess how much those changes impact on users' behavior. Consequently, it is possible to validate which designs achieve business goals and therefore which ones are the best.

2.3 Interaction

UIDesign has two steps to optimize web page designs. The first one consists of experiments configuration, in which common information about each experiment (e.g. number of versions to evaluate, experiment duration etc.) is set. The second one consists of experimentation itself. An experiment consists of: a) selection of changes that will be applied to elements; b) exhibition of generated design to each user group; c) collection and analysis of data about each version.

2.3.1 Configuration

UIDesign receives a set of parameters from an administrator (admin) user. The admin user is represented by each client that is interested in design experimentation, for example, the web application owner. Those parameters are relevant information to experiments accomplishments: a) Number of versions/designs to be generated for each experiment; b) Duration of one experiment; c) Maximum number of experiments; d) Changes that will be applied to HTML elements; e) Web analytics tool to data extraction; f) Metrics of interest (i.e. relevant data for business).

There are four steps to collect those parameters (fig 6):

- access step: admin user gives a user name and a password to log in or sign in;
- genetic algorithms configuration step: admin user provides number of designs that can be generated per experiment and which changes are allowed to be applied to web page elements. Experiment duration is fixed and there is no maximum number of experiments.
- web analytics tool configuration: admin user provides information that will be used by genetic algorithms to access the web analytics tool and to evaluate versions (i.e. metrics of interest).
- experiment start step: the whole process of experimentation is started.

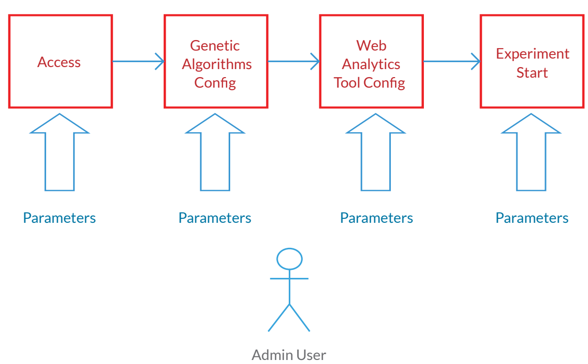


Fig. 6: The configuration phase is divided into steps. Each step has a set of parameters. When an admin user returns to UIDesign, he proceeds from the last step. It is not possible to change parameters from previous steps.

2.3.2 Running experiments

Experimentation starts after configuration step is completed. This phase is composed by sequences of controlled experiments which are represented by a genetic algorithm generation. At the end of each experiment, individuals are evaluated and they generate the next population, i.e., the next controlled experiment. This process continues until admin user is satisfied.

Each request to the web application receives a web page as a response. This page then requests to UIDesign changes that should be applied to its elements. UIDesign randomly selects an individual (fig 7) from its current generation. Changes proposed by selected individual are applied to page design. Notice that different users can interact with a web page modified by the same individual (i.e., they will have the same appearance).

Now that the user can interact with the web page, each interaction between them is collected and stored by a selected web analytics tool as described in section 2.1.2. At the end of several days, each version is evaluated based on collected data.

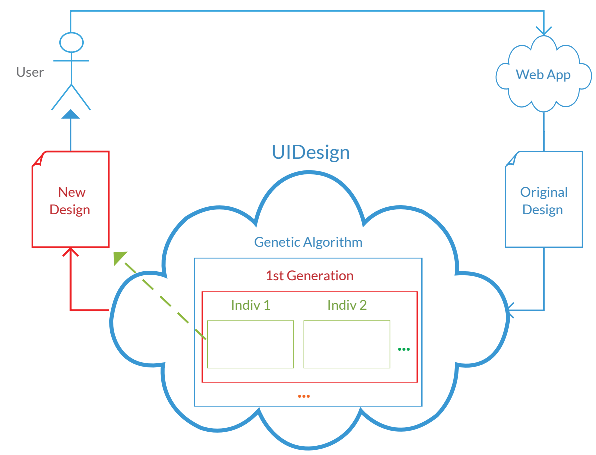


Fig. 7: User interacts with a web page whose design is modified by a random individual of the current generation (e.g. the first individual of first generation).

Genetic algorithm proceeds and creates the second generation. Crossover and mutation generates the next candidates solution based on selected individuals from the previous generation. At this moment, all requests to web application will show UIDesign version of the second generation.

At any time, the admin user can access UIDesign to visualize which individual/version provides the best results for business.

3. Experiments

To prove that interfaces can be optimized without human intervention, focusing on business, we applied UIDesing concept into a real web system.

3.1 Portal Aluno

Portal Aluno (portalaluno.ufrj.br) is a web system that allows UFRJ (Universidade Federal do Rio de Janeiro) students to access on-line services offered by the university, such as registration for courses, teacher evaluation and academic documents printing. Experiments with UIDesign are being conducted in academic documents printing service.

The main goal of the service (fig 8) is reducing the average time a user spends in page. As each user demand resources from server, the sooner he selects a document, the sooner those resources will be released.

3.1.1 UIDesign configuration

To meet business objective, UIDesign configuration parameters are:

- Number of versions/designs to be generated for each experiment : fourteen;
- Duration of one experiment: seven days;
- Maximum number of experiments: there is none;
- Changes that will be applied to HTML elements: basically in size of icons and font of their labels;

Fig. 8: Original design of selected service

Fig. 9: Design represented by individual 001100

Fig. 10: Design represented by individual 100011

- Web analytics tool to data extraction: Google Analytics;
- Metrics of interest (i.e., relevant data for business): minimizing the average time a user clicks on a document icon.

3.1.2 Results

Table 3 shows first generation individuals on the left column and the average time a user clicks a document icon on the right.

Table 3: First generation individuals and average time users take to select a document for printing

Individual	Avg. Time (seconds)
000001	98.15
000010	45.75
000110	77.52
000111	69.23
001100	149.60
001101	77.04
010000	81.08
011001	75.62
100011	44.51
101101	46.19
110000	84.03
111000	58.03
111101	64.68
111111	84.44

Looking at table 3, some individuals stand out. Individuals 000110 e 000111 do not apply changes to the web page, so they are equivalent to the original design. As we expected, changes in original interface cause impact on users and, consequently, on how they interact with the system and also with metrics.

To this experiment, the best design is represented by individual 100011 (fig 10), since it represents the lowest average time. Meanwhile, the worst case is represented by individual 001100 (fig 9).

4. Conclusion

Finding the best design for a web page is not an easy task. Elements and their properties can be combined, generating multiple possibilities. To opt for the best one, we usually trust our instincts. However, they might trick us as well [3].

For this reason, controlled experiments [4] offer a methodology to evaluate ideas based on analyzed data from users' behavior. Meanwhile, web analytics [5] provide means to collect and analyze information about how users interact with the website.

This paper proves that automated web page design optimization is possible. Therefore, we can get to the best web page design without any human interaction during the decision process. To accomplish this, we combined controlled experiments, web analytics and genetic algorithms to develop a tool - named UIDesign - and used it in a real world web application.

References

[1] (2015) Google Anaytics website. [Online]. Available: http://www.google.com/analytics/

[2] Jyh-Shing Roger Jang and Chuen-Tsai Sun and Eiji Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Ed. Prentice Hall, 1997.

[3] Kohavi, Ron and Henne, Randal M and Sommerfield, Dan, "Practical guide to controlled experiments on the web: listen to your customers not to the hippo" in *Proc. of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 959–967, 2007.

[4] Kohavi, Ron and Longbotham, Roger and Sommerfield, Dan and Henne, Randal M, "Controlled experiments on the web: survey and practical guide" *Data Mining and Knowledge Discovery*, vol. 18, pp. 140–181, 2009.

[5] Kumar, Lakhwinder and Singh, Hardeep and Kaur, Ramandeep, "Web Analytics and Metrics: A Survey" in *Proc. ICACCI'12*, 2012, pp. 966–971.

[6] Mitchell, Melanie, *An Introduction to Genetic Algorithms*, Ed. Cambridge, USA: MIT Press, 1996.

[7] A. Oliver and N. Monmarch and G. Venturini, "Interactive Design Of Web Sites With A Genetic Algorithm" in *Proc. of the IADIS international conference WWW/Internet*, pp. 355–362, 2002.

[8] Penalver, J González and Merelo, JJ, "Optimizing web page layout using an annealed genetic algorithm as client-side script" *Parallel Problem Solving from NatureâĂŤPPSN V*, pp. 1018–1027, 1998.

[9] (2015) Piwik website. [Online]. Available: http://piwik.org

[10] Dan Siroker and Pete Koomen, *A/B Testing: The Most Powerful Way to Turn Clicks Into Customers*, Ed. Wiley, 2013.

[11] Sorn, Davy and Rimcharoen, Sunisa, "Web page template design using interactive genetic algorithm" *Computer Science and Engineering Conference (ICSEC), 2013 International*, pp. 201–206, 2013.

[12] (2015) HTML Elements website. [Online]. Available: http://www.w3schools.com/html/html_elements.asp

[13] (2015) HTML Styles website. [Online]. Available: http://www.w3schools.com/html/html_styles.asp

[14] Yokoyama, Takahiro and Takenouchi, Hiroshi and Tokumaru, Masataka and Muranaka, Noriaki, "Website Design System Based on an Interactive Genetic Algorithm Using Tournament Evaluation by Multiple People" in *Proc. of SCIS-ISIS*, pp. 2260–2263, 2012.

REM-ART: Reward-based Electromagnetic Adaptive Resonance Theory

Gyeong-Moon Park, Yong-Ho Yoo, and Jong-Hwan Kim
School of Electrical Engineering, KAIST, Daejeon, Republic of Korea
Email: {gmpark, yhyoo, johkim} @rit.kaist.ac.kr

Abstract— *This paper proposes a Reward-based Electromagnetic Adaptive Resonance Theory (REM-ART) mainly for reducing the retrieval error, which can store and retrieve episodes consisting of a temporal sequence of events. With the existing EM-ART it is difficult to predict a correct episode when there are noisy inputs that are sparse or distorted. To overcome this problem, the proposed REM-ART has a reward channel along with an event channel, lending it robustness to noisy inputs. The reward channel generates a reward signal to reinforce the activation of a proper episode node. The proposed REM-ART is applied to predict and retrieve a correct episode from noisy inputs, out of episodes that are stored through the visual demonstration using an RGB-D camera.*

Keywords: EM-ART, episode retrieval, episodic memory, fusion ART, REM-ART, reward channel.

1. Introduction

Semantic memory and episodic memory are the main categories of the declarative memory in the human brain. These concepts have been illustrated in detail by Tulving [1], [2]. Semantic memory represents generalized knowledge and meaning, and it is not related with individual experiences. Semantic memory is objective information such that anyone can share the meaning, akin to a pen. Episodic memory, on the other hand, is the subjective memory of an individual, which is composed of his or her own experiences, like the first day of attending school. This memory contains contextual connections of several events. For this reason, episodic memory can be implemented for multiple cognitive capabilities, such as sensing, reasoning, and learning [3], [4]. It is helpful for the human to retrieve whole sequential events, i.e., an episode at a specific time from only partial events observed [5].

Recently, many researchers have introduced computational models of episodic memory. Ho *et al.* [6] proposed an autonomous agent having a sequential character of episodic memory. Mueller and Shiffrin presented a model representing the relation between semantic knowledge and episodic memory [7]. The neural network models focusing on the retrieval capability of episodic memory have been designed [8]–[10]. These models represent relations between events, but they have difficulty in describing complicated orders of time sequential events.

EM-ART is a recent neural network model for representing intricate sequences of events [11]. It is based on fusion ART, which has multi-channel inputs in fuzzy ART, to encode each event from input data [12]. The activation values of recognized events are decayed and buffered in the event field by following the time sequence. This is the main difference between this model and the other episodic memory models. The model encodes an episode using a set of temporal events, and thus it can easily retrieve all sequential events by a readout process.

EM-ART can retrieve a correct episode from sparse events or some noisy inputs. In particular, the closer input data are to whole sequential events, the more robust this model is to noisy inputs such as unrelated events. However, if input data are too sparse to predict a correct episode, the model will be affected by the unbalance caused by the number of all committed events constituting an episode. If the model encodes two episodes with different numbers of sequential events, the activation value of an event that belongs to the episode with a smaller number of elements (events) is higher than that from an event that belongs to the other episode with a larger number of elements. This causes a retrieval error when EM-ART receives sequential inputs in real time.

