SESSION

ROBOTICS, PATH FINDING METHODS, NOVEL APPLICATIONS, AND RELATED TOPICS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

A Faster Alternative to Traditional A* Search: Dynamically Weighted BDBOP

John P. Baggs, Matthew Renner and Eman El-Sheikh
Department of Computer Science, University of West Florida, Pensacola, FL, USA

Abstract – *Video games are becoming more computationally complex. This requires video game designers to carefully allocate resources in order to achieve desired performance. This paper describes the development of a new alternative to traditional A* that decreases the time to find a path in three-dimensional environments. We developed a system using Unity 5.1 that allowed us to run multiple search algorithms on different environments. These tests allowed us to find the strengths and weaknesses of the variations of traditional A*. After analyzing these strengths, we defined a new alternative algorithm named Dynamically Weighted BDBOP that performed faster than A* search in our experiments. Using this search, video game developers can focus the limited resources on more complex tasks.*

Keywords: A*, Navigation Mesh, Path-finding, Dynamically Weighted BDBOP, Video Games

1 Introduction

Path-finding algorithms are used in many Computer Science application areas, including robotics and video games. One of the most popular path-finding algorithms used is the A* search. A* is an improvement on Dijkstra's shortest path algorithm. The improvement significantly cuts down the number of nodes that need to be visited during the search. A decrease in the number of nodes correlates directly with a decrease in the space and time complexities of the search. A*'s improvement upon Dijkstra's algorithm comes from the use of a heuristic method to reduce the search space [1].

Today, video games require an enormous amount of computational resources. A majority of these resources are dedicated to graphics rendering and calculations in the physics engine [2]. Given these limited resources, all computational components of video games must be optimized for speed and memory management. Path-finding, a component of non-player character (NPC) intelligence, is a computationally intensive task; therefore it is essential to improve the path-finding computational efficiency. Because of it's search speed, A* is one of the most widely used path-finding algorithms in the video game industry. This research addresses improving the efficiency of path-finding algorithms.

A path-finding algorithm requires the following: a start point, an end point, and a representation of the environment in which the search is to take place. Today, the two main methods environments are represented in video games are with square grids and navigation meshes. Square grids take a specific section of space in the environment and map the space to a matrix location. The matrix has to be large enough to contain the number of squares needed to represent the environment. Navigation meshes were developed in the 1980s for robotic path-finding and most games after the 2000s have adopted these navigation meshes. In [3], a navigation mesh is defined as "…a set of convex polygons that describe the walkable surface of a 3-dimensional environment".

There are several efficiency related issues with square grid environments. The smaller the squares in the grid, the more detailed the description of the environment becomes. However, at the same time, the number of squares increases the search space for the path-finding algorithm. As the search space increases, the time complexity increases. Another issue with square grids is that the squares are laid on top of non-movable or static obstacles. Because the obstacles are represented in the grid, collision detection is required during path-finding. Navigation meshes directly address these two challenges. A navigation mesh represents the environment with polygons that are completely free of obstacles. This condition, and the condition that all polygons are convex, removes the need for collision detection anywhere inside a polygon [4]. There is no restriction on the size that a polygon can be as long as the rules of being convex and obstacle free are maintained. A navigation mesh search space consists of significantly fewer nodes than the square grid search space for the same environment. For these reasons, our research focuses on developing and implementing a navigation mesh for experimentation with path-finding algorithms.

This paper summarizes two main contributions:
1. Results and conclusions from an experiment comparing different path-finding algorithms on a set of three-dimensional environments developed using Unity 5.1. We implemented A* and popular variations of A*. These variants include Bi-Directional A*, Fringe search, optimized versions of traditional A*, Beam search, Dynamically Weighted A*, and Bi-Directional A*. During the experiment, each search was run on each environment 1000 times, collecting data for a comparison of each search algorithm's performance on a given environment.

2. A proposal for a new variant on A*. The proposed algorithm outperforms A* in both time and space complexity. The new algorithm sacrifices A*'s guaranteed optimal solution in order to attain these performance gains. The solutions found by the new algorithm were optimal in some cases, but not all. Later in the paper we argue that the differences between solutions found by our algorithm and the optimal solution are not concerns to a video game developer.

2 Related Work

Common practice in the video game industry for representing 3 dimensional environments, up until the early 2000s, was to use waypoints. Waypoints are a set of manually generated points in the environment that form a path that the agent can walk upon. The time it takes for developers to manually place these waypoints is not feasible for larger maps. Another drawback is that waypoints do a poor job of representing the environment because they only represent single points and the line connecting those points. The restrictive nature of Waypointer led to the use of the more flexible navigation meshes and square grids for representing environments. Xiao Qui discusses the limitation of square grids representing environments [1].

Square grids often require a large number of cells to adequately describe an environment. A well designed navigation mesh that is used to describe the same environment as a square grid will result in far fewer total nodes needed than the square grid will. One reason for this is that the nodes in a square grid cover the entire environment including the non-walkable areas, while a navigation mesh represents the walkable areas in the environment. Another reason for this is that the way a navigation mesh is designed is to create convex polygons as nodes that represent as much of the environment as they can in a given space and that do not include static obstacles. This allows the navigation mesh to use fewer nodes than a square grid to represent the same environment, effectively reducing the search space for a path-finding algorithm, which serves to reduce the search time for the algorithm to find a path.

Various path-finding algorithms can be used across both square grids and navigation meshes to find a valid path between two points. Fringe search is one such algorithm the use of which on square grids was researched in an attempt to determine if it could outperform A*. The Fringe search algorithm is a take on Iterative Deepening A* (IDA*) which uses recursive depth-first search of the environment until either the goal is found or a node that exceeds the threshold is found. If the threshold is exceeded it is increased and the search starts over [5, 6]. Fringe search iterates over the frontier of the search tree like IDA* but uses a data structure to help the search. "The data structure used by Fringe Search can be thought of as two lists: one for the current iteration (now) and one for the next iteration (later)"[5]. The paper concluded that Fringe search was faster in its calculations even though Fringe search expanded and visited more nodes than A*. Our results using an implementation of Fringe search differ from this and are discussed in the Results section.

A valuable resource during our research into different path-finding algorithms was a blog written by Amit Patel [7]. Patel describes multiple variations on the A* search in his blog. We implemented several search algorithms based on these brief descriptions, including Dynamically Weighted A*, Beam Search and Bi-Directional A*. The concise and clear descriptions of the algorithms made the implementations relatively straightforward.

3 Algorithm Design

Dynamically Weighted Bi-Directional Beam OPtimized (BDBOP) search is a variation on A* that we created which uses features of Dynamically Weighted, Bi-Directional, Beam and Optimized A* searches. During our research into variations and methods of optimizing A* we noticed that there are different benefits gained from using each algorithm. After narrowing down our focus to algorithms that specialized in reducing the number of nodes visited in the search we formulated our algorithm design. Additionally, we optimized these searches with respect to the speed at which they navigate their data structures, strictly in terms of their Big O complexities. Dynamically Weighted BDBOP visits far less nodes than traditional A*. This allows Dynamically Weighted BDBOP to outperform A* even without the efficiency optimizations to the data structures.

The defining characteristic of Bi-Directional search is that it conducts two A* searches in parallel or two A* searches in alternating successions. One search starts from the start node and searches for the goal node, while the opposite search begins at the goal node and searches for the start node. The search is complete when the frontiers of the searches meet or one of the two searches finds their destination node. Bi-Directional search visited fewer nodes when used to navigate our environments than A* did.

The next search we evaluated was Dynamically Weighted A*. Dynamically Weighted A* places a high priority on getting the search moving quickly in the general direction of the goal in the beginning and then as the search gets closer to the goal precision becomes more important. This was accomplished by adding a weight to the heuristic. The weight decreases as the search gets closer to the goal. This results in fewer nodes visited than traditional A* but does not guarantee an optimal path.

The Beam Search algorithm improves upon the space complexity of traditional A* by placing a limit on the size of the open list used in A*. One must be very careful when deciding the limit to impose on the open list, because too severe of a limit may lead to an incomplete search.

```
agentToGoal is the search starting from the agent and going to the goal
goalToAgent is the search starting from the goal and going to the agent
ρ is a constant that will divide the number of polygons to get the number
   of times each search will iterate
α is the number of iteration (# of Polygons / ρ)
0.while (pathFound = false)
1. if (α <= 1)
2.    returnValue := agentToGoal's doSearch (# of Polygons *2)
3.    if(returnValue = 1) // goal was found
4.       pathFound := true
5.    else
6.       no path can be found
7. else
8.    returnValue := agentToGoal's doSearch(α)
9.    if (returnValue = 1) // goal was found
10.      pathFound := true
11.   else if (returnValue = 2) // a frontier connection was found
12.      pathFound := true
13.   else if (returnValue = -1) // openList is empty no path is found
14.      no path can be found
15.   else
16.      returnValue := goalToAgent's doSearch(α)
17.      if (returnValue = 1) // agent was found
18.         pathFound := true
19.      else if (returnValue = 2) // a frontier connection was found
20.         pathFound := true
21.      else if (returnValue = -1) // openList is empty no path is found
22.         no path can be found
```

Figure 1. Pseudocode for the coordinator of Dynamically Weighted BDBOP

The main improvement made to the data structures used for A* was to use a min-heap for retrieving the node in the open list with the lowest cost value. Inserting into the open list goes from an O(n) operation to an O(log(n)) operation [8]. The open list was ordered so that a removal would be O(1), but in the min-heap we must maintain the heap order each time we remove the lowest node. This requires an O(log(n)) operation. Another improvement we made was to use a lookup table for searching the closed list. Each node in the navigation mesh is given an id that corresponds to an index in the lookup table. The A* algorithm also requires a search of the open list to see if a node is on it, so we created a lookup table for the open list as well. The lookup tables will take up some space since you need two arrays, each of which has a length that is equal to the number of nodes in the search space. A single byte can represent each node in the lookup table since all that is required is a true or false value, which will help with space requirements. Our considerations for optimizations were dealing with time complexity, and thus could accommodate an extra data structure for the open list.

3.1 Dynamically Weighted BDBOP

The algorithm requires four mechanisms: nodes to represent the environment, the two searches that are needed for the bi-directional functionality, and a coordinator to run the two searches.

3.1.1 The Coordinator

The coordinator is in charge of creating the two searches, running them, and receiving information from them. For reasons we discuss further in the System Design section, we were unable to conduct our two searches in parallel. Therefore, the coordinator runs the searches in alternating successions. If a designer also elects to use alternating successions for the bi-directional aspect of the search, they must choose how many iterations, which we refer to as α, each search completes before switching to the opposite search. Our implementation uses an α defined by the number of polygons in the navigation mesh divided by some constant, ρ. Deciding what value to use for ρ may require some experimentation before finding the appropriate value of α for your particular system. We used a trial and error process, during which we noticed that the statistic affected most by changing the value was nodes visited. We used the value of ρ that kept the average number of nodes visited across all of our environments the lowest. Figure 1 lists pseudocode for the Dynamically Weighted BDBOP algorithm's coordinator.

```
openList is the heap that holds the nodes that can be expanded
closedList is a table that holds expanded nodes
currentNode holds the node the search is currently expanding
goal is the desination of the search
 0.nodesExpanded := 0
 1.while (openList <> Empty AND nodesExpanded < α)
 2.  currentNode := openList's popNode()
 3.  nodesExpanded := nodesExpanded + 1
 4.  closeList's addNode(currentNode)
 5.  if (hasGoal(currentNode, goal) = true)
 6.    finalSolutionStart := currentNode
 7.    return 1 // currentNode has the goal in it
 8.  if (doesContainNode(otherAgentClosedList) = true)
 9.    finalSolutionStart := currentNode
10.    return 2 // frontiers meet at currentNode
11.  for all k ∈ N where N is the set of neighbors for currentNode
12.    if (doesContainNode(closedList, k) = false)
13.       g : getCenter(currentNode) – getCenter(k) + gFromStart
14.       if (doesContainNode(openList, k) = false)
15.          openList's addNode(k, currentNode, gCost)
16.       else
17.          lastG := getG(openList, k)
18.          if (g < lastG)
19.             openList's updateNode(k, g)
20.  if (openList = Empty)
21.    return -1 // no path is found
```

Figure 2. Pseudocode for each search's doSearch() function

3.1.2 The Searches

Each of the searches needs an open list, a closed list, a current node, a goal location, and the closed list of the opposite search. The open list is implemented as a min-heap with all of the typical heap operations, with one alteration being that a limit is imposed on the size of the heap. If the size of the heap exceeds the limit after a node is added, the node with the greatest *f* cost value is removed. Figure 3 demonstrates how the limit can be applied to the open list when adding a new node. This limit lowers the space complexity of the search. One must be careful when choosing the limit because a limit too low can lead to an incomplete search. The open list also uses a lookup table to see if a specific node is in the list and a table to see at what index in the list a node is located. The closed list was implemented as an array of nodes whose indices correspond to the id of each node in the search space.

```
n is the max number of nodes the developer wants the list to hold
startingH is the distance from the starting node to the goal
heap a min-heap that represents the openList
nodesHeld is the number of nodes held by the currently heap
E is the number of Nodes Expanded to reach the current Node
-----------------------------------------------------------------
0. addNode (polygon, parentNode, g)
1. if ( nodesHeld = 0)
2.     newNode := createANode (polygon,parentNode,g, goal, startingH, 1)
3. else
4.     nodesExpanded := getExpandedNodesTo (parentNode) + 1
5.     newNode := createANode (polygon, parentNode, g, goal, startingH, E)
6. if (nodesHeld > n)
7.     deleteMaxNode ()
8. heap [nodesHeld] := newNode
9. shiftUp (nodesHeld)
```

Figure 3. Pseudocode for addNode() function on the openList of a search

This allows all searches of the closed list to become O(1) operations. When the coordinator tells a search to execute it gives the search α and the closed list of the opposite search. A search ends when the goal is found, the current node is in the closed list of the opposite search, or the open list is empty. Figure 2 shows the pseudocode for the doSearch() function for each search in Dynamically Weighted BDBOP algorithm. If the iterations complete before any of these conditions are met the search pauses and tells the coordinator that it has not found the goal yet. The coordinator then starts the opposite search, either for the first time or where it previously left off. This continues until one of the searches satisfies an end condition.

3.1.3 The Node

A node holds information about a polygon from the navigation mesh, a reference to its parent node, and the *f*, *g* and *h* costs for the polygon's location in the navigation mesh. The node also calculates the dynamic weight, ω, used on the heuristic. Figure 4 shows pseudocode for calculating the dynamic weight in a node for the Dynamically Weighted BDBOP algorithm.

```
node holds the polygon in the navigation mesh that the new node will hold
parentNode is the node that this node follows in the solution path
g is the cost to get to this node from the start of the search
h is the heuristic cost that is used for each node
f is the total cost to expand this node
E is the number of nodes that were expanded to reach this node
nCost is the startingH divided by the smallest polygon length
startingH is distance from the starting node to the goal
ε is used for the dynamic weight
ω is the dynamic weight that is added to the totalCost
-----------------------------------------------------------------
0.createANode(polygon, parentNodeToAdd, gToAdd, goal,startingH, E)
 1. node := polygon
 2. g := gCostToAdd
 3. parentNode := parentNodeToAdd
 4. nCost := startingH / getSmallestPolygonLength ()
 5. if (E <= nCost)
 6.     ω := 1 – (E  / nCost)
 7. else
 8.     ω := 0
 9. h := getHeuristicCostToGoal (node)
10. f := g + ( 1 + (ε * ω)) * h
```

Figure 4. Pseudocode for creating a node with a dynamically weighed heuristic

4 System Design

4.1 Using the Unity 5.1 Game Engine

Unity 5.1 [9] provided us a platform for creating 3 dimensional environments and a means to run our experiment on them. All the rendered objects in an environment in Unity are made up of triangles. The navigation mesh was constructed by merging the triangles Unity generated that represent walkable surface area and then subdividing this area into convex polygons until every polygon no longer held a static obstacle. It is worth mentioning that our navigation mesh is a polygonal navigation mesh not strictly a triangular mesh. Once the navigation mesh is completed the polygons are given to a search to represent the search space. One impact of using Unity to conduct our searches was that Unity discourages the use of user threads. For this reason, we decided to use alternating successions instead of running searches in parallel for the searches that had a bi-directional nature.

4.2 Environments

The 11 environments used in our experiment were 3 dimensional environments. These environments encapsulated an agent (the starting point), a goal (the end point) and static obstacles. These obstacles were later cut out of the navigation mesh. The size of the environments and the number of obstacles in the environments varied. Most environments were square, with the exception of two special cases we constructed. One was a horseshoe with the goal and starting point each on opposite ends of the horseshoe. The other was a very long narrow rectangle. Both of these environments yielded interesting results when the searches were tested on them. The number of polygons in the navigation meshes ranged from 32 to 8,173. In general, the more obstacles there are in an environment, the larger the number of polygons in the navigation mesh.

4.3 Data Collection

Each search algorithm was run 1000 times on each environment. The following data was collected from each search algorithm for each run:

- Search time (seconds)
- Number of nodes expanded

- Max number of nodes in Open List
- Path length (# of polygons)
- Path cost (meters)

The following data was collected for each environment:

- Number of initial polygons (before building navigation mesh)
- Number of static obstacles
- Number of final polygons (after navigation mesh is built)

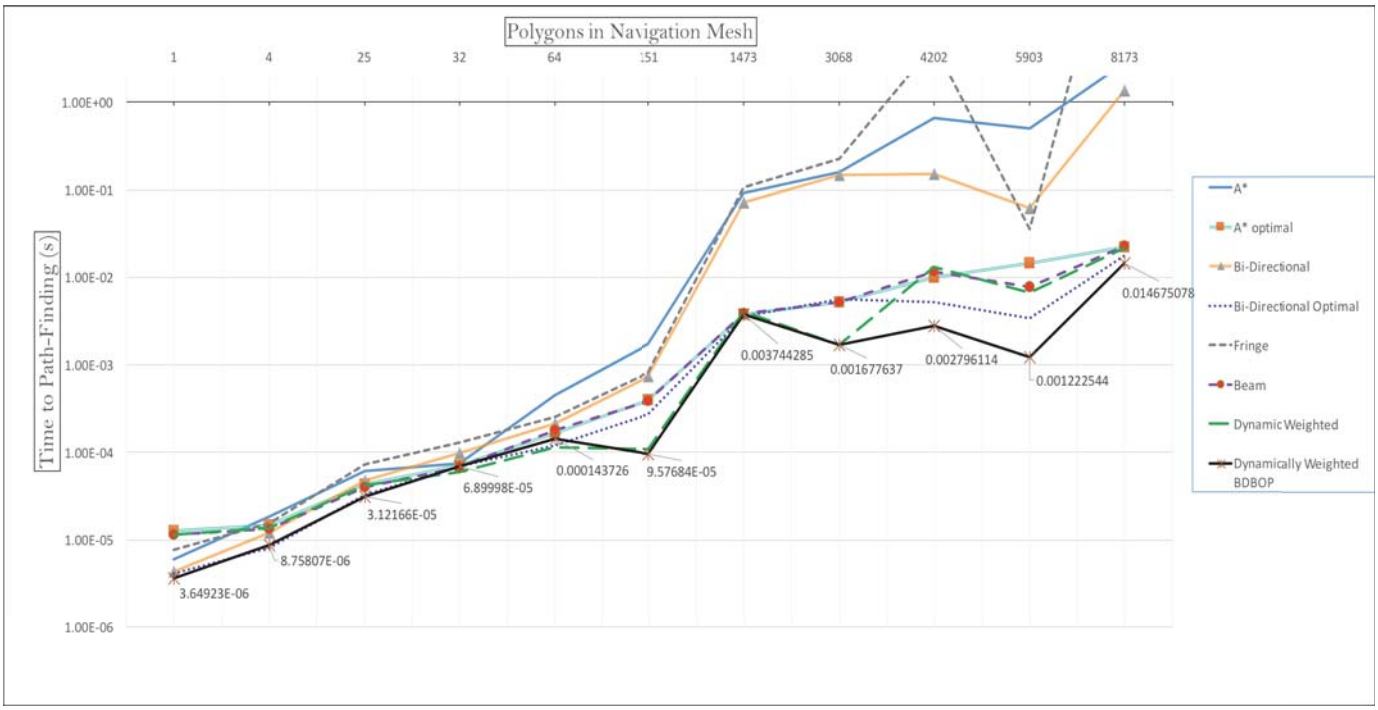


Figure 5. Average search time of each algorithm on each test environment

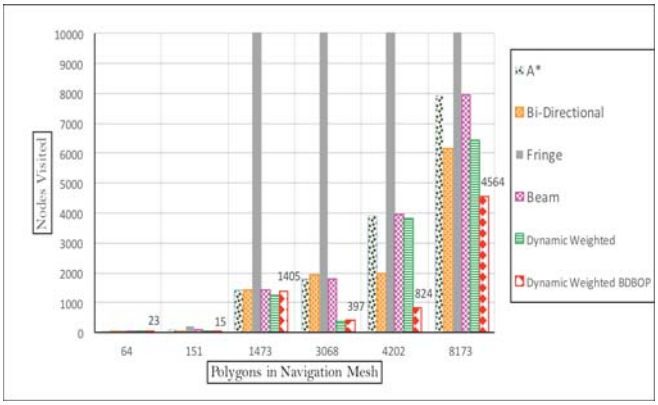


Figure 6. Number of Nodes visited by the 6 search algorithms on 6 test environments

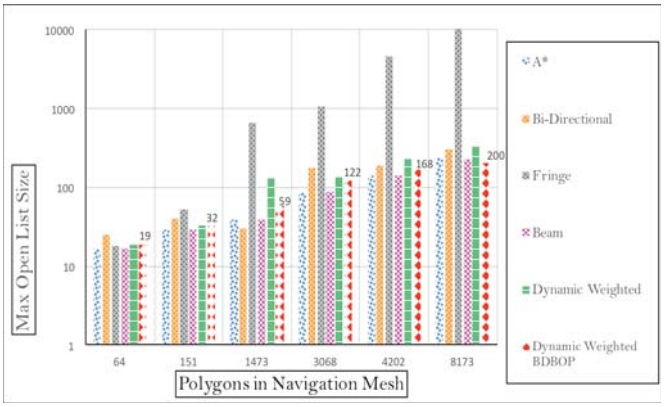


Figure 7. Max Open List Size for the 6 search algorithms on 6 test environments

5 Results

After comparing the data from the different search algorithms against each other we found that Dynamically Weighted BDBOP was the fastest and visited the fewest nodes in eight out of the eleven environments we tested on, losing only by a small margin on the other three environments. The Dynamic Weighting and Bi-Directional components of Dynamically Weighted BDBOP appear to have done their job in reducing the total number of nodes visited by the search. Figure 5 shows the average search time for each algorithm on each of the eleven environments. Figure 6 shows the number of nodes visited by each search on six of the different environments.

Each environment used relatively equal amounts of space with the exception of Fringe search, which performed poorly in terms of space used. Figure 7 shows the maximum number of nodes in the open list for each search algorithm on six of the eleven environments. The limit imposed on the size of the open list of Dynamically Weighted BDBOP was only reached once during testing. This occurred on the environment with the largest number of final polygons (8,173).