To make the ART model with a more precise retrieval capability from partial input data, we propose Reward-based EM-ART (REM-ART). REM-ART has a multi-channel architecture not only in the first layer but also in the second layer, unlike other ART models. The proposed model has a reward channel in the second layer, which generates a reward signal to compensate recognition errors. The magnitude of the reward signal is proportional to the number of input events under the corresponding episode. REM-ART is learned using both unsupervised learning for encoding events and episodes and reinforcement learning for the reward signal. The sequential inputs to demonstrate the effectiveness of the proposed REM-ART are obtained through a RGB-D camera. For the demonstration, four scenarios are provided and the sequences of events are learned using REM-ART. The experimental results show stable and robust retrieval performance to noisy inputs and incomplete cues.

This paper is organized as follows. Section 2 introduces conventional fusion ART and EM-ART models. Section 3 proposes REM-ART with a detailed description. Section 4 presents experiment results along with a discussion. Finally, concluding remarks follow in Section 5.

2. Episodic Memory Learning

When a human performs a specific task, his or her episodic memory plays an important role in carrying out the task, referring to the memory that stores the sequential events for the task [3], [14]. A robot can also learn and memorize a series of events efficiently using an ART neural network model, which helps the robot recall time sequential events. In this section, fusion ART and EM-ART models are briefly reviewed, and how the robot can learn episodes by these models is described.

2.1 Event Learning

Fusion ART is an extended model of fuzzy ART. This model can receive several types of inputs at a time, since it has a multi-channel in the first layer. The architecture is useful to classify an event, since the event normally consists of several inputs, such as objects and related actions. Therefore, fusion ART is used for the basic structure to learn needed events. Fig. 1 shows the fusion ART architecture.

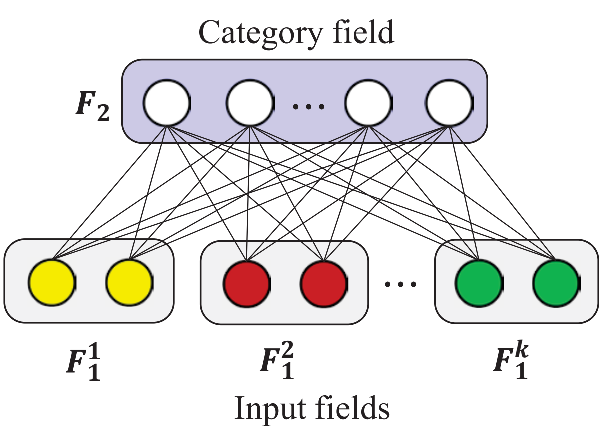


Fig. 1: Fusion ART architecture.

A summary of the fusion ART procedure is given below.

1) Complement coding: Each input field F_1^k receives an input vector $\mathbf{I}^k = (I_1^k, I_2^k, \ldots, I_n^k)$, where $I_i^k \in [0,1]$ indicates the input i to channel k, for $k = 1, \ldots, n$. The input vector $\mathbf{I}^k$ is converted to the activity vector $\mathbf{x}^k$ by a concatenation of the input vector $\mathbf{I}^k$ and its complements $\bar{\mathbf{I}}^k = 1 - \mathbf{I}^k$.

2) Code activation: The F_2 has one channel, which is represented as $\mathbf{y} = [y_1, y_2, \cdots, y_m]$, where m is the number of nodes. The jth output node in the category field is activated by the choice function:

$$T_j = \sum_{k=1}^{n} \gamma^k \frac{\left|\mathbf{x}^k \wedge \mathbf{w}_j^k\right|}{\alpha^k + \left|\mathbf{w}_j^k\right|} \quad (1)$$

where γ^k is a contribution parameter, $\mathbf{w}_j^k$ is a weight vector associated with the category j, α^k is a choice parameter, the fuzzy AND operator $\wedge$ is defined by $(\mathbf{p} \wedge \mathbf{q})_i \equiv min(p_i, q_i)$ and the norm $|.|$ is defined by $|\mathbf{p}| \equiv \sum_i p_i$ for vectors $\mathbf{p}$ and $\mathbf{q}$.

3) Code competition: A code competition is a selection process to distinguish the largest activation value in F_2 layer. The largest activation value is indexed at J as follows:

$$T_J = max\left\{T_j : \text{for all } F_2 \text{ node } j\right\}. \quad (2)$$

Once the maximum activation value is found among all F_2 nodes, this output value is set to one, and all the other values are set to zero. This is the winner-take-all strategy.

4) Template matching: For each channel k, the template matching process checks the resonance between each channel and all F_2 nodes. The resonance is defined as the similarity between the activity vector $\mathbf{x}^k$ and the weight vector $\mathbf{w}_J^k$ associated with the selected execution node, which is calculated using the following function:

$$m_J^k = \frac{\left|\mathbf{x}^k \wedge \mathbf{w}_J^k\right|}{\left|\mathbf{x}^k\right|} \geq \rho^k \quad (3)$$

where ρ^k is a vigilance parameter that is a constraint for the network resonance. If there are no matched nodes in F_2, an uncommitted node is added in F_2 as a new category node.

5) Template learning: Once the resonance occurs, the weight vector w_J^k for each channel k is updated by the following learning rule:

$$\mathbf{w}_J^{k(new)} = (1-\beta^k)\mathbf{w}_J^{k(old)} + \beta^k(\mathbf{x}^k \wedge \mathbf{w}_J^{k(old)}) \quad (4)$$

where β^k is the learning rate. If the weight vector is initialized as $J_{1,2n}$, which is the one row of all-ones matrix, it can be arranged by the following procedure:

$$\begin{aligned}
\mathbf{w}_J^{k(1)} &= (1-\beta^k)\mathbf{w}_J^{k(0)} + \beta^k(\mathbf{x}^k \wedge \mathbf{w}_J^{k(0)}) \\
&= (1-\beta^k)J_{1,2n} + \beta^k\mathbf{x}^k \\
\mathbf{w}_J^{k(2)} &= (1-\beta^k)\mathbf{w}_J^{k(1)} + \beta^k(\mathbf{x}^k \wedge \mathbf{w}_J^{k(1)}) \\
&= (1-\beta^k)^2 J_{1,2n} + (1-\beta^k)\beta^k\mathbf{x}^k + \beta^k\mathbf{x}^k \\
&= (1-\beta^k)^2 J_{1,2n} + (1-(1-\beta^k)^2)\mathbf{x}^k \\
&\vdots \\
\mathbf{w}_J^{k(n+1)} &= (1-\beta^k)^n J_{1,2n} + (1-(1-\beta^k)^n)\mathbf{x}^k
\end{aligned} \quad (5)$$

$$\lim_{n\to\infty} \mathbf{w}_J^{k(n)} = \mathbf{x}^k \quad (6)$$

Eq. (6) means the learned weight vector $\mathbf{w}_J^k$ converges to $\mathbf{x}^k$. It is applied to readout inputs from the category field such that $\mathbf{x}^{k(new)} = \mathbf{w}_J^k$.

2.2 Episode Learning

To recognize and learn episodes from continuous events, the feature of the time sequential representation of events is crucial. EM-ART is the neural network model for qualifying this character to fusion ART [11]. Fig. 2 shows a generalized EM-ART model for episodic memory learning.

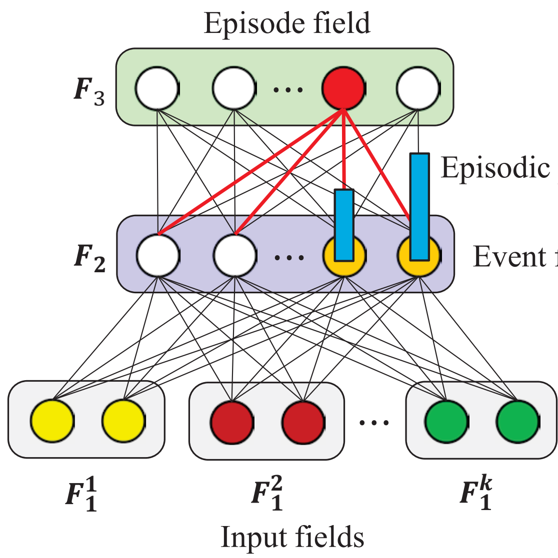


Fig. 2: EM-ART architecture.

As shown in Fig. 2, an episodic pattern is introduced to represent a temporal sequence of events. The encoding process of an episode by using a sequence of events as an input is the same as the encoding process of an event from an input vector, except that episodic encoding uses a time decaying factor to make a sequential input vector. In event field F_2, the activation values of recognized events are successively multiplied by the time decaying factor to keep track of the sequential order. If a set $\mathbf{y}$ represents the output vector of the event field, the first activation value of $\mathbf{y}$ is set to one. The index of the activated node is memorized until the next activation occurs in the event field. When a new activation occurs, y_j is decayed by multiplying the decaying factor using the following equation:

$$y_j^{new} = y_j^{old}(1-\tau) \tag{7}$$

where τ is the decaying factor. It makes an event encoded earlier than the other events have a smaller activation value to represent time decay. After finishing the matching process of event field nodes, non-activated nodes are set to zero. The output vector $\mathbf{y}$ containing temporal information is then the new input for the next additional layer.

2.3 Episode Retrieval

Once EM-ART is learned, it can retrieve episodes and time sequential events of each episode from time sequential inputs. Complete or incomplete cues can be the input vector for EM-ART. When input fields get cues composed of partial events, the template matching process checks the resonance in F_2. After recognizing events, the template matching between F_2 and F_3 is processed in the same manner to retrieve the matched episode. Every time sequential inputs are entered into input fields, EM-ART predicts an episode matched with inputs. This allows the model to retrieve a target episode from complete or incomplete cues in real time.

After recognizing the episode, EM-ART should be able to readout all sequential events matched with the episode. First, the activated node in F_3 readouts the input vector in F_2 using complement coding and learned weight vector. In the retrieved $\mathbf{y}$ vector, the highest value is selected and this value readouts the input vector in F_1. Successively, the value is set to zero and the next highest value is selected as the next input vector. After this process, EM-ART can readout all time sequential events of the selected episode.

2.4 The Limitation of Episode Retrieval

The episode is made of time sequential events, and each episode can have a different number of events for its elements. This implies that the contribution of each event belonging to the different episodes for the activation is different. In particular, the activation value from an event belonging to the episode with a smaller number of events is larger than that from an event belonging to the episode with a larger number of events. It can be shown from the choice function in F_2:

$$T_j = \frac{|\mathbf{y} \wedge \mathbf{w}_j|}{|\mathbf{w}_j|} \tag{8}$$

where k is omitted because F_2 is a single channel. In this equation, the choice parameter α is set to zero and the contribution parameter γ is set to one. As the learning iteration is increased, the learned weight vector $\mathbf{w}_j$ is almost the same as the input vector $\mathbf{y}_j$. Therefore, the summation of all elements in the weight vector $\mathbf{w}_j$ is computed as follows:

$$\begin{aligned} |\mathbf{w}_j| &\simeq |\mathbf{y}_j| \\ &= \sum_{n=0}^{h_j-1}(1-\tau)^n + \sum_{n=0}^{h_j-1}(1-(1-\tau)^n) + (N-h_j) \\ &= \sum_{n=0}^{h_j-1}(1) + (N-h_j) \\ &= N \end{aligned} \tag{9}$$

where h_j is the number of events corresponding to the jth episode, and N is the number of all learned events. This equation shows $|\mathbf{w}_j|$ is always constant at N. To compare the activation values calculated from an event, consider that the event field gets only one event corresponding with the jth episode for the input. In this case, the choice functions of the ith and jth nodes are respectively as follows:

$$\begin{aligned} T_i &= \frac{|\mathbf{y} \wedge \mathbf{y}_i|}{N} \\ &= (\sum_{n=0}^{h_i-1} (1-(1-\tau)^n) + (N-h_i-1))/N \\ T_j &= \frac{|\mathbf{y} \wedge \mathbf{y}_j|}{N} \\ &= (\sum_{n=0}^{h_j-1} (1-(1-\tau)^n) + 2(1-\tau)^p + (N-h_j-1))/N \end{aligned} \tag{10}$$

where $p \in [0, h_j - 1]$ denotes the order of a selected event, and $h_j > h_i$. Therefore, the difference between them is as follows:

$$\begin{aligned} T_j - T_i &= \frac{|\mathbf{y} \wedge \mathbf{y}_j|}{N} - \frac{|\mathbf{y} \wedge \mathbf{y}_i|}{N} \\ &= (2(1-\tau)^p - \sum_{n=h_i}^{h_j-1} (1-\tau)^n)/N. \end{aligned} \tag{11}$$

The difference can be a negative value under the condition $h_j > h_i$. This means the result of episode retrieval may be the ith episode, although the input event belongs to the jth episode as well. This demonstrates that the number of events in each episode affects the activation value. Therefore, the noisy input vector having mixed events may lead to retrieval of a wrong episode, which is critical in reproducing a complete pattern of sequential events.