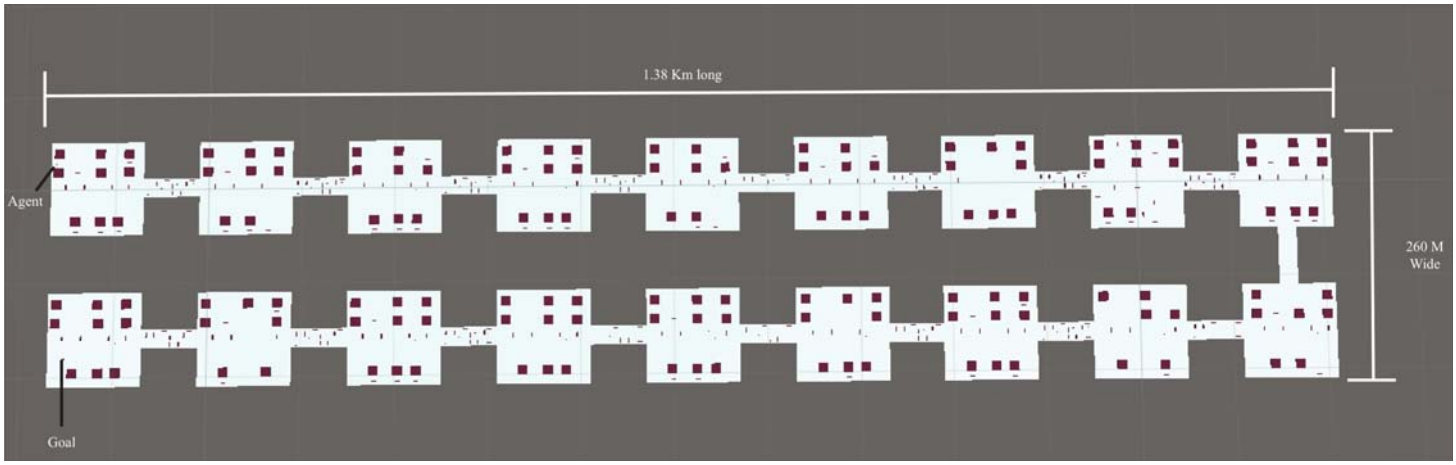


Figure 8. Xanadu map used for testing

		DWBDBOP	A*	Bi-Directional	Dynamic Weighted	Fringe	Beam
Polygon Count	Map Size (m^2)	Path Cost (m)	Path Cost (m)	Path Cost (m)	Path Cost (m)	Path Cost(m)	Path Cost (m)
4	400	20.01665	20.01665	20.01665	20.01665	20.01665	20.01665
25	400	29.2007	29.2007	29.2007	29.2007	29.2007	29.2007
32	1,200	58.10884	48.336	48.336	55.96035	48.336	48.336
64	1,200	63.57973	63.57973	63.57973	63.57973	66.07505	63.57973
151	3,600	103.7901	90.30271	90.30271	103.7901	100.5537	90.30271
1473	200,400	3862.1	3437.157	3445.417	3760.751	3474.831	3437.157
3068	89,600	676.8464	642.2347	643.4597	681.0273	700.7254	642.2347
4202	250,000	1483.118	1483.118	1483.118	1483.118	1483.118	1483.118
5903	2,760,000	31495.76	31376.67	31376.67	31495.76	32732.87	31376.67
8173	500,000	5809.746	4004.154	4005.499	5312.438	4050.381	4004.154

Figure 9. Path costs for each search on 10 test environments

Fringe search performed consistently worse than the other searches except on environments that consisted of long narrow corridors. Our testing included two such environments. One is a straight and narrow path 120 meters wide and 23 kilometers long, the other resembles a horseshoe (Xanadu, shown in Figure 8) and is about 2.8 kilometers in total length. Even though Fringe Search did not perform the best among all the algorithms on these two environments, Fringe search's performance disparity between different types of environments demonstrated that certain searches are better suited for certain environments than for others. Naturally, a game developer will want to pick the appropriate search for any given environment. On an environment such as Xanadu the heuristic is telling the search to go in a direction that it cannot for nearly half the search leading to inefficiency. An intermediary goal in the middle of the horseshoe would help the performance of A* by improving the accuracy of the heuristic in deciding which node to expand next.

The next step in our testing would be to increase the number and diversity of environments we run the search algorithms on in order to get a better idea of the strengths and weaknesses of each search. We hope to discover another unique situation that we can learn from, such as the one we encountered with Fringe Search. The knowledge gained from analyzing such performance anomalies may lead to further methods for improving the searches.

6 Discussion

Dynamically Weighted BDBOP is the fastest of the searches we tested, but does not always give an optimal path. The need for an optimal path in video games is not always the number one concern. Quite often a developer will prefer a faster search with a near optimal path to a slower search with an optimal path. Figure 9 shows the path costs for all the search algorithms on several of the environments along with the environment size and polygon count. Dynamically Weighted BDBOP's path cost is very close to optimal in nearly all cases and even achieves an optimal solution in four of the environments. There was one environment where Dynamically Weighted BDBOP's path cost had a 36 percent difference from the optimal path cost. This was on the map with highest polygon count (8,173), encompassing 500,000 square meters. The regular Dynamically Weighted search also performed poorly on this map in terms of path cost. Environments with such high polygon counts are very

uncommon in video games. For example, Unity 5.1 has a limit of 228 polygons for navigations meshes built with Unity's nav mesh baker [9], and games like 'Jak and Daxter' have a maximum polygon count of 256, with meshes rarely exceeding 64 polygons [10]. Further testing on a larger set of environments that are similar to those used in video games today will provide a better understanding of Dynamically Weighted BDBOP's viability as a path-finding algorithm in video games.

The speed increase Dynamically Weighted BDBOP has over traditional A* can help video game developers with the CPU constraints. With video games becoming more complex for systems to run, the constraints they put on CPUs are becoming a bigger problem. These constraints leave limited resources for other video game requirements such as artificial intelligence. With Dynamically Weighted BDBOP the developer has more resources available after a path is found to handle more complex tasks like decision making for NPCs. This is due to how fast Dynamically Weighted BDBOP can complete a search and give a path back to an agent. A* took on average over a tenth of a second on larger environments and then over two seconds to complete the path on the environment with the largest polygon count (8,173). This would put a major lag in the game. Dynamically Weighted BDBOP on the other hand never took longer than two hundredths of a second to complete a search.

7 Conclusion and Future Work

As video games continue to advance, they are becoming more and more computationally complex, requiring faster hardware and more efficient algorithms to achieve desired performance. Our research focused on improving the efficiency of a typical path-finding method used in video games. Representation of the environment was the first consideration. The use of a navigation mesh instead of a square grid allowed us to reduce the search space. Next we examined the performance of several search algorithms and proposed a variation to A*, which we termed Dynamically Weighted BDBOP. The top performer among the searches we tested was Dynamically Weighted BDBOP, which had the lowest average search time. The various alterations we made to the search algorithms and data structures demonstrated what mechanisms lowered the total search time, which mechanisms increased search time and possible areas for further improvement.

Our future work will be to continue testing and modifying the search algorithms to enhance efficiency. One path involves comparing Bi-Directional search run in parallel against Bi-Directional Search run in alternating successions. We would also like to see how the performance of Bi-Directional Search is affected when the search goal is updated to the opposite search's current node every time that current node changes. Also, we would like to explore if Dynamically Weighted BDBOP would place a lag in the system during path-finding for multiple NPCs at the same time. For testing we would like to use a much larger set of environments, and would prefer to use environments that are typical of those used in the latest video games. Our hope is that further testing and experimentation with the search algorithms will lead to improvements in efficiency that can be beneficial to the video game industry and, more generally, other path-finding applications.

8 References

[1] Xiao Cui and Hao Shi, "Direction Oriented Pathfinding in Video Games", IJAIA, Vol. 2, No. 4, October 2011.

[2] Xiao Cui and Hao Shi, "An Overview of Pathfinding in Navigation Mesh", IJCSNS, Vol. 12, No. 12, December, 2012.

[3] P. Tozour, "Building a Near-Optimal Navigation Mesh," in *AI Game Programming Wisdom*, vol. 1. Rockland, MA: Charles River Media, 2002, ch. 2, sec. 3, pp. 171-185.

[4] G. Snook, "Simplified 3d Movement and Pathfinding Using Navigation Meshes," in *Game Programming Gems*, vol. 1. Boston, Ma: Cengage Learning, 2000, ch. 3, sec. 6, pp. 288-304.

[5] Y. Bjornsson, M. Enzenberger, R. C. Holte and J. Schaeffer, "Fringe search: Beating A* at Pathfinding on Game Maps", *IEEE CIG'05*, 2005, pp. 125-132.

[6] Andru Putra Twinanda, "Fringe search vs A* for NPC movement ", School of Electrical dan Informatics Engineering Institut Teknologi Bandung, Bandung, Indonesia, 2008.

[7] Amit Patel, Amit's A* Pages [Online], Available: http://theory.stanford.edu/~amitp/GameProgramming/, accessed April 9, 2016.

[8] S. Rabin, "A* Speed Optimizations," in *Game Programming Gems*, vol. 1. Boston, Ma: Cengage Learning, 2000, ch. 3, sec. 5, pp. 272-287.

[9] Unity Online Manual [Online], Available: https://unity3d.com/unity/whats-new/unity-5.0, accessed April 9, 2016.

[10] S. White and C. Christensen, "A Fast Approach to Navigation Meshes," in *Game Programming Gems*, vol. 3. Boston, Ma: Cengage Learning, 2002, ch. 3, sec. 6, pp. 307-320.

Using A 3D Interval Type-2 Fuzzy Interpolation System to Improve Robots Calibration Accuracy

Ying Bai [1] and **Dali Wang**[2]
[1]Department of Computer Science & Engineering, Johnson C. Smith University, Charlotte, NC, USA
[2]Department of Physics & Computer Science, Christopher Newport University, Newport News, VA, USA

Abstract - *This paper is an extended research for a novel technique used in the pose error compensations of the robot and manipulator calibration process based on an Interval Type-2 Fuzzy error interpolation (IT2FEI) method. Robot calibrations can be classified into model-based and modeless methods. A model-based calibration method normally requires that the practitioners understand the kinematics of the robot therefore may pose a challenge for field engineers. An alternative yet effective means for robot calibration is to use a modeless method; however with such a method there is a conflict between the calibration accuracy of the robot and the number of grid points used in the calibration task. In this paper, an interval type-2 fuzzy interpolation system is applied to improve the compensation accuracy of the robot in its 3D workspace. An on-line type-2 fuzzy inference system is implemented to meet the needs of on-line robot trajectory planning and control. The simulated results given in this paper show that not only robot compensation accuracy can be greatly improved with this method, but also the calibration process can be significantly simplified, and it is more suitable for practical applications.*

Keywords: Type-2 Fuzzy interpolations, Interval Type-2 Fuzzy logic, Modeless Robotic Calibrations, Robot kinematics, Type-2 Fuzzy control.

1 Introduction

A model-based method for robot calibration involves setting up a kinematic model for the robot, measuring positions and orientations of the robot end-effector, identifying its kinematic parameters and compensating its pose errors by modifying its joint angles [1]. Most of published research results for robot calibration belong to this category. The advantage of a model-based calibration method is that a large workspace can be calibrated accurately and all pose errors within the calibrated workspace can be compensated by joint angles. Its disadvantage lies in the fact that the understanding of kinematic modeling and identification processes needs advanced knowledge in robot kinematics, which may pose a challenge to field engineers.

On the other hand, a modeless method does not go through any kinematic modeling and identification steps. In a pose measuring process, a robot workspace is divided into a sequence of small squares in a 2D case, or cubes in a 3D case with nominal grid points around each cell are assumed known. All position errors on the grid points are measured and recorded by moving the robot through all the grid points. These position errors are stored in memory for future usage. With a modeless method, simple error compensation for a target position can be realized by interpolating errors from its neighboring grid points [2]. Its disadvantage, however, is conflict between calibration accuracy and number of grid points. In spite of this, because of its simplicity and effectiveness, the modeless calibration technique is widely adopted in industrial applications.

There are also several different alternative approaches to the model-based and modeless robot calibration methods discussed at the above. Whitney and Shamma [3][4] reported a non-parametric accuracy compensation method using polynomial approximating functions; however a nominal inverse kinematic model of the robot to be calibrated is still needed to calculate the joint position vector corresponding to the desired position of the robot end-effector. Another non-parametric compensation approach is to divide the workspace of the robot into a sequence of discrete areas or cubic cells and then use a numerical procedure to determine the inverse kinematic solution for each area or cubic cell. The problem of implementing that approach is that a huge size of memory space is needed to store those inverse kinematic solutions. To solve this problem, James Albus [5-6] developed a so-called Cerebellar Model Articulation Controller (CMAC) to reduce the memory size. By using this non-parametric compensation method, the workspace of the compensated robot is about a quarter of the normal robot workspace and the accuracy is around 0.12 mm [3].

Most traditional modeless calibration methods use some interpolation techniques, such as linear interpolation, bilinear interpolation and cubic spline interpolation, which are widely utilized in industrial settings. Because actual error distribution of a robot may not satisfy linear or cubic models, the estimated errors would not be as accurate as desired. To improve numerical interpolation accuracy, different approaches have been attempted by researchers. Zhuang and Wu [7] reported a histogram method for estimating the optimal membership function distribution in order to improve the accuracy of measured positions. Carlson and Looney reported a new image interpolation method that refined the bilinear and fuzzy interpolation techniques [8]. Song, Smith

and Rizk provided an optimized fuzzy logic controller for the generation of optimal trajectories based on Optimal Control Table (OCT) [9]. Bai and Wang developed a type-1 fuzzy interpolation technique to improve the accuracy of robots calibration in the 2D and 3D spaces [10-12].

In the research reported in this paper, a modeless on-line interval type-2 fuzzy interpolation method is developed and implemented to improve the calibration accuracy of the robot in its 3D workspace. A comparison between the compensation results of using the type-1 and the type-2 fuzzy interpolation techniques is made through simulation studies. Three error models, sinusoidal waveform, normal and uniform distributed errors, are tested. Because the actual neighboring errors on each grid point are random distributed, and the error surfaces on each cell are also random functions at a certain moment, the traditional membership functions are not suitable for our study (each cell is a cubic that is surrounded by 8 grid points). Moreover, the lookup table may not be calculated in advance until the output membership functions that are associated with the actual position errors on grid points are determined during the error measurement process. This means that one cannot use the traditional off-line fuzzy technology to obtain the lookup table for fuzzy error interpolation. The crisp output must be estimated on-line based on the errors at the neighboring grid points, which means that the output membership functions must be dynamic functions of the neighboring errors.

The remainder of the paper is organized as follows: The type-1 fuzzy interpolation technique tailored to robot calibration is discussed in Section 2. Section 3 provides an introduction to interval type-2 fuzzy interpolation technique, and the simulation results to illustrate the feasibility and merits of using the type-2 fuzzy interpolation over the type-1 fuzzy interpolation method is given in section 4. Concluding remarks are given in Section 5.

2 Online 3D Type-1 Fuzzy Interpolation

As has been mentioned, a modeless compensation method involves only two steps: robot pose measuring and error compensation. Compared with the model-based compensation, the kinematic modeling and identification steps are no longer needed for the modeless compensation method. This can significantly save workload in a robot calibration. It starts with a measurement process, in which position errors of the robot end-effector are measured at all specified grid points on a sequence of pre-determined cubic cell within the robot workspace. A suitable interpolation technique is then applied to fit the target position error based on the position errors of neighboring grid points around the target position. Among several interpolation methods, the linear interpolation technique is a simple and popular one, and is widely adopted by modeless robot calibration.

When using the modeless method to calibrate robots, it is necessary to have some proper measurement device with sufficient accuracy, such as a Coordinate Measurement Machine (CMM), a set of cameras (a stereo camera system) or

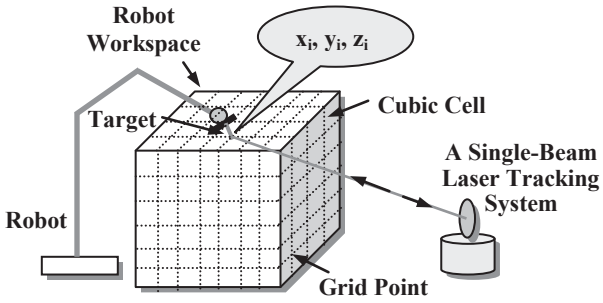


Fig.1. A setup for modeless robot calibrations.

a laser tracking system, which is used to gather robot position errors as the robot moves its end-effector to all grid points on the pre-determined cubic cell.

Fig.1 shows a configuration setup for modeless robots calibration process.

The robot workspace is divided into a sequence of small cubic cells and each small cell is surrounded by 8 neighboring grid points, which is shown in Fig.2. At each grid point, the laser tracker is used to check the position errors of the end-effector of the robot. In Fig.2, the desired position of the grid point i is (x_i, y_i, z_i), and the actual position of the robot end-effector is (x_i', y_i', z_i'). The position errors for this grid point are $e_x = x_i - x_i'$, $e_y = y_i - y_i'$ and $e_z = z_i - z_i'$. The robot will be moved to all grid points on the robot workspace, and all position errors on these grid points will be measured and stored in the memory for future usage. During the compensation process, as the robot is moved to a certain target point on the workspace, an interpolation technique, say the linear interpolation method, is used to estimate the target position error based on the errors of the neighboring grid points around the target point, and finally these errors are added into the target position to obtain the compensated position, with which, the robot is commended to move to that position.

Basically, the linear interpolation method is based on the assumption that errors in a cubic cell are distributed uniformly, and the interpolated error is obtained from three plans that are constructed based on 8 neighboring errors on the grid points around the target cubic cell. However in the real world, this assumption may not be valid. Position errors on each cell may not be distributed uniformly. We can consider the $e_x(x', y', z')$ as the fourth dimension function based on the position x', y' and z' inside each cubic cell. Same

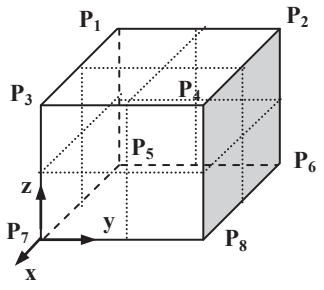


Fig.2. A cubic cell in robot workspace

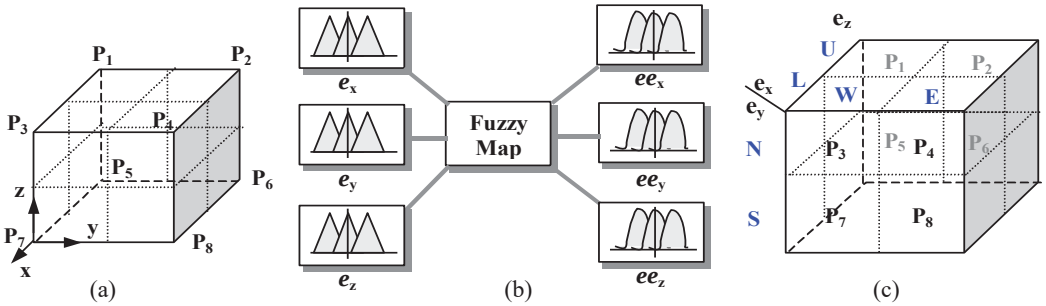


Fig. 3. Definition of the fuzzy interpolation inference system

consideration is held for e_y (x', y', z') and e_z (x', y', z'). The compensation accuracy of using either the linear or bilinear interpolation technique is limited by this assumption.

In order to solve this problem and to improve the compensation accuracy, an on-line fuzzy interpolation method is introduced. The traditional fuzzy inference system uses pre-defined membership functions and control rules to construct lookup tables, and then pick up the associated control output from the lookup table as the fuzzy inference system works in an application. This type of system is called off-line fuzzy inference system because all inputs and outputs have been defined prior to the application process. In this study, this off-line fuzzy system cannot meet the task requirement based on the following reasons: first, the position error of the target point is estimated based on 8 errors of the neighboring grid points, and these 8 neighboring errors are random distributed. The off-line fuzzy output membership functions are defined based on the errors range, in here, the neighboring errors' range. However this range estimation is not as good as the actual errors obtained on the grid points. Second, since one cell needs one lookup table for the off-line fuzzy system, it requires huge memory space to save a large number of lookup tables, which is both space and time consuming and therefore is not suitable for industrial applications. For example, if the robot workspace is divided into $40 \times 40 \times 40$ small cubic cells, one needs to have 64000 lookup tables! By using an on-line dynamic fuzzy inference system, one can estimate the target position error on-line by combining the output membership functions (MFs) that are obtained from the actual errors on the neighboring grid points. With this approach, one doesn't need off-line lookup tables at all. This means that the system determines the output membership functions only after the fuzzy inference system is applied to an actual process, using real errors on the grid points, not a range.

Fig.3 shows conceptually the definition of the fuzzy error interpolation inference system. Each small cubic cell that is surrounded by 8 neighboring grid points is defined as a cell, and furthermore this cell is further divided equally into 8 smaller cells, which are shown in Fig. 3 (a). The position error at each grid point is defined as P_1, P_2, P_3, P_4, P_5, P_6, P_7 and P_8.

For the type-1 fuzzy inference system, we apply the interpolations in three dimensions separately, so the inputs to the fuzzy inference system are e_x, e_y and e_z. The outputs are ee_x, ee_y and ee_z, which are shown in Fig. 3 (b).

The control rule is shown in Fig. 3 (c), which is straightforward and based on the human knowledge. The only point to be emphasized is that each P_i should be considered as a combination of three error components on each grid point, P_{xi}, P_{yi} and P_{zi}, which are responding to errors in x, y and z directions. The distances between the neighboring grid points of each cell on the workspace are 20 *mm* in x, y and z directions for this study, which is a standard interval for a mid-size calibration workspace. Totally the entire workspace

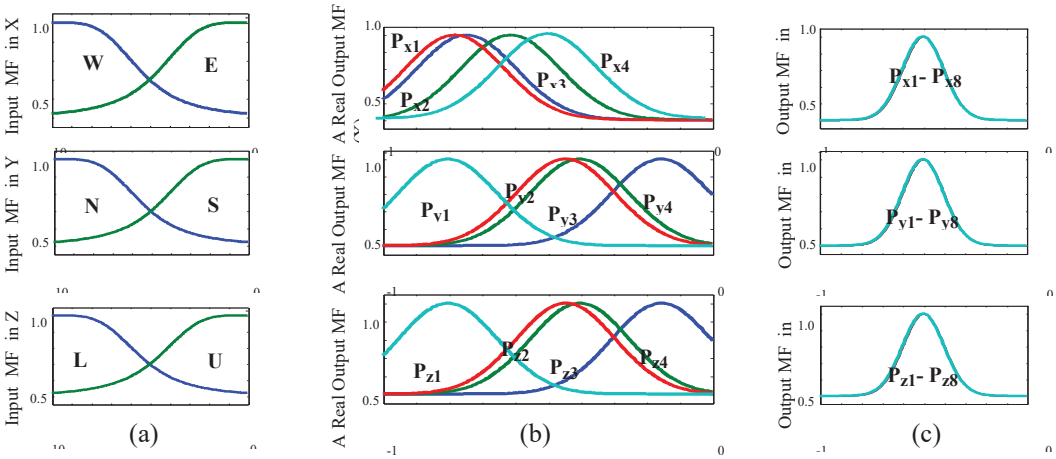


Fig. 4. Input and output membership functions

Includes 40×40×40 cells, which is equivalent to an 800×800×800mm^3 space. The input MFs and the predefined output MFs are shown in Fig. 4.

The predefined output MF is used as a default, and the final output MF will be obtained by shifting the default one by the actual error values on the grid points. For each cell, 8 output MFs are implemented and each one is associated with the error at one grid point. In Fig. 4 (b), only 4 output MFs are shown here because of the space limitation. In an application, total 8 membership functions should be utilized.

The Gaussian-Bell waveforms are selected as the shape of the membership functions for three inputs As shown in Fig. 4 (a), the ranges of inputs are between –10 and 10 *mm* (20 *mm* interval on grid points). W and E represent the inputs located at different areas in the x direction, N and S represent the inputs located at different areas in the y direction, and L and U represent the inputs located at different areas in the z direction. Unlike the traditional fuzzy inference system, in which all membership functions should be determined to produce the lookup table prior to the implementation of the fuzzy system, in this study, the output membership functions will not be defined until the implementation of the fuzzy error mapping to compensate the position errors. So the output membership functions will be determined during the application of the fuzzy inference system on-line or dynamically. This is called dynamic fuzzy system. Fig. 4 (b) shows an example of the output membership functions, which are related to the simulated random errors at neighboring grid points. Each P_{xi}, P_{yi} and P_{zi} responds to the position error at the *ith* grid point in the x, y and z directions, respectively. During the design stage, all output membership functions should be 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. These output membership functions will be determined on-line based on the errors of the neighboring grid points around the target point in the workspace as mentioned above.