3. Proposed REM-ART

EM-ART performs well when the input vector in F_1 is exactly matched with time sequential events, or for little noisy inputs. However, this model does not easily preserve robustness against noisy inputs. Moreover, if the number of time sequential events in each episode is different, EM-ART may not retrieve a target episode from the noisy inputs. To overcome this problem, we propose Reward-based EM-ART (REM-ART) consisting of input, category, episode fields. The category field is composed of event and reward channels. The role of the reward channel is to generate a reward that makes the network more robust to noisy inputs.

3.1 Architecture of REM-ART

REM-ART network has multi-channels in F_2 as well as in F_1. Fig. 3 shows the architecture of the model. Reward channel F_2^r in Fig. 3 receives the information about recognized events from event field F_2^e to generate the reward signal. This architecture is similar to the model having two connected fusion ART in the first and the second layers. However, the REM-ART structure in F_2 is different from the conventional fusion ART in that each channel is connected unidirectionally from one to the other channel.

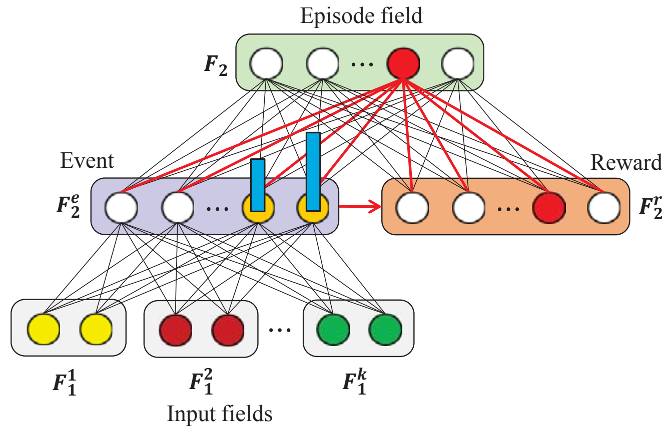


Fig. 3: REM-ART architecture.

3.2 Episode Learning

To encode events, REM-ART uses the basic fusion ART between input fields F_1 and event channel F_2^e. This is the same as the encoding process of EM-ART. After perceiving events, the event channel makes time sequential inputs for episode field, and simultaneously the episode field informs the reward channel of the number of sequential events of the encoded episode. The reward channel then makes the following reward values:

$$\begin{aligned} \mathbf{y}^r &= \{y_1^r, y_2^r, \cdots, y_f^r, \cdots, y_F^r\} \\ &= \left\{\frac{n_1}{h}, \frac{n_2}{h}, \cdots, \frac{n_f}{h}, \cdots, \frac{n_F}{h}\right\} \end{aligned} \tag{12}$$

where $\mathbf{y}^r$ is a reward vector, F is the number of learned episodes, n_f is the number of events of the fth episode among all input events, and $h = \sum_{f=1}^{F} n_f$. In the network learning process, the reward signal is always one because $n_f = h$. The other reward signals are zero because all $n_i = 0$, when $i \neq f$. The reward vector $\mathbf{y}^r$ and the time sequential input vector $\mathbf{y}^e$ constitute multi-channel inputs for encoding each episode. After the learning process, the learned reward weight vector asymptotically converges to

$$\mathbf{w}^{r(n+1)} = (1-\beta^r)^n J_{1,2n} + (1-(1-\beta^r)^n)\mathbf{y}^r \simeq \mathbf{y}^r \tag{13}$$

3.3 Episode Retrieval

Learned REM-ART memorizes the weight vectors for the corresponding episodes. It can be used to retrieve all time sequential events from the observed partial cues. When REM-ART recognizes an episode, only the weight vector is required to recall time sequential events. This can be done simply by substituting $\mathbf{y}^e$ into $\mathbf{w}^e$, such as $\mathbf{y}^e = \mathbf{w}^e$. The reward weight vector $\mathbf{w}^r$ is independent of $\mathbf{w}^e$, and thus it is not needed when retrieving events. The event node with the highest activation value retrieves the input vector in F_1. The activation value of the retrieved node then becomes zero, and the next highest activation node retrieves the next sequential input recursively.

When REM-ART tries to predict an episode from incomplete cues, it recognizes events in F_2^e first. Recognized events in F_2^e are forwarded to reward channel in real time. At the same time, REM-ART loads all learned events and gives them to the reward channel. Then the reward channel generates the reward signals using both sets of data. Since h is the number of recognized events, it increases as a new input comes to input field F_1. To calculate n_f, the reward channel matches recognized events with learned events. For instance, input events matched with the fth episode increase the value of n_f. After the matching process, y_f^r can be calculated using Eq. (12), and the multi-channel input vector of $F_2 = \{F_2^e, F_2^r\}$ is used to calculate activation values for each episode. y_f^r is closer to one when events corresponding the fth episode enter the network. This generates a larger activation value for the fth episode, since the reward value y_f^r reinforces the activation of the fth episode node. This makes the REM-ART network more robust against noisy inputs.

To compare the activation values with EM-ART in the case when the input vector $\mathbf{y}$ in F_2 has only one event for the jth episode, the choice functions of the ith and the jth episodes in REM-ART are respectively as follows:

$$\begin{aligned}
{}^{r}T_i &= \sum_{k=1}^{2} \gamma^k \frac{\left|\mathbf{y}^k \wedge \mathbf{w}_i^k\right|}{\alpha + \left|\mathbf{w}_i^k\right|} \\
&= \gamma^1 \frac{\left|\mathbf{y}^1 \wedge \mathbf{w}_i^1\right|}{N_1} + \gamma^2 \frac{\left|\mathbf{y}^2 \wedge \mathbf{w}_i^2\right|}{N_2} \\
&= \gamma^1 T_i + \gamma^2 \frac{N_2 - 2}{N_2} \\
{}^{r}T_j &= \sum_{k=1}^{2} \gamma^k \frac{\left|\mathbf{y}^k \wedge \mathbf{w}_j^k\right|}{\alpha + \left|\mathbf{w}_j^k\right|} \\
&= \gamma^1 \frac{\left|\mathbf{y}^1 \wedge \mathbf{w}_j^1\right|}{N_1} + \gamma^2 \frac{\left|\mathbf{y}^2 \wedge \mathbf{w}_j^2\right|}{N_2} \\
&= \gamma^1 T_j + \gamma^2 \\
\therefore {}^{r}T_j - {}^{r}T_i &= \gamma^1 (T_j - T_i) + \gamma^2 \frac{2}{N_2}
\end{aligned} \tag{14}$$

where T_i and T_j are the activation values of the ith and the jth episodes, respectively, in EM-ART. This shows the difference between T_i and T_j is decreased by γ^1, and the second term related with the reward signal is added to make the difference positive. Therefore, the reward signal weakens the activation of the episode not concerned with the target episode, while it reinforces the activation of the target episode.

REM-ART encodes and learns episodes in an unsupervised manner, and also it utilizes reinforcement learning for retrieval of the episodes. The reinforcement learning helps the network robustly retrieve the target episode exactly using the reward signal when it receives incomplete or noisy inputs.

4. Experiments

4.1 Experimental Setup

The proposed method was applied to predict which episode would be executed during a visual demonstration captured by a RGB-D camera, mounted on the robotic head of Mybot-KSR, developed in the RIT Lab, at KAIST. The robot is shown in Fig. 4. The Mybot-KSR recognizes the environment through the RGB-D camera. By assuming that objects were on the table, regions of the object hypothesis were detected by removing the plane that is detected using Random Sample Consensus (RANSAC) [13]. The feature of each object hypothesis was extracted by SIFT [15], [16]. The action on objects is related to a hand posture. Thus, the region of one hand near a certain object was found by a skin color detector and it was followed by feature extraction using SIFT. To enhance the accuracy of the action recognition, a speed term, given by the hand's distance between two consecutive frames, was concatenated to the SIFT feature.

The feature extractions of objects and an action were followed by general SVM for classification. The objects in this experiment were $\{$*wateringpot, flowerpot, bottle, cup, toy, box, bread, toaster, dish*$\}$ and the actions were $\{$*grasp, move, tilt, put down, push down*$\}$. Using these recognition algorithms, a single image in video frames could be parsed into objects and their corresponding action, which is defined as an event. The task consists of a sequence of events. To evaluate how well the performed task is predicted, we defined four scenarios: 1) Water the flower; 2) Pour the contents of a bottle; 3) Sort the toys; and 4) Toast a slice of bread. Lists of time sequential events in individual episodes are provided in Table 1.

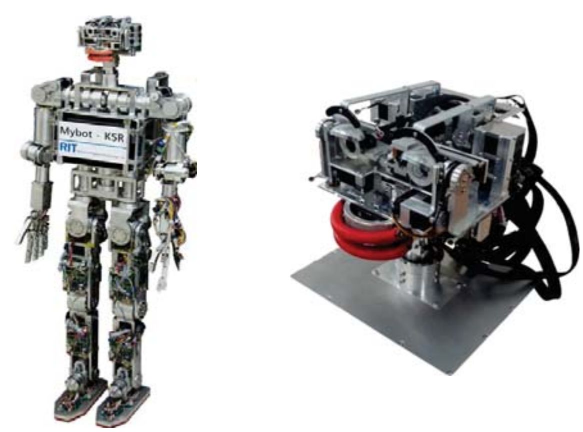


Fig. 4: Mybot-KSR and its robotic head.

The event used in these scenarios needed two input categories, the object and the action. Therefore, we used REM-ART architecture having two channels in the input

Table 1: The lists of scenarios and events.

Scenario	Event
Water the flower	Grasp a watering pot. Move the watering pot to the flowers. Tilt the watering pot to the flowers. Put down the watering pot.
Pour the contents of a bottle	Grasp a bottle. Move the bottle to the cup. Tilt the bottle to the cup. Put down the bottle.
Sort the toys	Grasp a toy. Move the toy to the box. Put down the toy in the box.
Toast a slice of bread	Grasp a bread on the dish. Move a bread to the knob of the toaster. Put down the bread in the toaster. Grasp the toaster. Push down the toaster.

field to store the information on objects and actions. The user acted the sequence of events in all scenarios in front of the humanoid robot. The robot then observed and perceived objects and actions. These sequential inputs are learned by REM-ART. After the episode learning, the user acted each scenario ten times again, and the robot predicted which episode was performed. For comparing the proposed REM-ART with EM-ART for the retrieval error rate, EM-ART was also learned using the same sequential inputs.

4.2 Experimental Results

Fig. 5 shows the recognized image from the RGB-D camera of the robotic head. Each row of Fig. 5 is the perceived sequential events when the user performed each scenario. The features extracted from RGB-D images were made of objects and actions, and they were used to learn and test the neural networks.

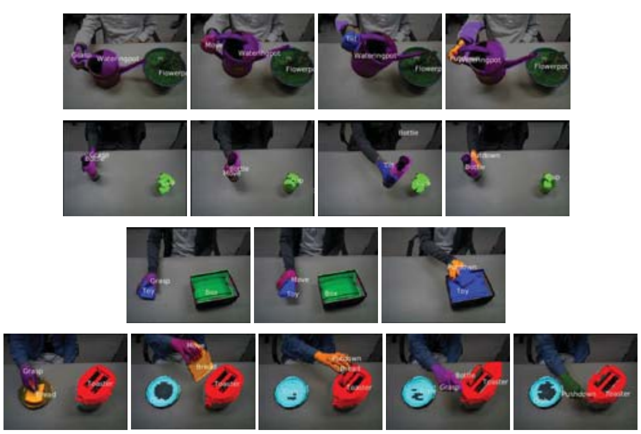


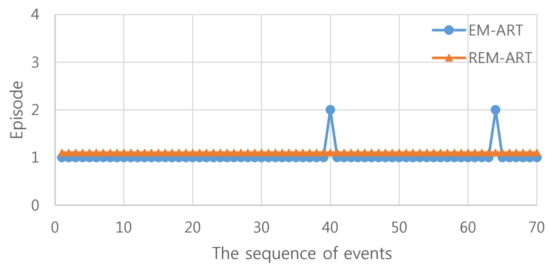
Fig. 5: The sequence of recognized images from *Water the flower* (top row), *Pour the contents of a bottle* (second row), *Sort the toys* (third row), and *Toast a slice of bread* (bottom row).

The prediction results of REM-ART and EM-ART are shown in Fig. 6. Since the sequential inputs are recognized from camera images, they contain many recognition errors. For instance, the robot can extract sequential inputs such as "Grasp a bottle", "Move a chair", and "Throw a pen" from the images. If the user acts scenario 1, these inputs are all noisy inputs. As shown in Fig. 6, EM-ART predicted wrong episodes as the sequence of events is provided to the inputs, because the noisy cues made episode prediction errors. However, REM-ART could predict the exact scenarios from these noisy inputs. The reward signal reinforced the activation of the desired episode node, and at the same time it attenuated the activation of the other episode nodes. This helps REM-ART to recognize the correct episode even though the input vector was disturbed from noises. The important point is that REM-ART becomes increasingly robust as the input sequence enters the input fields, because the reward signal of the desired episode is more and more strengthened whenever relevant sequential events enter. As shown in Fig. 6(b), the prediction result of REM-ART converged to scenario 2 immediately and it was not changed, but EM-ART predicted wrong scenarios after finding the correct scenario, because of error inputs.

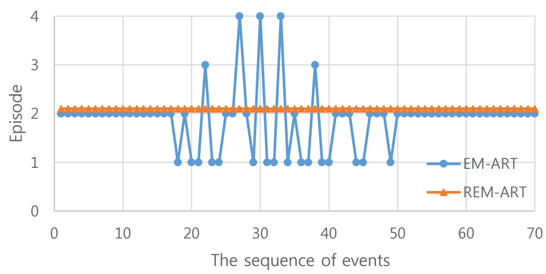
To analyze and compare the prediction accuracy of each model, the confusion matrices are provided in Fig. 7. The prediction accuracy of EM-ART is distributed from 52% to 97%. In particular, the prediction accuracy of scenario 4 is quite low, since this scenario has the largest number of events among all scenarios. As mentioned above, the events relevant to the target episode with a large number of events generate relatively small activation values for the episode. This leads to the other episode nodes being activated, and thus the prediction accuracy is reduced. Therefore, with the EM-ART it is difficult to predict the correct episode from all noisy inputs when the input vector is quite distorted in the real environment. On the other hand, the prediction accuracy of REM-ART is almost 100% for the four scenarios. This demonstrates that the REM-ART model is remarkably effective in predicting and retrieving the target episode from noisy inputs.

5. Conclusion

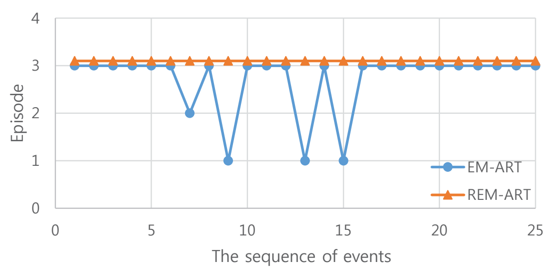
We proposed a novel Reward-based ART model called REM-ART. This model can store and retrieve episodes consisting of time sequential events. The proposed REM-ART could easily increase its memory size to learn events and episodes dynamically. The proposed REM-ART model has multi-channels for input and category fields, and thereby grants the neural network new functionalities. A newly added channel to a category field was designed to make the episodic memory learning model robust to noisy inputs. This channel is called the reward channel, which generates a reward signal. This reinforces the episode prediction accuracy of REM-ART, such that the robot can perceive contextual situations more robustly in the real environment. The experimental



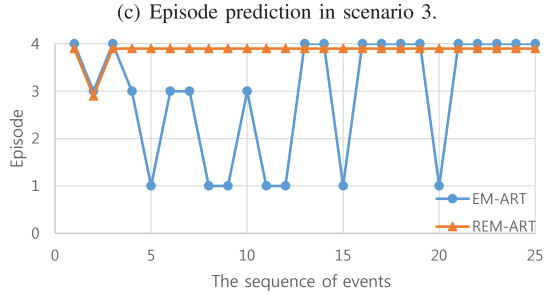
(a) Episode prediction in scenario 1.



(b) Episode prediction in scenario 2.



(c) Episode prediction in scenario 3.



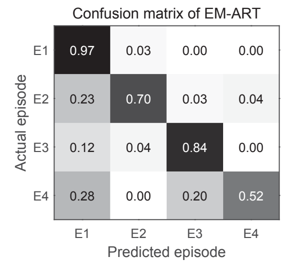
(d) Episode prediction in scenario 4.

Fig. 6: Episode prediction results from sequential input events of each scenario.

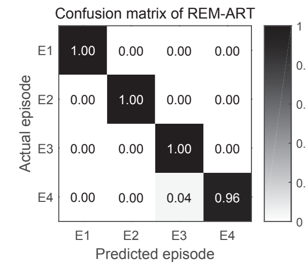
results demonstrated that the reward channel effectively reduces the prediction error of REM-ART regardless of noisy inputs.

Acknowledgement

This work was supported by the Technology Innovation Program, 10045252, funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea).



(a)



(b)

Fig. 7: Confusion matrices of the episode prediction using (a) EM-ART and (b) REM-ART.

References

[1] E. Tulving, *"Episodic and semantic memory 1,"* in *Organization of Memory*. New York: ACADEMIC PRESS, 1972.
[2] E. Tulving, *Elements of episodic memory*. New York: Oxford Univ. Press, 1983.
[3] A. Nuxoll and J. E. Laird, *"Extending cognitive architecture with episodic memory,"* in *Proc. 22nd Nat. Conf. Artif. Intell.*, 2007, pp. 1560-1564.
[4] A. Nuxoll and J. E. Laird, *"A cognitive model of episodic memory integrated with a general cognitive architecture,"* in *International Conference on Cognitive Modeling*, 2004.
[5] Matthew L. Lee and Anind K. Dey., *"Providing good memory cues for people with episodic memory impairment,"* in *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility*, ACM, 2007, pp. 131-138.
[6] W. C. Ho, K. Dautenhahn, and C. L. Nehaniv, *"Comparing different control architectures for autobiographic agents in static virtual environments,"* in *Proc. IVA*, 2003, pp. 182-191.
[7] S. T. Mueller and R. M. Shiffrin, *"REM-II: A model of the developmental co-evolution of episodic memory and semantic knowledge,"* in *Proc. Int. Conf. Develop. Learn.*, vol. 5. 2006, pp. 1-7.
[8] M. W. Howard and J. Lim, *"A distributed representation of temporal context,"* in *Journal of Mathematical Psychology*, vol. 46. 2001, pp. 269-299.
[9] A. B. Samsonovvich and G. A. Ascoli, *"A simple neural network model of the hippocampus suggesting its pathfinding role in episodic memory retrieval,"* in *Learn. Memory*, vol. 12, pp. 193-208, Mar. 2005.
[10] S. Becker and J. Lim, *"A computational model of prefrontal control in free recall: strategic memory use in the california verbal learning task,"* in *J. Cogn. Neurosci.*, vol. 15. 2005, pp. 821-832.
[11] W. Wang, B. Subagdja, A.-H. Tan, and J. A. Starzyk, *"Neural modeling of episodic memory: Encoding, retrieval, and forgetting,"* in *IEEE Trans. on Neural Netw. and Learning Syst.*, vol. 23, 2012.
[12] G. Carpenter, S. Grossberg, and D. Rosen, *"Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system,"* in *Neural Netw.*, vol. 4. 1991, pp. 759-771.
[13] M. A. Fischler and R. C. Bolles, *"Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,"* in *Commun. ACM*, vol. 24, no. 6, 1981, pp. 381-395.
[14] B.-S. Yoo, Y.-H. Yoo, W.-R. Ko, S.-J. Lee, S.-H. Cho and J.-H. Kim, *"Realization of task intelligence based on the iOA for assistive robots,"* in *Proc. International Conference on Artificial Intelligence (ICAI)*, Las Vegas, U.S.A., accepted, Jul, 2015.
[15] D. G. Lowe, *"Distinctive image features from scale-invariant keypoints,"* in *Int. J. Comput. Vision*, vol. 60, no. 2, 2004, pp. 91-110.
[16] Y.-H. Yoo and J.-H. Kim, *"Procedural memory learning from demonstration for task performance,"* in *Proc. IEEE. Sys. Man. Cybern*, Oct. 2015.

Linguistic Approach to Scientometrics

M. Charnine[1], N. Somin[1], S. Klimenko[2], and V. Ezhela[3]
[1]Institute of Informatics Problems, Federal Research Center,
Russian Academy of Sciences, Moscow, Russia
[2]Institute of Computing for Physics and Technology, Protvino, Russia
[3]Institute for High Energy Physics, Protvino, Russia

Abstract - *Different approaches for measuring the quality of scientific articles are considered. Difficulties and problems associated with using Science Citation Index (SCI) as well as various methods to improve it are discussed. Optimal methods for collective expertise are proposed. The advantages of the linguistic approach to the problem are discussed. A specific method for estimating the information content of scientific texts based of the linguistic approach is presented. The method can measure "idea-based" influence and how ideas propagate between articles. Idea is presented very precisely as a set of phrases with similar meaning. Phrases representing the ideas in the texts of the articles form together a system of implicit references/links. Implicit link is a reference to the author or his ideas in the text. Bibliometric measures which in addition to bibliographic references also take into account the implicit links can more accurately measure scientific impact and the quality of the articles.*

Keywords: Scientometrics, Citation counts, Bibliometric measures, Bibliographic references, Implicit links, Big Data.

1 Introduction. The problem of measuring the quality of scientific articles

The problem of measuring the quality of scientific articles has always been very important for scientific community. This problem became even more important recently. A huge flow of scientific information requires the development of automated methods and tools for solving this problem.

Scientometrics is the study of measuring and analysing science, technology and innovation. Bibliometrics and scientometrics are two closely related approaches to measuring scientific publications and science in general, respectively. The measuring of the quality of scientific publications is one of the main problems of bibliometrics.

A long time ago specialists in bibliometrics have understood that assessment of the quality of the content of individual articles is very complex procedure so it can be done only by colleagues and experts in a given subject area. Nevertheless scientometrics has achieved some success in measuring the quality.

2 Background

In 1960, Eugene Garfield introduced the Science Citation Index (SCI) [1], which was the first formal measure of the quality of scientific paper. The main indicator of SCI is the number of bibliographic citations/references to the paper.

Later, other measures were introduced to evaluate the scientific activity of the authors. Particularly, in 2005 Jorge E. Hirsch suggested an index to quantify an individual's scientific research output, called Hirsch index or H-index [13].

A variety of methods have previously been proposed for analyzing bibliographic references that form together citation graph. Measures of importance such as PageRank (Brin and Page, 1998) [17] can be derived recursively from the citation graph. Other graph based methods have been introduced such as node ranking according to authorities and hubs (Kleinberg, 1999) [16], and link prediction (Xu et al., 2005) [15]. Such graph based measures do not in general make use of the textual content of the articles.

From the analysis of the above methods we can see that scientific paper is considered to have higher index and quality if it has more impact on subsequent works. Also we can see that the importance of a scientific work has previously been measured mainly through metrics derived from citation counts.

The above methods are based on the assumption that bibliographic reference/citation is the only mechanism used in the scientific community for giving credit to prior work. But there are other methods that take into account the possibility to give a credit to prior work by mentioning the author or his idea in the text of the article.

These methods, which take into account more aspects of scientific influence/impact, have better accuracy. For example, Dietz et al. (2007) [11] has included both text and links into a probabilistic model to infer topical influences of citations.

Also J.Foulds (2013) [14] has analyzed citation graph taking into account the content of the articles. J.Foulds sought to identify the most influential scientific articles. In particular, he sought to measure “idea-based” influence and how ideas propagate between articles. J.Foulds emphasized that topical influence measures a specific dimension of scientific importance, namely the tendency of an article to influence the ideas of citing articles. According to J.Foulds, “ideas” are represented as mixes of topics or topic distributions. The topic is represented as a discrete distribution over words in the text of the article.

In our approach we use more precise way to represent ideas as sets of phrases with similar meaning [19]. Phrases representing the ideas in the texts of the articles form together a system (hypertext) of implicit references/links. Each implicit link is a reference to the author or his ideas in the text of the article that can be realized as hyperlink. Our method can measure very precisely “idea-based” influence and how ideas propagate between articles.