The control rules shown in Fig. 3 (c) can be interpreted as follows after the output membership functions are determined:

- If e_x is W and e_y is N, and e_z is U, then ee_x is P_{x1}, ee_y is P_{y1} and ee_z is P_{z1}. (P_1)
- If e_x is W and e_y is N, and e_z is L, then ee_x is P_{x3}, ee_y is P_{y3} and ee_z is P_{z3}. (P_3)
- If e_x is W and e_y is S, and e_z is U, then ee_x is P_{x5}, ee_y is P_{y5} and ee_z is P_{z5}. (P_5)
- If e_x is W and e_y is S, and e_z is L, then ee_x is P_{x7}, ee_y is P_{y7} and ee_z is P_{z7}. (P_7)
- If e_x is E and e_y is N, and e_z is U, then ee_x is P_{x2}, ee_y is P_{y2} and ee_z is P_{z2}. (P_2)
- If e_x is E and e_y is N, and e_z is L, then ee_x is P_{x4}, ee_y is P_{y4} and ee_z is P_{z4}. (P_4)
- If e_x is E and e_y is S, and e_z is U, then ee_x is P_{x6}, ee_y is P_{y6} and ee_z is P_{z6}. (P_6)
- If e_x is E and e_y is S, and e_z is L, then ee_x is P_{x8}, ee_y is P_{y8} and ee_z is P_{z8}. (P_8) (1)

The control rules are straightforward and they are based on the human knowledge. The error on P_1 grid point should be taken more weight if the target position (input) is located inside the NWU area on a cell. Similar conclusion can be derived for errors on all other grid points.

3 Interval Type 2 Fuzzy Interpolations

3.1 Overview interval type 2 fuzzy interpolation

Similar to type 1 fuzzy inference system, the type 2 fuzzy inference systems still use the input and output membership functions, combined with the control rules, to derive the outputs [13-21]. 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. 5 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 $\tilde{B}$ is:

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

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

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

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

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

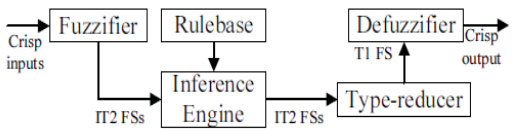


Fig. 5. A functional block diagram of the Interval Type-2 Fuzzy system.

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 [24-25] 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 [24]:

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

4) Compute the defuzzified output as:

$$y = \frac{y_l + y_r}{2} \qquad (3.5)$$

3.2 Membership functions

Similar to type-1 fuzzy interpolation system, the input membership functions for *x*, *y* and *z* directions and the predefined output membership functions for IT2 FLS are shown in Fig. 6. 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, and U and L to the location of the inputs in *z* direction. For real outputs, 8 membership functions, $p_{x1} \sim p_{x8}$, should be designed for the *x* direction, and another 8 membership functions, $p_{y1} \sim p_{y8}$, and $p_{z1} \sim p_{z8}$ are to be built for the *y* and *z* directions. 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 locations based on the actual position errors on each grid point. In Fig. 6, these functions are all displayed but not in the default locations.

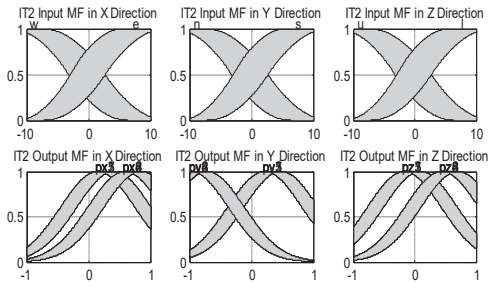


Fig. 6. The input and output membership functions for IT2 FLS

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.

4 Simulation results

Extensive simulation for position compensations has been performed in order to illustrate the effectiveness of this IT2 fuzzy error interpolation technique in comparison to the type-1 FLS. Because of the similarity, the angles simulations are not shown in this paper. Due to the random nature of the position errors, three different types of error are simulated in this study. These are:

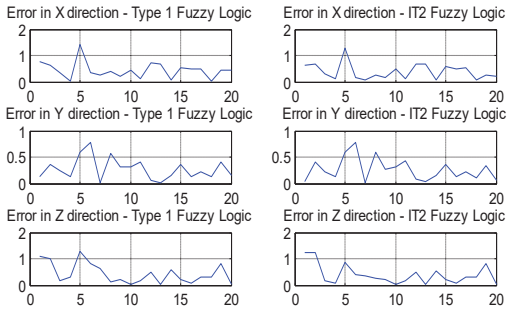


Fig.7. Interpolation results – normal distributed errors

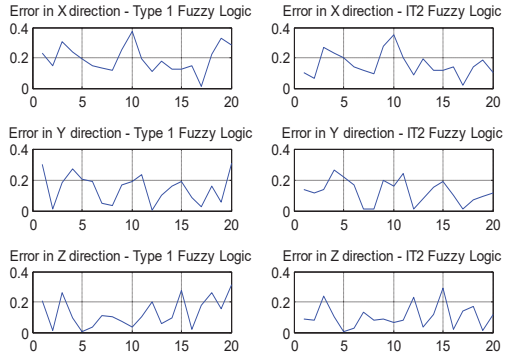


Fig.8. Interpolation results – uniform distributed errors

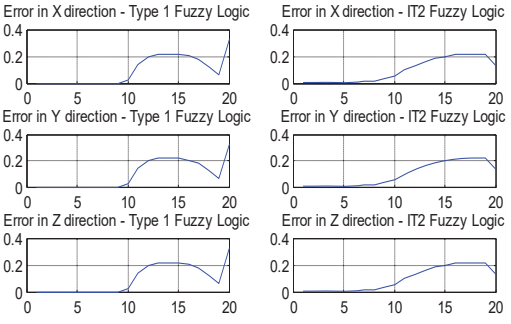


Fig.9. Interpolation results – sinusoid waveform errors.

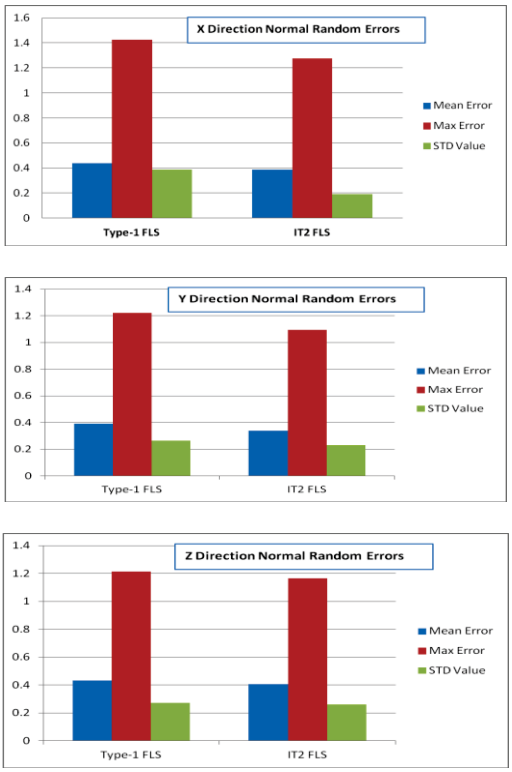


Fig.10. Normal distributed errors

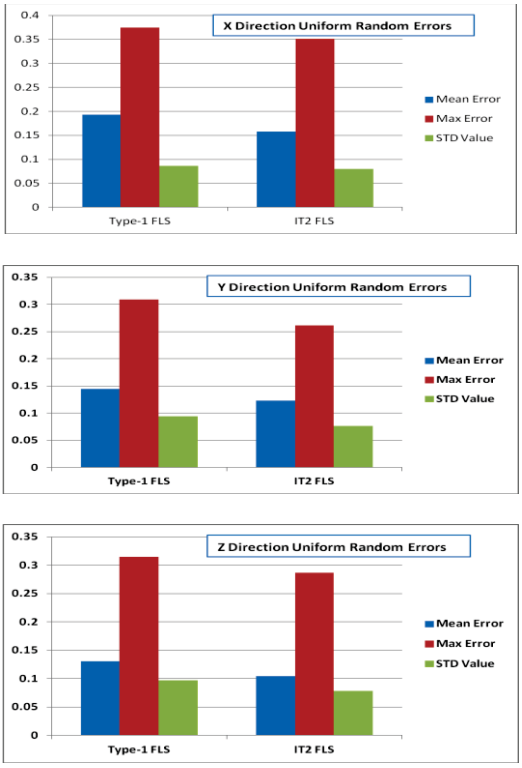


Fig.11. Uniform distributed errors.

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

Figs 7, 8 and 9 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 10 to 11 show comparisons in mean error, maximum error and STD values between type-1 and IT2 fuzzy error interpolation techniques in the histograms. Only the normal and uniform random errors are shown here since no significant differences are existed between the type-1 and type-2 fuzzy interpolations for the sinusoidal error distributions.

It can be seen that both mean errors and maximum errors of the IT2 fuzzy error interpolation technique are smaller than 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, y* and *z* direction), the maximum errors of the IT2 fuzzy interpolation method are about 4% smaller than those of the type-1 FLS method.

5 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.

6 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 [27-28]. We could not develop and share this paper with all of you without his sincerely help and support.

7 References

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

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

[3] J. S. Shamma and D. E. Whitney, A method for inverse robot calibration. *Journal of Dynamic Systems, Measurement, and Control*, 109 (1): 36-43, March 1987.

[4] J. S. Shamma, *A Method for Inverse Robot Calibration*, Master's thesis, Massachusetts Institute of Technology, January 1985.

[5] J. S. Albus, "Data storage in the cerebella model articulation controller (CMAC)," *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, 97:228-233, September 1975.

[6] J. S. Albus, "A new approach to manipulator control: The cerebella model articulation controller (CMAC)," *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, 97:220-227, September 1975.

[7] 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, February 2001, pp.125 – 131.

[8] B. Carlson and C. Looney, "A new method of image interpolation," *Proceedings of the ISCA 16th International Conference on Computers and Their Applications*, Seattle Washington, March 28-30, 2001, pp.25-28.

[9] 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*, Tokyo, Japan, Oct 12, 1999, pp.138-143.

[10] Ying Bai and Dali Wang, "Improve the position measurement accuracy using a fuzzy error interpolation technique," *Proceedings Of The IEEE 2003 International Symposium On Computational Intelligence For Measurement Systems and Applications*. July 29-31, 2003, Lugano, Switzerland, pp. 227 – 232.

[11] Ying Bai & Dali Wang, "On the Comparison of Trilinear, Cubic Spline and Fuzzy Interpolation Methods in the High Accuracy Measurements", *IEEE Trans. on Fuzzy Systems*, Vol. 18, No. 5, October 2010, pp.1016-1022.

[12] Ying Bai, "On the Comparison of Model-Based and Modeless Robotic Calibration Based on a Fuzzy Interpolation Method", *International Journal of Advanced Manufacturing Technology*, Springer, Vol.31, No.11-12, February 2007, pp.1243-1250

[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] Cheol-Joong Kim and Dongkyoung Chwa, "Obstacle Avoidance Method for Wheeled Mobile Robots Using Interval Type-2 Fuzzy Neural Network", IEEE Trans. On Fuzzy Systems, Vol.23, No.3, June 2015, pp.677-687.

[20] Ziyad T. Allawi and Turki Y. Abdalla, "An Optimal Defuzzification Method for Interval Type-2 Fuzzy Logic Control Scheme", 2015 Science and Information Conference, July 28-30, 2015, London, UK, pp.619-627.

[21] Tufan Kumbasar and Hani Hagras, "A Self-Tuning zSlices-Based general Type-2 Fuzzy PI Controller", IEEE Trans. On Fuzzy Systems, Vol.23, No.4, August 2015, pp.991-1013.

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

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

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

[27] 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.

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

Automatic Surveying and Recognition of a Remote Target Using Blob Detection and Filtration for Unmanned Mobile Surveying System

Jarjees A. Khidir[1,3] and Gary T. Anderson[2,3]
[1]Dept. of Computer Science, jakhidir@ualr.edu
[2]Professor and Chair, Dept. of Systems Engineering, gtanderson@ualr.edu
[3]University of Arkansas at Little Rock, 2801 South University Ave., Little Rock, AR, USA, 72204

Abstract – *In this research, a fully automated system is designed to survey and recognize a remote and stationary target using a single monocular camera and a pan/tilt unit. The system development is specially directed to build an unmanned mobile biogenic gas concentration detection system. An automatic target recognition algorithm is designed to achieve fast and accurate target recognition using blob detection and filtration under OpenCV library. Two formulas have been developed to automatically drive the pan/tilt unit and adapt its panning and tilting speeds to the current zooming level of the camera. Indoor and outdoor test results are presented to show that the proposed algorithm has been successfully applied to recognize the target from various distances, view angles, and environmental conditions.*

Keywords: Image Processing, Blob Detection, Target Surveying, Target Recognition, Target Alignment.

1 Introduction

Environmental monitoring systems are designed to help collect data, analyze it, and then provide an indication of the environmental status. Environmental monitoring is often a highly labor-intensive activity and poses a risk to human monitoring teams in hazardous environments.[1] Therefore, it is important to research the automation of these systems. Air quality monitoring is one of the many applications of environmental monitoring systems.[2,3] A traditional gas concentration monitoring system consists of a network of sensors "electronic noses" placed at key locations or carried by mobile robotics.[4] These sensors are able to detect and collect information about gas concentrations at the local position of the mobile robots. A big disadvantage of electronic noses is that they can only sense over a limited range of distances.[1] More efficient monitoring systems use an open-path laser beam to detect gas concentration over longer distances by measuring the optical absorption of the laser signal.[5,6]

In a project supported by NASA, a new unmanned air quality monitoring system has been proposed to look for gases of biogenic origins.[1,7,8] The system is designed to achieve rapid and efficient gas concentration detection and source localization over a wide region of local terrain and a circle of up to 1 km in radius. The system can be used in unmanned missions to detect toxic gases in life threatening environments. A mobile robot moves along a spiral trajectory to quickly survey a wide area while approaching a remote retro-reflector (the target).[7] An open-path spectrometer is built into the proposed monitoring system to be used for near-ground biogenic gas concentration detection. The spectrometer shoots a laser beam to the target and examines the reflected beam looking for the absorption peak of the investigated gas.[1] The laser beam needs to be constantly aligned to the target while the mobile robot is moving.[9] Therefore, the system has to be able to automatically survey and recognize the retro-reflector using an appropriate Automatic Target Recognition (ATR) algorithm.

Automatic target recognition refers to the use of a computer processing power to automatically detect and recognize the pattern of a specific target within a collected sensor data.[10-12] Based on the targeted application, the sensor data may be collected from various input sources, like sound, image, etc. Image based ATR algorithms are designed to acquire data from visual sources, like forward-looking infrared (FUR) cameras, synthetic-aperture radars (SAR), video cameras, and laser radars.[10]

This research investigates the automation of the target surveying and recognition operations of the proposed monitoring system. Due to the limited power supply in some deserted areas, the target is designed to be passive and stationary. Therefore, the only way to survey and recognize it is to apply image processing techniques on a captured video, looking for its image in the video frames. A new algorithm has been developed to automatically survey and recognize the target, using a single monocular camera and a pan/tilt unit.[9] The algorithm sweeps over a large area of the surrounding scene, surveying for the target without prior information about its exact location. The pan/tilt unit is used to search along the x and y directions (panning and tilting), while the camera is used to achieve an in-depth search through the z direction (zooming). Two formulas were developed to dynamically adapt the speeds of both panning and tilting motions of the pan/tilt unit so that it is appropriate to the current zooming level of the camera. Using blob detection and filtration under OpenCV library,

the proposed algorithm is able to recognize the target from various distances, view angles, and environment conditions. When the target is detected, the system automatically aligns the laser beam to the retro-reflector, and keeps it aligned in real-time while the mobile robot platform is moving.[9]

The following sections describe the various hardware and software aspects of the proposed air quality monitoring system. Section 2 presents an overview of the hardware components used to build the monitoring system. Section 3 shows the detailed stages of the automatic target surveying and recognition system algorithm. Section 4 shows some experimental indoor and outdoor test results for the program execution.

2 Air Quality Monitoring System

The operation of the proposed monitoring system is pictured schematically in Fig. 1.

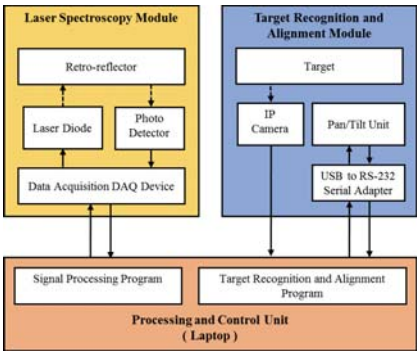
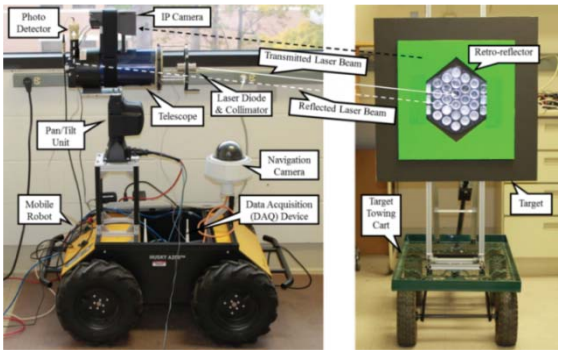


Fig. 1 A schematic diagram of the proposed monitoring system.

The system consists of two main modules, the laser spectroscopy module, and the target recognition and alignment module.[9] The robotic vehicle (Clearpath Robotics, Husky A200) (Fig. 2a) that carries the spectrometer moves around the local terrain to detect and measure the concentration of the investigated gas. The vehicle is powered by four 24VDC motors with max speed 1 m/s. Execution of commanded mobile monitoring, site surveying, trajectory tracking, basic vehicle control and movement, and obstacle avoidance are directly handled by the embedded robot controller.



(a) (b)

Fig. 2 The proposed monitoring spectroscopy system: (a) The laser spectrometer mounted on the Husky robot, and (b) The target assembly.

2.1 Laser Spectroscopy Module

The open-path Tunable Diode Laser Spectrometer (TDLS) system (Fig. 2) is designed for wide area surveys. The major components of the system include a laser diode, a photo detector, a data acquisition (DAQ) device, a telescope, and a retro-reflector. The system works when the laser diode transmits a laser beam to the remote retro-reflector. The retro-reflector reflects the laser beam back to the spectrometer, to be caught by the photo detector. The existence of an absorption peak on the reflected signal indicates that the investigated gas is present in the space between the spectrometer and the retro-reflector.

2.2 Target Recognition and Alignment Module

The target recognition and alignment module consists of a single monocular IP camera (Samsung SNZ-5200); powered by a 12 VDC power supply, and a pan/tilt unit (FLIR PTU-D48E); powered by a 24 VDC power supply (Fig. 2a). Both are controlled by a laptop computer (Lenovo ThinkPad W530). The camera is connected to the LAN connector of the laptop; whereas the pan/tilt unit is connected to the laptop via a USB-to-RS-232 serial adapter. The whole system is mounted on a mobile robot platform (Clearpath Robotics, Husky A200). While the mobile robot is moving, the pan/tilt unit is used to continually adjust the pose of the telescope so that the laser beam is always aligned to the retro-reflector. Finally, the target assembly is mounted on a mobile cart to be easily towed by the mobile robot whenever it is necessary (Fig. 2b).

The target consists of a passive stationary retro-reflector assembly integrated into a black hexagonal panel (Fig. 3). The retro-reflector consists of a flat array of small corner cube mirrors embedded into a hexagonal housing. With this arrangement, the retro-reflector is able to reflect an emitted laser bean back to the spectrometer from a range of angles from $-60^{\circ}\ to\ +60^{\circ}$. The largest width (major axis) of the hexagonal housing is 38 cm and its smallest width is 20 cm.

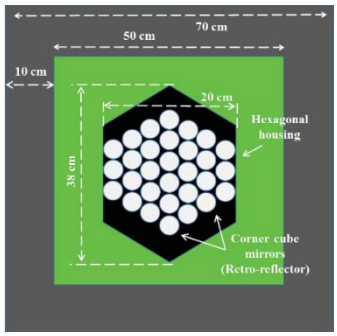


Fig. 3 The target assembly

The retro-reflector is backed by a green square object of 50 cm wide to help recognizing it from longer distances. The green square is in turns framed by a 10 cm wide black border to help separating its image from the background scene, especially when the target is installed in front of a green background while performing an outdoor mission. Because of the high color contrast between the target and the surrounding scene, the program can easily recognize it when it comes to the field view of the camera.[9]

3 Automatic Target Surveying and Recognition System Algorithm

This algorithm was developed to perform automatic surveying and recognition of a remote, passive, and stationary target (Fig. 2). The target is supposed to be recognized from various x, y, and z pose angles. Built with C++ language and the underlying device drivers of the incorporated hardware equipment, a software program was designed to implement the proposed algorithm. This program starts surveying for the target by commanding the pan/tilt unit to yaw around and capturing video frames of the surrounding scene from the IP camera at the same time. The captured video is processed frame by frame in real-time to search for the existence of the target in the field of view (FOV) of the camera. When the target is detected, the pan/tilt unit movement is automatically fine-tuned in both yaw and pitch motions to align the target to the center points of the successive video frames. Once the spectrometer is aligned to the remote retro-reflector, the air quality monitoring task is started. A Graphical User Interface (GUI) was designed to help managing and controlling the program execution interactively (Fig. 4).

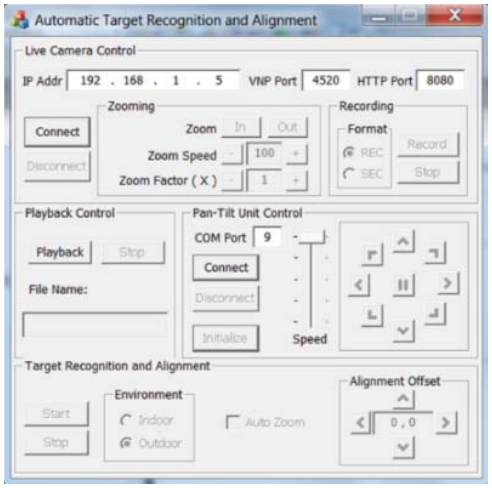


Fig. 4 Graphical user interface of the program.

3.1 Target Surveying Stage

Since the target is passive and stationary, it does not provide an easy indication about its presence, like an electrical signal, a light source, or a simple motion. Therefore, the only way to find the target is to search for its distinctive features within images captured from the surrounding scene. In this stage, the program sweeps the scene searching for the target from a panning angle range of $-180^{\circ}\ to\ +180^{\circ}$. The search is started when the program commands the pan/tilt unit to yaw around, and captures live video from the camera (Fig. 5). The captured video is processed frame by frame to search for the existence of the target in the field of view of the camera.

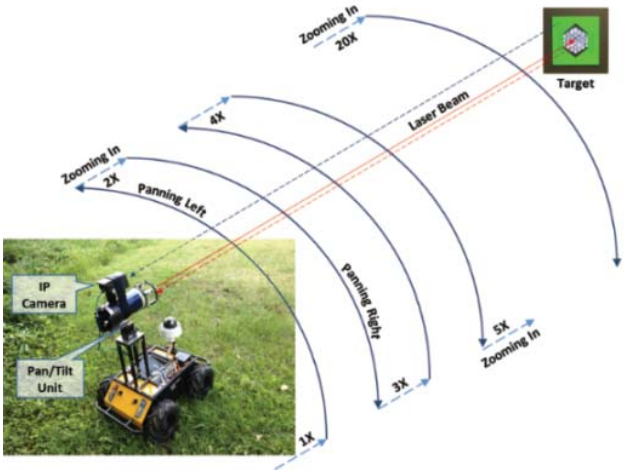


Fig. 5 The program surveying for the target.

To help recognize the target at longer distances, an automatic zooming option with zooming levels of 1x to 20x, is provided (Fig. 5). As the outdoor tests in the experimental results section will show, the target could be recognized from 100 m away at a zooming level of 15x. Two formulas, Equations (1) and (2), have been developed to automatically adapt the speeds of both the panning and tilting motions of the pan/tilt unit to be proportional to the current zooming level of the camera.

$$P_{speed} = \lfloor P_{min} * (F_{max} - F_{current} + 1) \rfloor \quad (1)$$

$$T_{speed} = \lfloor T_{min} * (F_{max} - F_{current} + 1) \rfloor \quad (2)$$

Where:

P_{min} is the minimum panning speed (30 positions/second).

T_{min} is the minimum tilting speed (30 positions/second).

F_{max} is the maximum zooming level of the camera (20x).

$F_{current}$ is the current zooming level of the camera.