Of course, bibliographic references to other papers (that are specified in the reference lists or footnotes) are quite universal and informative indicators of their scientific significance. In addition to measuring the quality of scientific papers, bibliographic references can help to solve many other problems, such as automatic classification of texts, searching of documents relevant to some topic [3], and measuring the impact that papers have on each other.

3 Disadvantages of citation counts

However, the there are some disadvantages of using only bibliographic references and citations counts for measuring the quality of scientific papers. First of all, many citations are made out of “politeness, policy or piety” (Ziman, 1968) [18].

Second, all references are different, because they may come from sources with different reputation. To solve this problem, the journals in which articles are published, are divided into classes of significance, forming different “citation databases” or citation indices, for example, Web of science, Scopus, and so on. However, this method is not always successful, because it is highly dependent on the subjective factor, and many valuable papers are published in journals not belonging to these citation indices.

Also, methods based on counts of bibliographic references have significant disadvantages due to the problem of dependence of citations. This problem has several aspects. First of all, because of the different kind of plagiarism, counterfeiting, and other information schemes the references to a particular scientific paper may be unnecessarily duplicated, increasing SCI index. At the same time, there are possible cases of plagiarism and borrowing of ideas without citation, which underestimates the true significance of not cited scientific work. Another aspect is related to dependences in citations due to specific structure of scientific community [2]. Life experience shows that it is impossible to consider different scientists as completely independent. There are scientific teams with superior-subordinate relationships; there are hierarchical relationships such as professor-undergraduate, teachers-students, etc. As a result citations became dependent. Finally, there are some informal networks of scientists taking any paradigm of scientific research, or vice versa, reject it, so that references become dependent. If some authors have cited some paper, then it is very likely that another author will cite it too. Of course, such dependences in citation can distort the ranking of scientific works.

4 Methods for collective decision making

Science Citation Index (SCI) can be considered as a collective measure of the quality of articles performed by scientific community. That’s why SCI as a measure of the quality of scientific papers can be improved by using methods for collective decision making. These methods are quite relevant in this situation because authors by making references actually perform their work like experts that make decision about significance/importance of cited scientific paper. All together the authors writing articles on a particular topic perform a collective work of creating list of recommended bibliographic references on the subject. These methods for collective decision making can be used for the optimal combination/integration of individual decisions of the authors.

The method of S. Nitzan and J. Paroush [4] is one of the most promising methods of collective decision-making. The optimal aggregation rules for combining experts/authors decisions proposed by Nitzan [5], involve an iterative process of gradual refinements of experts ratings and includes the following steps.

- Select some initial weights (shares) for decisions of each expert/author;
- For each expert, using the history of his decisions, calculate the probability of convergence of his individual decisions with the weighted collective decision;
- Calculate new weights (shares) of each expert based on the probability calculated in the previous step;
- Repeat three previous steps until the new weights converge with those calculated in the previous iteration.

The authors of this algorithm proved that it works better than any other judgment aggregating rule; in particular, it is better than collective decision making based on a simple majority rule. The above algorithm uses linear combination of expert decisions. The main result of this work was the claim that if the probability of a correct decision of each expert is known, then a linear aggregation rule of their decisions is optimal, and the maximum probability of correct collective decision is reached when the weights (shares) of experts are calculated by the formula:

$$\mathbf{Wi = log (Pi / 1\text{-}Pi)} \quad (1)$$

where Pi - is the probability of a correct decision of the expert with number i.

Thus, Nitzan’s algorithm combines the individual experts’ decisions into optimal collective decision. In accordance with this algorithm it is possible to make better collective decisions about inclusion of particular bibliographic references into collection of recommended links of particular topic.

It should be noted that Nitzan’s method works well only for independent experts. This necessary condition is not strictly fulfilled in the case of mass evaluation of scientific work. Furthermore, this method is assumed that all the experts evaluate all alternatives; and that condition is also not fulfilled in the case of bibliographic references: each author cites only a small part of other authors. So there are some problems associated with direct application of Nitzan’ method for improving of SCI index based on bibliographic references. Described below special linguistic methods can help us to overcome these problems.

5 Linguistic methods for measuring scientific impact

Special linguistic methods of text analysis can help us to greatly improve the above described methods and also can enable the development of new innovative methods. Linguistic methods help us to exploit textual information in conjunction with citation information. Using this extra information, we should be able to gain a better understanding of scientific impact than simple citation counts. Linguistic methods help us to identify how ideas propagate between articles. These methods allow us to analyze not only the impact on bibliographic references (citation impact), but also take into account the impact on the texts of other articles.

Also the great advantage of linguistic methods is that they allow us to automate the search and collection of scientific information from different sources including the Internet that dramatically expands the volume of the indexed scientific texts.

Methods of linguistic analysis can be implemented in three levels.

Lexical level. At this level, a text stream is parsed into tokens: words, punctuation and other elements of the text, followed by the definition of the type of each lexical element. This is the initial level of analysis. At this level it is possible to solve many important problems. First of all, it's an automatic recognition of different elements of bibliographic descriptions such as: author, title of the source (magazine), month and year of publication, and several others. This makes possible to organize automatic updates of citation databases from the Internet. This also will boost the completeness of these databases and indices. In addition, this opens the prospects for automatic growth of databases for indexing authors and journals.

Morphological and syntactic level. It is the next (after the lexical level) step of analysis where all the tokens get morphological characteristics, and restore the syntactic relationships between words. This level already allows us to realize the idea of implicit links.

Semantic level. Implicit links are the terms and phrases that express certain scientific ideas or indicate their authors. If they occur in different scientific papers, we can assume that there are links from the later to the earlier paper. Implicit links can help us to achieve the following features.

Firstly, in addition to bibliographic references/links, implicit links dramatically increase the density of the network of links between publications. Thus, the measure of the quality of scientific papers becomes more objective when we take into account implicit links too. As a result, Nitzan's method should provide more accurate results when we use implicit links in addition to bibliographic references.

Second, implicit links often presented as brief formulation of scientific ideas. This raises the possibility of identifying continuity of ideas, prioritizing their wording, plagiarism detection, and so on. In particular, it should be noted that implicit links provide an opportunity to analyze the structure of scientific information not only from citation graph, but based on the content of the papers, that is extremely important for conducting research. Indexing using implicit links can help us to discover and eliminate the effect from poorly motivated bibliographic references that are not relevant to the subject of the work.

Morphological and syntactic level is sufficient for detecting similarity (up to morphological paradigm) of phrases that represent similar implicit links. However, in more complex cases, the same idea can be represented by different sentences that are not identical lexically or grammatically. In this case, it is necessary to use more deep semantic level of analysis.

One of the basic ideas of modern approach to semantics is the so-called distributional hypothesis of meaning, expressing the fact that “words (or short phrases) that occur in the same contexts tend to have similar meanings” [8,13]. This hypothesis helps us to find a set of different phrases with the same meaning. We call such phrases as semantic equivalents. We can use synonym replacements, grammatical transformations and translation programs to find semantic equivalents for a given phrase.

Thus, on the semantic level it is possible to detect fragments of texts with the same meaning and use them as implicit links. As a result, one semantic reference can be represented by many different semantically equivalent phrases. This greatly expands the possibilities of using implicit links.

6 Discovering implicit links

By definition, an implicit link is a term or phrase that expresses certain scientific idea or indicates its author. If such terms or phrases occur in different scientific papers, we can assume that there are links from the later to the earlier paper. Implicit links occur in the texts of scientific papers and may be not confirmed by formal bibliographic references.

It is possible to discover implicit links by using method of machine learning from examples. As an example, let’s consider a set of scientific papers that have the same bibliographic reference. These papers form a training set for special category of texts, and machine learning algorithm can automatically classify a new text – does it belong to this category or not.

In particular, let’s consider some scientific papers from physical database that have bibliographic references to the article of "B. Aubert et al. (BABAR Collaboration), Nucl. Instrum. Methods Phys. Res., Sect. A 479, 1 (2002)". There are frequent references (implicit links) to this paper in the texts of other scientific papers in the following forms.

- The BABAR detector is described elsewhere [...].
- In this paper a data sample of ..., collected by the BABAR detector [...] at the PEP-II asymmetric energy storage ring, is analyzed.
- We analyze a data sample corresponding to an integrated luminosity of ... recorded with the BABAR detector [...] at the PEP-II asymmetric-energy storage rings.

These examples show that implicit links are designed in various ways, but all of them contain specific keyword "BABAR detector". We can use the following formula (2) to find such specific keywords in the texts of scientific papers.

Weight (Keyword) = Weight (Category),

Weight(Keyword) = Log(P(Keyword & Category) / (P(Keyword) P(Category))) (2),

here Weight (Keyword) is a measure of association between Keyword and Category;

P (Keyword & Category) is a probability that an arbitrary text from Category contains the specified Keyword,

P(Keyword) is a probability of encountering this Keyword in any text,

and P(Category) is a probability that an arbitrary text belongs to the specified Category.

Explanation of the formula (2) was given by P.Turney [9], when he described the algorithm for searching synonyms online. Equation (2) calculates the measure of association between keyword and category of texts.

In a simplest case this measure of association between keyword and category can be calculated as the frequency of occurrence this keyword in the texts of category, divided by the frequency of keyword in general, in all analyzed texts. This method is similar to the well-known measure TF/IDF [10], which takes into account not only the frequency of the keyword in the documents of category, but also the frequency of all documents containing keyword.

Automatic identification of relevant keywords for some category and calculation of their weights allow us by using contextual search to find new articles that are relevant to this category. If such article contains the keyword "BABAR detector", which is found in all texts of training sample, then we can assume that an implicit link is found. The final decision whether or not this keyword represents an implicit link can be granted to the experts. It is also possible to use methods of crowdsourcing and collective decision-making to identify implicit links.

The above described method for calculation of significant terms and their weight coefficients can be used not only for identification of implicit links, but also for automatic classification of new documents, as well for the discovering of new knowledge and evaluation of information content from Internet.

7 An example of using implicit links to measure the informativeness of texts

As an example of using implicit links we propose the following method for measuring informativeness of documents/texts. This method has passed approbation in real projects [6]. By definition, the informativeness is a measure of the quality of information content of the document. In this example the informativeness of scientific document is computed as a linear combination of the number of bibliographic references and implicit links pointing to this document. This example presents ranking approach to automatic estimation of informativeness of phrases and documents. The scope of presented method is an evaluation of information content of Internet documents in a particular subject area. Therefore, this example includes the method of description of the subject area.

Initially, the subject area is described by a set of user-defined keywords that are used for generating search queries to the Internet. The algorithm for generating search query is iterating through all combinations of keywords (single keywords, pairs of keywords, triplets, etc.) so that the query length does not exceed a certain threshold N. The threshold N is determined empirically so that search results contain several hundred of relevant documents in average.

Then the search queries are processed using search engines (Google, Yandex and other), and as a result a set M1 of text documents is retrieved. The set M1 can be used as a source of URL-links, and as a result the extended set of documents M2 is formed. A set M2 also contains URL-links, and this way the process of expanding the set of documents M2 can be continued by several iterations of additions of linked documents. It is necessary to check the newly found documents for the presence of the original keywords. After subsequent iterations the set of required documents M2 can be significantly increased.

Next, the set of documents M2 is divided into text fragments (or phrases) similar in length to the conventional sentences. As a result we have a database with records in the form

< text fragment > - <URL, indicating this fragment>

The size of this database can be up to several terabytes. To store and search in the database it is not enough traditional software tools working with databases, but it is necessary to use special hardware and software technologies named «Big Data».

By using the obtained database the statistical calculations are made: how many different independent URLs refer to the same text fragment and its semantic equivalents (paraphrases and translations). We calculate semantic equivalents using automatic translator and linguistic processor [12].

Note that there is a problem of independence of web pages/documents. The fact is that for advertising purposes there are often created doubles of pages, which content almost completely duplicates the content of the main web page. We should delete such doubles from statistical calculations. We can identify possible doubles of web pages and web sites by using our accumulated database, because it actually stores all the important content from all web sites.

Thus, each text fragment gets a rating equal to the number of occurrences of its semantic equivalents in the independent documents/ web pages. Evaluation of the informativity of the phrase/fragment is performed by calculating the index of occurrences of its semantic equivalents in other independent documents of subject area. The informative phrase (that first appeared in a specific document) represents a number of implicit links pointing to this document. The rank of scientific document is computed as a linear combination of the number of bibliographic references and implicit links pointing to this document. This rank is considered as a measure of the quality of information content of this document. We call this measure as iformativeness of this document.

The proposed methods for estimating the information content of documents and phrases (text fragments), open the access to boundless and growing body of knowledge and valuable information from the Internet. These methods allow us to automate the search and download professional scientific information from the Internet that dramatically expands the volume of indexed scientific texts.

It should be noted that the proposed approach can be used not only to measure the quality of scientific papers, but also to rank web sites. This approach has been tested on several subject areas both scientific and applied [7].

8 Acknowledgements

This work is supported by Russian Foundation for Basic Research, grants 13-07-00272, 13-07-00958, 14-07-31288, 14-07-00362, 15-07-06586.

9 Conclusions

Using science citation index as a measure of the quality of scientific articles is not sufficiently accurate measure. In this paper, we propose another more advanced measure of the quality of scientific papers. In addition to bibliographic references it takes into account implicit links that actually are informal references to the authors and their ideas in the content of the articles.

The proposed measure also takes into account temporal relationships between articles and independence of the referring authors. Proposed methods allow us to automate the search and download of professional scientific information from the Internet that greatly expands the database of indexed scientific texts.

In our approach we use precise way to represent ideas as sets of phrases with similar meaning. Phrases representing the ideas in the texts of the articles form together a system (hypertext) of implicit references/links. Each implicit link is an informal reference to the author or his ideas in the text of the article. Implicit links can be realized as hyperlinks in the hypertext. Our method can measure very precisely “idea-based” influence and how ideas propagate between articles.

Bibliometric measures which in addition to bibliographic references also take into account the implicit links can more accurately measure scientific impact and the quality of the scientific articles.

10 References

[1] Garfie1d E., Sher I.H., Torpic R.J. The Use of Citation Data in Writing the History of Science. – Philadelphia: Institute of Scientific Information, 1964. – 76 p.

[2] T. Kuhn, 1962. The Structure of Scientific Revolutions, Chicago: University of Chicago Press.

[3] Ezhela, Bunak, S. Lugovsky V. Lugovsky, K. Lugovsky "Citing as means of finding additional knowledge and automatic indexing," Third All-Russian Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections", 2001, Petrozavodsk.

[4] Nitzan, S., and J. Paroush. 1982. “Optimal Decision Rules in Uncertain Dichotomous Choice Situations.” International Economic Review 23(2): 289-297.

[5] E. Baharad, J. Goldberger, M Koppel, S. Nitzan, “Beyond Condorcet: Optimal aggregation rules using voting records”, 2010, Working Papers, Bar-Ilan University, Department of Economics, No. 2010-20.

[6] M. Charnine, N. Somin, V. Nikolaev, “Conceptual text generation based on key phrases”, Proceedings of the 2014 International Conference on Artificial Intelligence (ICAI 2014), WORLDCOMP’14, July 21-24, 2014, Las Vegas, Nevada, USA, CSREA Press, pp.639-643.

[7] I. Galina, M. Charnine, N. Somin, V. Nikolaev, Yu. Morozova, O. Zolotarev, “Metod for generating subject area associative portraits: different examples”, Proceedings of the 2015 International Conference on Artificial Intelligence (ICAI 2015), WORLDCOMP’15, July 27-30, 2015. Las Vegas Nevada, USA (in press).

[8] Peter D. Turney, Patrick Pantel, “From Frequency to Meaning: Vector Space Models of Semantics”, J. Artif. Intell. Res. (JAIR) 37: 141-188 (2010).

[9] Turney, P.D. (2001), Mining the Web for synonyms: PMI-IR versus LSA on TOEFL, Proceedings of the Twelfth European Conference on Machine Learning (ECML-2001), Freiburg, Germany, pp. 491-502.

[10] Salton, G. and McGill, M. J. 1983 Introduction to modern information retrieval. McGraw-Hill, ISBN 0-07-054484-0.

[11] L. Dietz, S. Bickel, and T. Scheffer. 2007. Unsupervised prediction of citation influences. In Proceedings of the 24th International Conference on Machine Learning, pp. 233–240.

[12] I. Kuznetsov, E. Kozerenko, M. Charnine. “Technological peculiarity of knowledge extraction for logical-analytical systems”. Proceedings of ICAI’12, WORLDCOMP’12, 2012, Las Vegas, USA, CRSEA Press.

[13] Hirsch, J. E. 2005. An index to quantify an individual's scientific research output. Proceedings of the National Academy of Sciences of the United States of America, 102, 16569–16572.

[14] Foulds, J. R. and Smyth, P. Modeling scientific impact with topical influence regression. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 113–123, USA, 2013.

[15] Xu, Z., Tresp, V., Yu, K., Yu, S., & Kriegel, H.-P. (2005). Dirichlet enhanced relational learning. ICML ’05: Proceedings of the 22nd international conference on Machine learning (pp. 1004–1011). New York, NY, USA: ACM Press.

[16] Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. Journal of the ACM, 46, 604–632.

[17] Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. Proc. of the Seventh International Conference on World Wide Web, Vol. 7, pp. 107-117.

[18] John M. Ziman. 1968. Public Knowledge: An Essay Concerning the Social Dimensions of Science. Cambridge University Press, Cambridge, UK.

[19] M. Charnine, V. Ezhela, S. Klimenko, A. Raikov. “The search for alternative approaches to the construction quality assessment of scientific publications”, Proceedings of CPT2014 The International Scientific Conference, May 11-18, 2014, Larnaca, Republic of Cyprus, Pub. ICPT, Moscow-Protvino, 2015, ISBN 978-5-88835-035-5, pp.53-61.

Comparative semantic study of Russian and English constructions with light verbs

Julia Ignatova, Irina M. Kobozeva
Department of Theoretical and Applied Linguistics, Moscow Lomonosov State University, Moscow, Russia

Abstract — *In the paper we explore semantic properties of typical Russian and English Verb-Object constructions with light verbs (LV). In section 1 our approach to the problem is outlined. In section 2 the criteria for delimiting the constructions under study are formulated. In section 3 two semantic constraints on the choice of a LV for a given «key» noun are proposed (one based on aspectual features, and the other — on argument structure). Section 4 focuses on LV-construction of the type* have a V *(*swim, nap *etc.) and its Russian partial equivalent. In section 5 semantic properties of LV-construction of the type* give a V *(*laugh, grunt *etc.) are described and two existing ways of translating it into Russian are compared.*

Keywords: Constructions, Light Verbs, Lexical Function $Oper_1$, Actionality, Argument Structure, Translational Equivalents.

1 Introduction

On the cline of grammaticalization there exists an intermediate stage between content verbs and auxiliaries, so called *light verbs*. The term was introduced in [1: 117] to refer to verbs which, though they possess a full semantic meaning in other contexts, can be used in combination with some other element, typically a noun or a prepositional phrase, where their contribution to the meaning of the whole construction is reduced in some way (cf. the content verb *have* in *He had a car* and the correlative light verb in *He had a nap.*). A lot of papers were dedicated to this phenomenon (see e.g., [2], [3], [4], [5]) but some problems it poses still remain unresolved.

Although light verbs were primarily defined in semantic terms Western generative linguists mainly studied their syntactic behavior. In Russian linguistics attention to such verbs was drawn by the proponents of Meaning — Text Theory (MTT). In MTT such semantically weak verbs were studied from the perspective of lexical restrictions on their combinability with content nouns. Such restrictions were captured with the help of lexico-semantic rules called «lexical functions» (LFs). E. g., LF $Oper_1$ (from Latin *operare*) takes some «key» noun with event meaning (e. g. *a promise*) as its argument and gives as its value a semantically «empty» verb (here *give*), which is used to form a V + NP_{acc} construction with this noun (here *to give a promise*), that is semantically equivalent to the corresponding content verb (here *to promise*). This stable syntagmatic relation is represented in our case as $Oper_1$ (*a promise*) = *to give* (see the definition and inventory of LFs in [6: 32–54])[1]. A «key» N in such constructions is usually (but not necessarily) an action or result nominalization of some verb and such a semantic relation is represented in MTT as a LF S_0, e. g. S_0.(*negotiate*) = *negotiations*; S_{res}.(*to promise*) = *a promise.* So we can formulate the following meaning equation $Oper_1(S_{0/res}(V))$ = V. Such «empty» verbs, represented as values of LFs in MTT are called lexical-functional verbs and it is obvious, that they may be subsumed under the notion of light verbs (LVs). At first collocations with LVs were treated in MTT as a bundle of word combinations that have to be listed in the lexical entry of the corresponding «key» noun but later some semantically based regularities, or semantic constraints on the choice of LV for different kinds of «key» nouns were discussed in [7], [8], [9] among others. In this paper we move in the same direction, trying to elucidate and to compare semantic constraints on constructions with LVs in Russian and English. For our analysis we will take Russian LVs *delat'* (‘do / make’), *davat'* (‘*give*’), *vesti* (‘conduct / lead / guide’) in Russian and their typical counterparts in English — *make, give, have*. We will take only such constructions in which these verbs appear as values of the LF $Oper_1$ defined above[2]. Our first step will be to provide a clear delimitation of such constructions. After that we will give the typical examples of such constructions in both languages and finally we will make an interlinguistic comparison of the selected examples.

2 The full characterization of the constructions under study

The constructions we are going to compare possess the following set of properties:

1) they have the syntactic structure V + NP_{acc}, that is they consist of a verb with its direct object NP;
2) the verb belongs to the category of LVs: it is semantically ‘bleached’, i. e. it does not denote any particular kind of situation on its own, its meaning is functional (aspectual or modal), while its second argument, or «key» noun possesses event semantics and solely contributes lexical meaning to the whole

[1] Unlike mathematical function, LF may give more than 1 value for its argument, e. g. $Oper_1$(*a conversation*) = *to have*, *to hold.*
[2] That means that such verbs project the direct object NP, that denotes an event, and the subject NP that denotes the 1st semantic argument of the event denoted by t direct object NP.

predication[3]. Therefore, we eliminate 2 kinds of V + NP_{acc} constructions:
a) the ones with nouns that do not denote events as in e. g. *delat' stanki (to make machines), vesti poezd (to conduct a train), davat' kuklu (to give a doll), to make a dress, to give a present, to have a car*;
b) the ones with event nouns, in which the V does contribute lexical meaning to the construction as a whole as in *vesti soveschanie* ('to conduct a meeting'), where the noun *soveschanie* 'a meeting' denotes a particular kind of event whose 1st semantic argument is a group of all people that participate in it, but the verb *vesti* ('conduct') is not lexically «empty», because the construction as a whole does not mean any participation in a meeting but only the activity of the chair person.

3 The comparison of English and Russian light-verb constructions

In this section we expose common and differential features of selected LV-constructions in both languages and draw some interlinguistic conclusions.

Let us take the following pairs of semantically equivalent Russian and English collocations:

(1)	*a. delat' zajavlenie*	*make a statement,*
	b. delat' oshibku	*make a mistake,*
	c. delat' pryzhok	*make a jump,*
(2)	*a. davat' sovet*	*give advice*
	b. davat' signal	*give a signal*
	c. okazyvat' podderzhku	*give support*
(3)	*a. vesti peregovory*	*carry on negotiations*
	b. vesti besedu / razgovor	*have a conversation*
	c. *vesti perepisku*	*hold correspondence*

In (1)–(3) respective nouns are translational equivalents. In (1) respective light verbs are derived from content verbs (later referred to as source verbs) with the same direct meaning 'do, make'. For the sake of brevity in such cases we shall simply say that LVs have the same "source meaning". In (2a, b) we have the same situation: Russian and English LVs have source meaning 'give', but in (2c) English LV has the same source meaning 'give', while its Russian counterpart may be said to have no source meaning at all, because *okazyvat'* is used now only as an "empty" LV, while etymologically it is a cognate of *pokazyvat'* ('to show'). In (3) one and the same Russian LV *vesti* with the source meaning 'to lead, to conduct (smb)' correspond to three LVs each with its own source meaning that is different from 'lead, conduct (smb)'.

[3] Most of such nouns are derived from a verb which is synonymous to the whole construction, e. g. *make an* ***announcement*** *= to* ***announce;*** *delat'* ***predlozhenie*** (lit. 'make a suggestion') = ***predlagat'*** ('to suggest'), but some nouns are synchronically just cognates to an adjective or a participle, that forms the content element of a synonymous predicate, e. g. *keep* ***silence*** *= be* ***silent***, *make a* ***mistake*** *= be* ***mistaken***.