With these formulas, the pan/tilt unit moves at a speed of 600 positions per second (15.43 degrees/second) when the camera is zoomed at the 1x level [$30 * (20 - 1 + 1) = 600$], given the minimum assigned speed is 30 positions per second. However, the speed is only 30 positions per second (0.77 degrees/second) when the camera is zoomed at the 20x level [$30 * (20 - 20 + 1) = 30$]. Thus, the panning and tilting speeds are decreased as the zooming level of the camera is increased accordingly (Fig. 6).

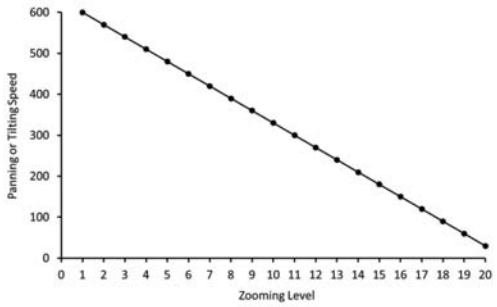


Fig. 6 Zooming levels vs panning or tilting speeds.

3.2 Frame Preprocessing Stage

While the pan/tilt unit is moving, during the target surveying, a live video is captured for the surrounding scene. In this stage, a sequence of image processing operations is applied to each frame to get it prepared for the next stage. In brief, the camera captures video frames of H.264 format with a resolution of 640×480 pixels. The contents of each video frame are scanned to filter out all but green colored pixels (Fig. 3, the target is green). Then, the resulted frame is converted from a colored image to a black and white (binary) image. Finally, the binary image is forwarded to the next stage to search for the target in its contents. The following is a detailed explanation for these operations.

First the color coding of the video frame is converted from RGB color space to HSV color space. The purpose of RGB to HSV conversion is to simplify color separation. Motivated by the human visual system, the HSV coding gives more meaningful color values over RGB coding.[13] The resulted HSV frame is scanned pixel-by-pixel to filter out any non-green pixels, and yet, converting the frame from colored coding to black and white (binary) coding. Each green pixel is changed to the value of 255 (white color), and any other color is replaced by the value of 0 (black color). This step is done by splitting the HSV frame into three separate frames for each of H, S, and V color components. Then, a separate filtering process is applied to each component frame (Equations (3) to (7)).

$$H_{ij} = \begin{cases} 255 & if\ 60 \le H_{ij} \le 90 \\ 0 & Otherwise \end{cases} \quad (3)$$

$$S_{ij} = \begin{cases} 255 & if\ S_{ij} \ge 70 \\ 0 & Otherwise \end{cases} \text{ (Indoor)} \quad (4)$$

$$S_{ij} = \begin{cases} 255 & if\ S_{ij} \ge 1 \\ 0 & Otherwise \end{cases} \text{ (Outdoor)} \quad (5)$$

$$V_{ij} = \begin{cases} 255 & if\ V_{ij} \ge 70 \\ 0 & Otherwise \end{cases} \text{ (Indoor)} \quad (6)$$

$$V_{ij} = \begin{cases} 255 & if\ V_{ij} \ge 200 \\ 0 & Otherwise \end{cases} \text{ (Outdoor)} \quad (7)$$

$$\forall i \in \{0,1,\dots,H_F(Frame\ Height)\}$$
$$\forall j \in \{0,1,\dots,W_F(Frame\ Width)\}$$

Then, the three component frames H, S, and V are combined into one binary frame B, using two bitwise AND operations (Equation (8)).

$$B_{ij} = H_{ij} \wedge S_{ij} \wedge V_{ij} \quad (8)$$
$$\forall i \in \{0,1,\dots,H_F(Frame\ Height)\}$$
$$\forall j \in \{0,1,\dots,W_F(Frame\ Width)\}$$

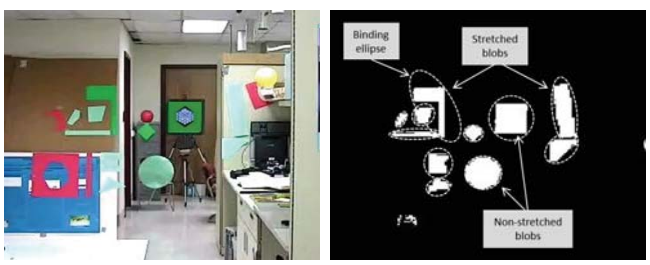
Finally, a Median Filter[14] with an aperture size of 3×3 is applied to the binary frame B to reduce the noise pixels while keeping the objects' edges relatively smooth. The advantage of noise reduction was to increase the program execution speed per frame.

3.3 Target Recognition Stage

In this stage, the program applies a blob detection algorithm for feature extraction to the binary image frame, obtained from the previous stage, to extract all the available blobs. Each blob is examined to compare its geometrical shape and contents pattern against the distinctive shape and pattern of our target. One might ask, however, why bother looking for the target's geometrical shape when it can easily be recognized from its color. The answer is that since the target can only reflect the laser beam back to the spectrometer from a range of angles from $-60^{\circ}\ to\ +60^{\circ}$, the program should only accept the target when it is being viewed from the specified angles. Furthermore, detecting the geometrical features of the target helps estimating its distance and pose with no additional hardware required.

The following are the main steps of the target recognition stage, illustrating the various operations applied to each binary image frame:

1. Using the connected-component labeling algorithm[15], the binary image is scanned for all the available white blobs. Then, each white blob is examined in steps 2 through 6 to determine whether it is a true or false target blob.
2. Any blob that has a convex hull area (Equation (9)) falling out of a range from (10×10) to (H_F×H_F) is filtered out, where H_F is the image frame height.
3. For each remaining blob, a cvBlobsLib function is called to find its bounding ellipse using ellipsoid method[16]. Since our target is square shaped, any stretched blob is filtered out from the remaining blobs (Fig. 7).



(a) (b)

Fig. 7 Filtering out stretched blobs: (a) Source image, and (b) Binary image.

4. For each remaining target blob, another cvBlobsLib function is called to find its convex hull points in a clockwise direction. Then, Equation (9) is used to find the target blob's hull area[17] from its detected hull points.

$$A_T = \frac{1}{2} \sum_{i=0}^{hull\ points-1} (x_i y_{i+1} - x_{i+1} y_i) \quad (9)$$

Where:

(x_i, y_i) and (x_{i+1}, y_{i+1}) are the coordinates of two successive hull points.

By repeating the blob extraction process applied in step (1), the interior space of the target blob is tested to find if it contains a black object (the retro-reflector blob). The hull area of any detected retro-reflector blob is then examined to determine whether it is proportional to its target blob hull area, in comparison to the proportions of our target. The formulas (10) and (11) are developed to estimate the minimum (A_{Rmin}) and maximum (A_{Rmax}) valid retro-reflector blob hull areas, based on the hull area of the current target blob (A_T).

$$A_{Rmin} = A_T * 0.15 \quad (10)$$

$$A_{Rmax} = A_T * 0.40 \quad (11)$$

Where:

A_T is the hull area of the current target blob.

5. The geometrical features of the target blob are examined to determine whether it is a convex quadrilateral shape. If it is, the validity of the target blob is confirmed and the (x, y) coordinates of its four corners are found. Because the edges of the blobs are rough and jagged, the number of detected hull points is usually more than four points, representing false target corners. This is mostly noticeable as an aliasing (staircase) effect[18,19] on the blob edges, especially when the blob is rotated around one of its axes. Furthermore, the image compression algorithm used by the camera produces quantization error to the pixels falling on the sharp edges of the image content.[20]

   To detect the true corners of a target blob, its four sides are found first. Using the law of cosines for a SSS (Side, Side, Side) triangle[21], the hull points that fall on the same line are found to form a blob side. Then, the intersection point between each two adjacent sides, using the line-line intersection technique[22], is found to be the coordinates of a true corner.

6. In this this step, the interior space of each target blob is examined, looking for a valid retro-reflector blob. A valid retro-reflector blob is the one having a black convex hexagonal shape (Fig. 8).

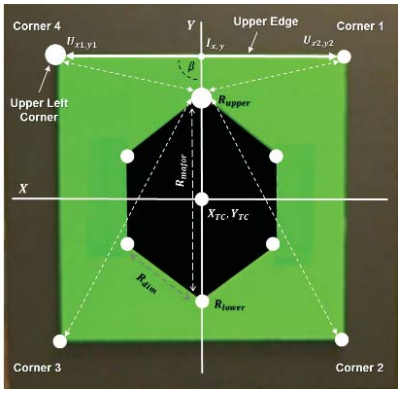


Fig. 8 The target.

4 Experimental Results

The performance of the automatic target surveying and recognition system was evaluated under different indoor and outdoor environmental conditions. The indoor tests were done while the environment was fully lit with florescent lights. The background scene was intentionally disturbed by hanging up several objects with different shapes and colors. The outdoor tests were done with the weather varying from cloudy to sunny. In each test, the program was executed, capturing video frames with a resolution of 640×480 pixels. The system was initially panned away from the remote target direction. With the help of the proposed program, the spectroscopy system started searching for the remote target and gradually turned back towards it. Eventually, the centroid of the target was located at the center of the video frame, indicating a successful laser beam to target alignment.

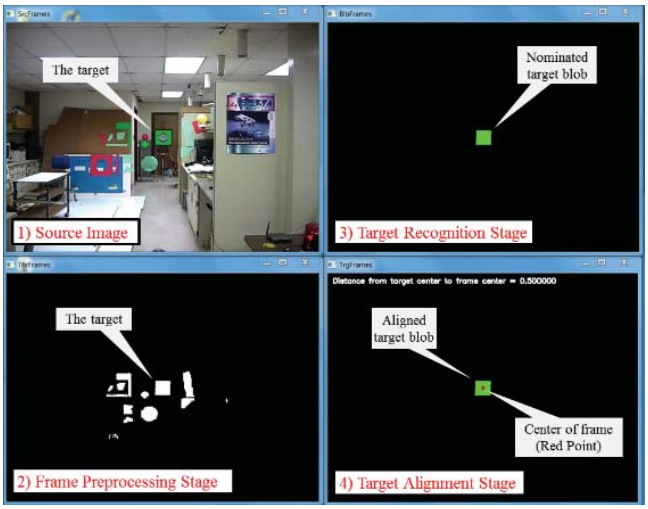
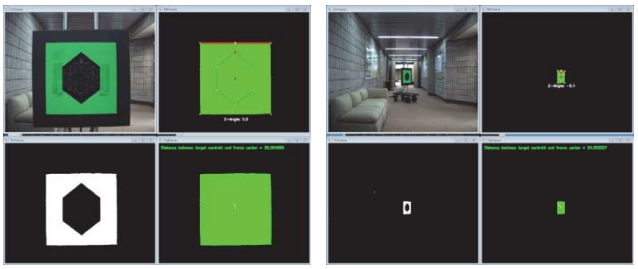


Fig. 9 Indoor test results with the target at 10 m.

Real-time images presented by the program execution were divided into four sections; representing various program execution stages (Fig. 9). The upper left section shows the source image frame captured from the camera. The lower left section corresponds to the result of the frame preprocessing stage, showing the collected green pixels

converted to a binary coded frame. The upper right section displays the target recognition stage, showing the blobs nominated as potential targets. Finally, the lower right section is specified for the target alignment stage. A red point is drawn at the center of the last section to show the location of the frame center. The position of the red point matching the centroid of the target blob means the automatic alignment process has successfully been achieved.

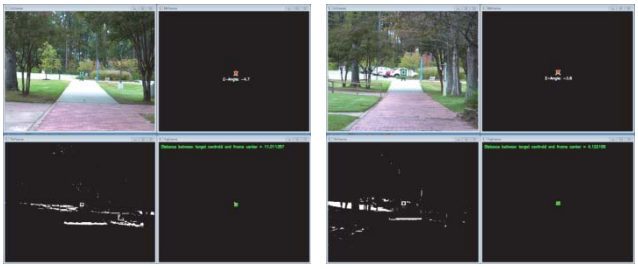


(a) (b)

Fig. 10 Indoor test results: (a) The target at 1m, front viewed, and (b) The target at 6m, 45° from y-axis.

The indoor tests were conducted with the target being placed at various distances and rotation angles. The results in Fig. 10 show that the program was able to successfully recognize the target at distances of 1 m and 6 m with a zooming level of 1x for each. The target at 1 m was in a front view (Fig. 10a); whereas at 6 m, it was deflected by 45 degrees off the y-axis (Fig. 10b).