First and foremost, we should answer the following question: are there any semantic reasons for choosing this or that LV for a given argument noun to form a collocation of $Oper_1$? The cases of one-to-one semantic correspondence between two languages as in (1) and (2a,b) are numerous enough to be taken as evidence that there might be some connection between the meaning of the «key» N and the source meaning of its LV. But the discrepancies in (2) and (3) clearly show that the "key" N does not predict the specific direct meaning of its LV. Still if we turn from specific lexical meanings in their entirety to abstract semantic features of "key" Ns and their LV partners we can observe some semantic regularities.

Without special effort we notice that argument Ns in (1) are derived from Vs that belong to the category of Achievements in Z. Vendler's terminology, that has aspectual semantic features [–durative, –telic], while argument Ns in (3) are derived from Vs that belong to Vendler's Activities ([+durative, –telic]). Now a certain pattern can be observed: event arguments that represent Achievements can combine with verbs *delat' = make.* Russian content verb *delat' / sdelat'* and its English equivalent *make* belongs to Vendler's Accomplishments, so N of Achievement can combine with LV derived from Accomplishment V. Examples (2a,b) show that N of Achievement (*advice*, *a signal* and their Russian pairs) can also combine with LV derived from Achievement V (*give — davat'*). What they cannot do is to combine with LV derived from Activity V (*carry on*) or State (*hold, have*), both having the feature [+durative, –telic] as shown in (4):

(4) a. *carry on /*hold /*have a mistake —
*vesti /*provodit' oshibku
b. *carry on /*hold /*have a jump —
*vesti /*provodit' pryzhok
c. *carry on /*hold /*have a statement —
*vesti /*provodit' zajavlenije
d. *carry on /*hold /*have advice —
*vesti /*provodit' sov'et[4]

Thus we can formulate a generalization (4):

(5) If event N in $Oper_1$ collocation belongs to Achievement type, then its LV cannot be derived from V of either Activity or State type

So actionality of V from which LV is derived is one of the factors that constraint the choice of possible LV for a given event N.

Another factor is the argument structure, i.e. the number and semantic roles of arguments (semantic actants) of the event N in the construction. As can be seen from the examples in (1) and (2) some of such Ns combine with the verb *delat' — make*, others — with *davat' — give*. Such a distribution is by no means

[4] *Vesti /provodit' sovet* is OK in Russian but only when *sovet* means 'council' (Activity type N), not 'advice' (Achievement type N) and the collocation as a whole means 'to hold a council' and not 'to give advice', thus supporting our generalization.

arbitrary. Situations that are denoted by Ns in (2) co-occuring with the LVs *davat' — give* always imply a participant with a role of the Recipient that is usually positively or neutrally affected by the Agent's action, i. d. Recipient-Benefactive. When we are giving advice or support or even a signal the person who is receiving it is the one who needs it for her own purposes or at least in order to act adequately in a given situation. One of the arguments of the source V *davat' — give* in its direct meaning is a Recipient and in the situation of 'giving' it is also implied that this Recipient gets something what he/she needs or at least what he / she could use in order to fulfill someone else's needs. In a sense, argument structure of the verb *give* (*davat'*) «agrees» with that of the event N in having a second major animate argument with the Recipient-Benefactive role. No event N in (1) has a Recipient in its argument structure. So *give* (*davat'*) is incompatible with such Ns. The same is true of the event Ns that have a Recipient-Malefactive argument (cf. *a threat (ugroza) or a curse (prokl'atie)*) as it is shown in (6):

(6) **give smb. a threat / a curse —*
**davat' komu-l. ugrozu / prokl'atie*

So we propose the following semantic constraint:

(7) Verb *to give* (*davat'*) can occur as LV in Oper_1 collocation only if an event N has an argument with a role of Recepient and this is not Recepient-Malefactive

These observations lead us to the conclusion that the choice of a certain verb to form a light-verb constructions is not arbitrary and complies with a certain set of semantic rules and the fact that such phenomenon appears in both languages makes this conclusion more powerful.

4. Periphrastic construction *to have a V* and the problem of its translation into Russian

Among the described English constructions there is one that is worth being discussed in more detail — a periphrastic construction *have a V* type, where *V* stands for a nominal constituent that is homonymous to a verb [2]. This constructions has the same set of properties as other light-verb constructions with one exception: a verb from which an event argument is derived is not fully synonymous to the whole construction. Unlike other light-verb constructions we discuss this ones ascribes a specific semantic property to its Agent. As an example, let us compare the sentences (8):

(8) a. *Yesterday after work I had a drink with my colleague at Dicey's.*
b. *Yesterday after work I drank with my colleague at Dicey's.*

(8a) can be interpreted as 'Yesterday after work I was drinking with my colleague at Dicey's and I was enjoying it' (vs. (8b) in which the second conjunct of the interpretation is absent). Moreover, there is an aspectual difference between such constructions and simple source verbs: a periphrastic construction implies that the action goes on for a limited and rather short period of time plus this action is considered to be repeatable. Therefore, such constructions serve as semantic modifiers of the action denoted by the corresponding verb. They transform it into "agentive, experience-oriented, antidurative, atelic and reiterative". As there is no parallel construction in Russian it is of interest to look at its translational equivalents in parallel sub-corpus of Russian National Corpus (RNC):

(9) a. *I'll see you when you come back. We'll* ***have a walk*** *and talk together.*
b. *Мы увидимся, когда вы вернетесь. Тогда* ***погуляем*** *и поговорим.*

(10) a. *There's a pool there. We could* ***have a swim*** *together.*
b. *Там есть бассейн, и мы могли бы вместе* ***поплавать****.*

(11) a. *Strange mushrooms —* ***have a smell****!*
b. *Странные грибы —* ***понюхай****!*

(12) a. *So can we* ***have a dance****?*
b. *Так мы можем* ***потанцевать****?*

(13) a. *John* ***had a smoke*** *of tobacco.*
b. *Джон* ***покурил*** *табака.*

(14) a. *John let Jimmy* ***have a throw*** *of his new boomerang.*
b. *Джон разрешил Джимми* ***побросать*** *его новый бумеранг.*

(15) a. ***Поплачем*** *о грехах человеческих.*
b. *Let's* ***have a cry*** *over the sins of mankind.*

As can be seen from (9)–(14) and many other examples from RNC periphrastic constructions are regularly translated with verbs with delimitative prefix *po-*, that presents the action denoted by the root verb as atelic and limited in time. If we now go back to the Wierzbicka's interpretation of the given periphrastic construction we will see that they also present actions as limited in time and atelic. As for Agent's specific semantic component, Russian agentive verbs with delimitative *po-* do not by themselves express the idea that Agent enjoys the action as examples like (16) show:

(16) *Я уже достаточно* ***по****ползал под пулями.*
I already enough po-(DEL)crawl under bullets.
I have had enough of crawling under bullets.

In Russian translation equivalents of the English examples above the 'Agent's enjoyment component' may appear only as a probable entailment based on knowledge of the world or on relevant information from context.

5. Construction *to give a V* and its Russian equivalents

The construction to *give a V* was mentioned among other periphrastic constructions with a nominal constituent homonymous to a verb but not discussed in [2]. Let us have a look at the examples (17)–(20) containing typical instances of the construction (*give a sigh, give a laugh*, *give a cry, give a groan*).

(14) *There was a momentary silence during which she could imagine him rallying for another attack. For a moment she thought he was going to slam the phone down, then she heard him* ***give a*** *deep* ***sigh****. 'I'm sorry — I over-reacted'.*

(15) *'You're a handsome devil — you know that, Billy?' She lifts her eyebrows at me and then stands up and goes over to the sink. I* ***give*** *a* ***laugh****.*

(16) *At the very height of the paroxysm, he made a movement with his knee which caused him to* ***give a*** *great* ***cry*** *which she, lost in the abandon of the moment, construed as passion…*

(17) *We wait to be given the bad news. 'It's fitness today!' We all* ***give*** *a* ***groan****.*

We notice that all the passages above describe situations in which the number of participants is more than one. There is an Agent and an Observer. Agent through paralinguistic behavior (laughter, scream, groan or sigh) kind of "transmits" information about their feelings (probably unconsciously). Thus the role of the Observer can be reanalyzed as that of the Recipient (of information). This makes the verb *give* perfect candidate for the LV of the construction (cf. the constraint (7) above). It is unlikely to meet this construction in a context where there is only one participant (Agent) present in the episode of a narration In such cases corresponding simple verbs (*sigh, cry, laugh, groan*) are used which are indifferent to presence or absence of the Observer. Among the first 25 instances of *give a V* construction from British National Corpus only 3 examples referred to the action of an Agent being alone[5] vs. 22 with the Observer mentioned in the context). We may say that in cases, when a nominal *V* is derived from an agentive verb having only one argument, LV *give* is licensed only if all three arguments it inherits from the source verb are somehow instantiated: Agent and Patient at the surface level and Recipient at the discourse semantic level. And again, we find aspectual difference between such constructions and simple verbs: *give a V-construction* describes semelfactive actions. Therefore, as could be expected, the majority of these constructions is translated into Russian by means of the semelfactive affix *-nu-*: *give a sigh — vzdokh**nu**t', give a laugh — usmekh**nu**t's'a, give a cry — krik**nu**t', give a grunt — burk**nu**t', give a groan — okh**nu**t'*. However, we can also find the corresponding Oper$_1$ constructions in the parallel sub-corpus of RNC:

(18) a. *And when each person had got his (or her) cup of tea, each person shoved back his (or her) stool so as to be able to lean against the wall and* ***gave*** *a long* ***sigh*** *of contentment.*
b. *Получив свою чашку, каждый отодвинул от стола табурет, чтобы прислониться спиной к стене, и* ***испустил*** *глубокий* ***вздох*** *удовлетворения.*

(19) a. *Suddenly Legolas* ***gave*** *a* ***cry****. 'The horses! The horses!'*
b. *Неожиданно Леголас* ***издал крик****: — Лошади! Наши лошади!* (lit. *The horses! Our horses!*)

(20) a. *Teabing* ***gave*** *an unnatural* ***laugh****. "I'm afraid that is one show of faith I cannot afford".*
b. *Тибинг* ***издал*** *фальшивый* ***смешок****: — Боюсь, что просто не могу себе позволить проявить такую неосмотрительность.*

(21) a. *I sang out, I could not help it now; and* ***giving*** *a sudden* ***grunt*** *of astonishment he began feeling me.*
b. *Это уж было выше моих сил — я взвыл от ужаса, а он* ***издал*** *негромкий* ***возглас*** *изумления и принялся ощупывать меня.*

We can observe the following correspondence: English *give a V* constructions — Russian Oper$_1$ constructions with the LVs *izdat', ispustit'*. If we now take original Russian examples with these LVs from the RNC we will see that constructions with these verbs (just like *give a V* constructions) are used in situations in which the number of the participants is at least two:

(22) a. *Петя однажды нанес визит Главкому ВМС, выразил соболезнование (у него умер отец), на что тот* ***издал*** *легкомысленный* ***смешок****, дабы не огорчать своим несчастьем гостя…*
b. *Peter once paid a visit to the commander of Naval Forces, expressed condolences upon the loss of his father to which the commander* ***gave a*** *careless* ***laugh*** *in order not to upset his guest with the misfortune…*

(23) a. *Увидел нас,* ***издал*** *воинственный* ***возглас****, взмахнул дубинкой над головой…*
b. *He saw us,* ***gave a*** *warlike* ***cry****, waved the war club over his head…)*

(24) a. *Ляо Лан* ***издал*** *громкий* ***вздох*** *восхищения. Дар Ветер, не отрываясь, смотрел на неуклюжий, тяжелый остов древней твари.*
b. *Liao Lang* ***gave a*** *loud* ***sigh*** *of admiration. Wind Gift was staring at the clumsy, heavy body of the ancient creature.*

(25) a. *Розмысл* ***испустил вздох*** *облегчения и велел обшарить с тщанием весь участок.*

[5] A native speaker, a professional teacher of English commented that in these 3 uses of the *give a V construction* so described performance of an Agent can be considered odd.

b. *An army engineer **gave a sigh** of relief and told to search carefully the whole plot.*

(26) a. *На что, кажется, глупа птица-тетеря, а и та, когда своих цыплят по кустам учит кормиться, набрёдши на эту ягоду, **издаёт** предостерегающий **крик**…*

b. *Even though black-cock bird is generally thought to be stupid, yet when it teaches its chicks how to feed off berries and comes across this berry this bird **gives a** warning **cry**…*

The revealed regularity can be taken into consideration: English periphrastic *give a V* constructions with *a V* derived from verbs of sound and paraverbal behavior are more adequately translated into Russian by constructions with the verbs *izdat', ispustit'*.

6. Conclusions

To sum up, we would like to present the following table comparing light-verb constructions of $Oper_1$ variety in English and Russian:

	Prototypical English light-verb constructions	***to have a V*-construction**	***to give a V*-construction (for V of sounding and paraverbal behavior)**
Their functions in English	carry stylistic features	aspectual features (atelic, short duration, et al.) + 'Agent enjoyment' feature	aspectual feature (semelfactive) + imply the presence of the Observer in the context
Russian equivalents	light-verb constructions (=collocations with lexical-functional verbs)	verbs with delimitative prefix *po-*	constrctions with LVs *izdat', ispustit'*
Their functions in Russian	equal to English	the same aspectual features, no Agent-oriented features	equal to English
Interlingual pairs	*make a statement — delat' zajavlenie, give advice — davat' sovet, have a conversation — vesti besedu*, etc.	*have a swim — poplavat', have a chat — pobesedovat', have a kick — popinat'*, etc.	*give a laugh — izdat' smeshok, give a cry — ispustit' krik*, etc.

7. References

[1] Otto Jespersen. "A Modern English Grammar on Historical Principles. Part VI. Morphology". George Allen and Unwin Ltd., 1965.

[2] Anna Wierzbicka. "Why can you Have a Drink when you can't Have an Eat?"; Language Vol. No. 58, 753–799, 1982.

[3] M. Butt. "The Light Verb Jungle". In Harvard Working Papers in Linguistics, ed. G. Aygen, C. Bowern, and C. Quinn. 1–49. Vol. No 9, Papers from the GSAS / Dudley House workshop on light verbs, 2003.

[4] J. Algeo. "British or American English?" In A Handbook of Word and Grammar Patterns, 269–278, 2006.

[5] N. Shomoossi, A. Shomoossi. A review of light verb constructions in English and Persian Complex Predicates. In http://www.actalinguistica.com/journal/index.php/al/ article/download/14/42 (2012).

[6] Igor A. Mel'cuk. "The Rusian Language in the Meaning-Text Theory", Wiener Slawisticher Almanach, Sonderband 39, 1995.

[7] T. Reuter. "O perifrasticheskih naimenovanijah rechevoj dejatel'nosti" ["On Periphrastic Nominations of Speech Activity"]. In Logicheskij analiz jazyka. Jazyk rechevyh dejstvij. Nauka, 1994.

[8] Jurij D. Apresjan. "O semanticheskoj nepustote i motivirovannosti glagol'nyh leksicheskih funkcij" ["On semantic non-emptiness and motivatedness of verbal lexical functions"]; Voprosy jazykoznanija, No 4, 3–18, 2004.

[9] Lidia N. Iordanskaya, Igor A. Mel'čuk "Smysl i sochetaemost' v slovare" ["The Meaning and Compatibility in the Dictionary"]. Jazyki Slavjanskikh Kultur, 2007.

SESSION

LATE POSTER PAPERS

Chair(s)

TBA

Predicting Correctness of "Google Translate"

Yulia Rossikova, J. Jenny Li, and Patricia Morreale
Computer Science, Kean University
1000 Morris Ave, Union NJ 08340 USA
juli@kean.edu

***Abstract*— This paper presents a new modeless approach for Machine Learning predictions, called <u>Radius of Neighbors (RN)</u>. We applied RN to predict the correctness of Google translator and found it to be an improvement over K-Nearest Neighbors (KNN) in terms of prediction accuracy. Both methods are applicable to situations when a mathematical prediction model does not exist or is unknown. With RN, we will be able to create new applications that rely on the users' awareness of translation accuracy, e.g. an online instant messager, which allows users to chat in various natural languages.**

Keywords – K-Nearest Neighbors (KNN), Machine Learning, Prediction.

EXTENDED ABSTRACT

Machine learning in recent years has given us many new breakthrough applications such as self-driving cars, phone contact center voice recognision, effective web search and automatic natural language translators. Prediction is a key feature of machine learning and K-Nearest-Neighbor (KNN) is a well known prediction method that doesn't need any known model in advance, and thus is suitable for the situations when the model doesn't exist or is unknown. However the accuracy of KNN hinders his wider usage in prediction.

Many research found that even though KNN is simple and easy to use, its accuracy is often lacking as compared to the level of model-based methods. KNN calculates the predicted value by taking the average of the K nearest neighbors(usually just a simple mean: divide K into the total values). Consider the situation when some nearest neighbors are far away, the prediction in this case might not be very accurate. For example, if K is 2 and one of them is far away, their average might not be an accurate calculation of the node value under prediction. To improve the accuracy, we propose an innovative approach of Radius of Neighbors (RN) where all nearest neighbors within a certain radius will be taken into consideration for the average.

As an innovative method, RN does not have any related work published so far, though there were/are many researchers working with the widely known KNN [1]. P. Hart proposed the Condensed Nearest Neighbors Rule [2], which reduces a training set in a way that makes a prediction almost 100% accurate. T. Cover devoted to the KNN method in his work "Nearest Neighbors pattern classification" [3]. George Terrel and David Scott were working with KNN as a special case of a variable-bandwidth Kernel density "balloon" estimator [4]. A survey in [5] summarized all the work on using KNN for classification. Please note that our work is different because we use KNN and RN to predict actual values, rather than simple classes to which the predicted value belongs.

After our invention of RN method and its implementation algorithm, we compare KNN and RN methods with respect to accuracy and stability by estimating prediction errors for both. The prediction is an actual numerical value in the range of 0% to 100%, not a simple category value. Our data includes a Training Set of 100 sentences and a testing set of 9 sentences in English. These sentences were translated into three foreign languages Yoruba, Russian and Telugu by their native speakers, who fluently speak both, one of these languages and English. Table 1 shows some sample data we collected for the prediction with an English column, the translated sentences, and incorrect parts in the translation. It also include other factors that we use in prediction, such as search results and frequency grouping of words or sentences.

TABLE I. SAMPLE DATA COLLECTED

#	Sentence	Incorrect part	Correct translation	Google search, results	Frequency group
1	Come here!	.	Иди сюда!	753,000	3.
2	Move away now!	Move away.	Отойди сейчас же!	421,000	2.
3	He is trying to speak.	Speak.	Он пытается говорить.	15,600,000	6.
	...	...	...	...	...
1	Come here!	Come.	ఇక్కడికి రండి	13,700	4.
2	Move away now!	Move.	ఇప్పుడు దూరంగా వెళ్ళిపో	11,700	4.
3	He is trying to speak.	is trying.	అతను మాట్లాడటానికి ప్రయత్నిస్తున్నాడు	4,220	2.
	...	...	...	...	...
1	Come here!	.	Wa sibi	397,000	6.
2	Move away now!	away.	Kuro nbi nsin	839,000	6.
3	He is trying to speak.	Speak.	O fe soro	683,600	6.

We used "Google Translate" for the translation and manually evaluated the correctness of the translation in each language group with a numerical value. The length of each sentence and the length of the part of each sentence that was translated incorrectly were both found and used to calculate a percentage of translation correctness, which we are using as our prediction result. Another value that we predict is the incorrectness of the translation.

As to the factors to be considered that affect the translation correctness, we are also using a frequency of sentence, as well as frequency of words. This attribute factor was evaluated by copying each sentence into Google search and recording the number of Google search results gotten. For the sake of having a uniform distribution, we divided these outcomes into 6 approximately equal groups. For example, for Russian data set these groups include the following: less than 250000, 250001-500000, 500001-1000000, 1000001-4000000, 4000001-15000001 search results. For our testing set of 9 sentences, the values of the original sentence length and the usage frequency of the translated sentences are known, the correctness or incorrectness is unknown – we estimate it using both KNN and RN. We use the two factors to predict the correctness.

We implemented the above-mentioned KNN and RN methods in a Java program using the following algorithm so that the entire process is automated for us to conveniently extend this work to the translation of other natural languages. The algorithm includes 5 steps:

1) Obtain the list of input sentences to be predicted for translation correctness and create a template to hold each sentence's characteristics.
2) Calculate *incorrectness* of each sentence (incorrect part is divided by the total length)
3) Calculate the *distance* from the sentence to the training set and sort sentences in distance ascending order.
4) Estimate *correctness* using KNN and RN.
5) Calculate standard error of the estimations.

The running of the implementation on three languages gave us the following results (Figure 1) with standard errors for the situations of using 2 and 3 factors in the prediction.

2 dimensions:

- Yoruba: error(rn)=0.1752, error(knn)=0.2357 → RN is better
- Russian: error(rn)=0.1360, error(knn)=0.1792 → KNN is better
- Telugu - exception. (Exception is removed by increasing dimensions)

3 dimensions:

- Yoruba: errornew(rn)= 0.26663041925700515 error(knn)= 0.2758224340437895 (RN is better)
- Russian: errornew(rn)= 0.28196114745616635 error(knn)= 0.24578096547417805 (KNN is bet
- Telugu errornew(knn)= 0.20936123631412326 error(rn)= 0.27398600083751357 (KNN is bette

The above data indicates the following results as given in Table II. Yoruba training/testing data sets gave us better prediction results for RN than for KNN (error of 0.175 vs 0.253). Russian has the other way around and Telugu is inconclusive, which needs future study with more samples.

TABLE II. INITIAL RESULTS

Language	Better Method	
	2 dimension	3 dimension
Yoruba	RN	RN
Russian	KNN	KNN
Telugu	inconclusive as we have mixed results	KNN

Table II summarizes the initial result, which shows that the RN method is promising and has potential of being a better method with higher accuracy in modeless prediction. To further investigate this claim, we identified several research topics that deserve further investigation:

1) Lacking of data within the radius. This problem could be caused by the sampling size being not sufficient. For example, Figure 2 shows the data collected for Telugu with one radius not having any neighbors, and thus undecidable for comparison using this language. We are looking to collect more data with our next version having a training set of millions of sentences provided by existing translation sites.

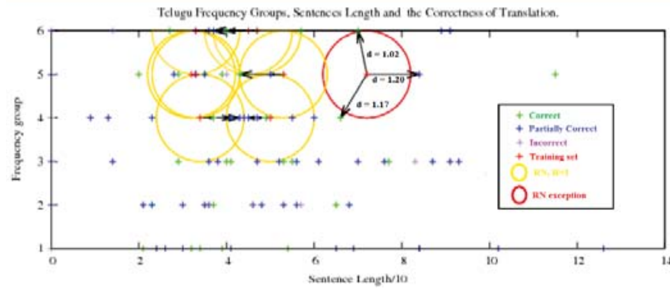


Figure 2. The red radius has no any nearest neighbor

2) Increasing dimension number. The number of factor/characteristics/dimension we used in our initial experiment is only 2 initially. We then increase it to 3 as shown in Figure 3 with an additional dimension of English word frequency. More other factors/dimensions or their combinatorial might affect translation correctness and they are under study.

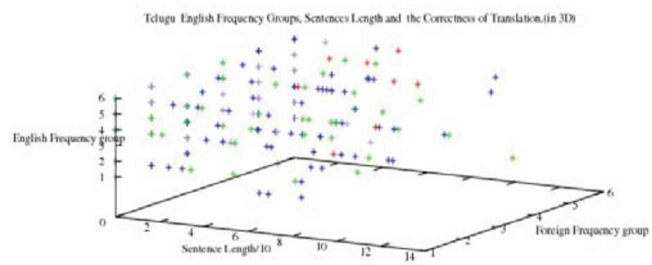


Figure 3. Data with one additional dimension

3) Application of the technology. With the implementation of the KNN and RN algorithms, we were able to predict the correctness of "Google Translate". To make this work useful, we are creating an application for mobile phones users to chat in different natural languages and to be aware of the correctness of the chatting content translation for smooth conversation and improved communciation.

REFERENCES

[1] Nigsch F, Bender A, van Buuren B, Tissen J, Nigsch E, Mitchell JB (2006). "Melting point prediction employing k-nearest neighbor algorithms and genetic parameter optimization". Journal of Chemical Information and Modeling 46 (6): 2412–2422.

[2] P. E. Hart, The Condensed Nearest Neighbor Rule. IEEE Transactions on Information Theory 18 (1968) 515–516.

[3] Cover TM, Hart PE (1967). "Nearest neighbor pattern classification". IEEE Transactions on Information Theory 13 (1): 21–27.

[4] D. G. Terrell; D. W. Scott (1992). "Variable kernel density estimation". Annals of Statistics 20 (3): 1236–1265.

[5] L. Jiang, Z. Cai, D. Wang, S. Jiang, "Survey of Improving K-Nearest-Neighbor for Classification", FSKD (1), 2007.