Outdoor tests were also performed with various target distances. The remote target was recognized when the appropriate zooming level was chosen for each test and the automatic target surveying and recognition system was implemented successfully. Fig. 11 shows the results of the outdoor test at distances of 50 m and 100 m. At 50 m, the target was detected with the camera lens being zoomed-in to a zooming level of 10x (Fig. 11a). The target at 100 m, however, was detected at a zooming level of 15x (Fig. 11b).



(a) (b)

Fig. 11 Outdoor test results: (a) The target at 50 m, front viewed, and (b) The target at 100 m, front viewed.

Finally, Table 1 lists more indoor and outdoor tests results. As the table shows, the longest distance the target could be detected, when the zooming level was at 1x, was 20 m. The target was recognized even when its view pose was deflected off the camera view around any of the x, y, or z axes. When the deflection was off the x-axis, the maximum angle was ±50 degree. For the y-axis, on the other hand, the maximum angle was ±60 degrees. However, the maximum angle for the z-axis was ±90 degrees. The reason behind the limited range of deflection angles for the target to be recognized is that the perspective projection of a target image viewed from wider angles disturbs the original proportions of its retro-reflector shape. This prevents the program from detecting the retro-reflector's corners and hence recognizing its hexagonal shape correctly, and consequently, eliminating the target blob at the target recognition stage.

Table 1 Indoor and outdoor tests results.

Test no.	Test Environment	Target Pose (axis)	Target Distance (m)	Zooming Level (x)	Detection Result	Frame Rate (FPS)
1	Indoor	Plain	1	1	Passed	10
2	Indoor	Plain	6	1	Passed	12
3	Indoor	Y:45	6	1	Passed	12
4	Indoor	Z:45	7	1	Passed	12
5	Indoor	X:30	8	1	Passed	12
6	Indoor	X:50	8	1	Passed	12
7	Indoor	X:55	8	1	Failed	13
8	Indoor	Y:30	8	1	Passed	12
9	Indoor	Y:60	8	1	Passed	12
10	Indoor	Y:65	8	1	Failed	13
11	Indoor	Plain	10	1	Passed	12
12	Indoor	Plain	20	1	Passed	12
13	Indoor	Plain	21	1	Failed	14
14	Outdoor	Plain	11	1	Passed	8
15	Outdoor	Plain	30	3	Passed	8
16	Outdoor	Plain	50	9	Failed	7
17	Outdoor	Plain	50	10	Passed	8
18	Outdoor	Plain	50	20	Passed	9
19	Outdoor	Plain	100	14	Failed	7
20	Outdoor	Plain	100	15	Passed	7
21	Outdoor	Plain	100	20	Passed	7

5 Conclusions

An automatic target surveying and recognition system has been developed to automatically survey and recognize a remote, passive, and stationary target using a single monocular camera and a pan/tilt unit. The system design includes:

- Automatic zooming control: automatic control of the camera's zooming levels from 1x to 20x. This allows recognizing the target over a range of distances from 1 m to 400 m.
- Automatic target surveying: two formulas, Equations (1) and (2), have been developed to automatically drive the pan/tilt unit and adapt its panning and tilting speeds to the current zooming level of the camera. This allows automatic surveying for the target from wide panning angles ($\pm 180^{\circ}$) with a speed of 15.43 degrees/second for the zooming level of 1x, and a speed of 0.77 degrees/second for the zooming level of 20x.
- Automatic target recognition: automatically recognizing the target from various distances (1 m to 400 m), view

angles ($\pm 50^{\circ}$ for the x-coordinate, $\pm 60^{\circ}$ for the y-coordinate, and $\pm 90^{\circ}$ for the z-coordinate), and various indoor and outdoor conditions, Equations (3) to (8).

- Real-time processing: the live video was processed at a frame rate of 12 FPS for indoor environments, and 8 FPS for outdoor environments, achieving an overall speed rate of 10 FPS.

6 Acknowledgment

This work was supported by the research project "Mobile Surveying for Atmospheric and Surface Gases of Biological Origins," funded by NASA (NASA Cooperative Agreement NNX09A072A).

7 References

[1] Y. Chen, G. Anderson, S. Mahdi, and J. Khidir, "A Proposed Robotic Air Quality Monitoring System," World Automation Congress - 14th International Symposium on Robotics and Applications (ISORA), Puerto Vallarta, Mexico (2012).

[2] X. Cui, et al., "Photonic sensing of the atmosphere by absorption spectroscopy." Journal of Quantitative Spectroscopy and Radiative Transfer 113.11, 1300-1316 (2012).

[3] G. Anderson, et al., "A mobile robot system for remote measurements of ammonia vapor in the atmosphere." Systems, Man and Cybernetics, 2006. SMC'06. IEEE International Conference on. Vol. 1. IEEE (2006).

[4] A. Lilienthal, A. Loutfi, and T. Duckett, "Airborne Chemical Sensing with Mobile Robots," Sensors, vol. 6, 1616-1678 (2006).

[5] H. Xia, et al., "An approach of open-path gas sensor based on tunable diode laser absorption spectroscopy." Chinese Optics Letters 6.6, 437-440 (2008).

[6] S. Zhang, et al., "Gas leakage monitoring with scanned-wavelength direct absorption spectroscopy." Chinese Optics Letters 8.5, 443-446 (2010).

[7] G. T. Anderson, E. W. Tunstel, and E. W. Wilson, "Robot system to search for signs of life on Mars." Aerospace and Electronic Systems Magazine, IEEE 22.12, 23-30 (2007).

[8] E. W. Wilson, E. W. Tunstel, and G. T. Anderson, "BioGAS spectrometer for biogenic gas detection and location on the surface of Mars." Proceedings of the AIAA Infotech@ Aerosp Conference and Exhibition, Rohnert Park, CA. (2007).

[9] J. Khidir, Y. Chen, and G. Anderson, "Automatic Laser Beam Alignment Using Blob Detection for an Environment Monitoring Spectroscopy ", Proc. SPIE 8744, Automatic Target Recognition XXIII, 87440U (May 20, 2013).

[10] D. E. Dudgeon, and R. T. Lacoss, "An overview of automatic target recognition." The Lincoln Laboratory Journal 6.1, 3-10 (1993).

[11] G. O'Brien, "Automatic Target Recognition." Encyclopedia of Optical Engineering, Volume 1, 114-120 (2003).

[12] M. Outhouse, A. Beach, and A. Parslow, "Automatic target recognition." Proceedings of SEAS DTC Technical Conference. Vol. 2, (2010).

[13] W. Chen, Y. Q. Shi, and G. Xuan, "Identifying computer graphics using HSV color model and statistical moments of characteristic functions." Multimedia and Expo, 2007 IEEE International Conference on. IEEE, (2007).

[14] R. C. Gonzalez, and P. Wintz, Digital Image Processing, Second Edition, Addison-Wesley Publishing Company, pp. 123-124 (1987).

[15] M. B. Dillencourt, H. Samet, and M. Tamminen, "A general approach to connected-component labeling for arbitrary image representations". ACM (1992).

[16] M. Grötschel, L. Lovász, and A. Schrijver, Geometric Algorithms and Combinatorial Optimization, Springer, pp. 69-70 (1988).

[17] P. Bourke, C. Fuhrman, and A. Pons, "Calculating The Area And Centroid Of A Polygon," 1988, <http://www.seas.upenn.edu/~ese502/lab-content/extra_materials/Polygon%20Area%20and%20Centroid.pdf> (accessed 4 April 2016).

[18] N. Nain, et al., "Morphological Edge Detection and Corner Detection Algorithm Using Chain Encoding." IPCV 6, 520-525 (2006).

[19] W. Wong, "Design of a Saccadic Active Vision System," M.S. Dissertation, University of Waterloo, Department of Electrical and Computer Engineering, (2006).

[20] G. Fan, and W. Cham, "Model-based edge reconstruction for low bit-rate wavelet-compressed images." Circuits and Systems for Video Technology, IEEE Transactions on 10.1, 120-132 (2000).

[21] L. Gates, and R. Pierce, "Solving SSS Triangles," Math Is Fun. Ed. Rod Pierce. 17 May 2012, <http://www.mathsisfun.com/algebra/trig-solving-sss-triangles.html> (accessed 4 Apr 2016).

[22] Wikipedia, "Line-line intersection," <http://en.wikipedia.org/wiki/Line-line_intersection> (accessed 4 Apr 2016).

An Abstract Model of Multimodal Fusion using Fuzzy Sets to Derive Interactive Emotions

Maha A. Thafar and Arvind K. Bansal
Department of Computer Science, Kent State University , Kent, OH 44242, USA

Abstract – *Recently, social robotics applications have attracted attention, as they address profound life demands. These intelligent robotic applications should be able to perceive, recognize, and respond to human emotional states. Humans express their emotions both verbally through speech and silence and nonverbally through facial expressions and gestures. Although there are several approaches that have been proposed to recognize a limited amount of human emotion based on one modality, limited ad hoc works have been done to integrate and fuse several modalities. In this paper, we propose an abstract model of multimodal fusion of facial expressions, speech and gestures. An abstract semantic model is presented identifying emotions that involves set-theoretic operations and functional mapping. A semantic algebra has been described.*

Keywords – Abstract model, social robotics, emotion recognition, interaction, multimodal fusion

1 Introduction

In recent years, there has been emphasis on social robotics [17] - a field of robotics that studies interaction of automated intelligent machines with humans to improve elderly care, education, personal companionship, and home care. In order to improve their acceptability, these machines and systems should be capable of understanding human behaviors and emotions, and respond appropriately.

Emotion is a language for communicating feelings, including pain, to others. Humans express a myriad of emotions based on their state of mind and partially regulated by personality, context, situation and cultural conditioning.

There are many emotions such as happiness, anger, embarrassment, contempt, surprise, discomfort, guilt, shame, rage, disappointment, confusion, elation, depression, etc. Emotion analysis has attracted scientists from many disciplines such as psychology, cognitive science and computer science. Several psychological theories have been proposed [11] to understand the origin and expression mechanisms of emotions.

Interactive emotions are a subclass of emotions that humans use to interact with each other in close proximity. Interactive emotions are expressed using multiple modes such as facial expressions, utterances, silence, different attributes of speech such as energy level, variations in phoneme and syllable durations, emotion-specific phrases, gestures that include posture and movement of various body parts and the intensity of the motion.

Currently, computational systems have been limited mostly to analyzing a single mode of emotion expression such as facial expression analysis [1], speech analysis [6], gesture analysis [14] and (to some limited extent) ad hoc multimodal analysis [5, 6, 7, 8]. Even a computational analysis of facial expressions is limited to six basic emotions: *happiness, sadness, surprise, anger, disgust*, and *fear*. There are many issues with current multimodal analysis systems such as: 1) lack of detailed facial expression analysis; 2) lack of detailed gesture-catalog based upon body-posture, motions of different body-parts and their correlation; 3) lack of catalog of phrases associated with different emotions; 4) lack of integration of contextual information such as situation, gender, culture etc.; and 5) lack of a formal model to combine multiple modes such as facial expressions, speech analysis and gestures.

There can be multiple models of multimodal fusion such as: 1) probabilistic models where probability is derived from each mode, and combined to enhance the probability from multiple modes; 2) dynamic weight model where the evidence from each mode is combined using a dynamic weight that changes with context; and 3) a language based semantic algebra model where each mode becomes a domain, and different domains are mathematically mapped to different emotions.

In this research, we develop an abstract model based upon semantic algebra where facial expressions, speech feature-vectors, and gesture parameters become three different domains, and interactive emotion is recognized using mathematical mapping of values in these domains to an identifiable emotion. After all, emotion is a language of communicating feelings. Currently, to the best of our knowledge, there has been no effort to develop such an abstract semantic model for modeling emotions. We have also developed an extensive catalog of gestures related to interactive emotions and algorithms [18] for gesture recognition using Kinect skeletal and image analysis [5]. However, the description of gesture-catalogs and algorithms for implementation is outside of the scope of this paper due to page limits, and will be presented elsewhere. Our study is limited only to upper body-parts involved in interactive emotions during conversation. Although we do not consider limb movement below torso, the model can be easily extended to incorporate other body-parts.

This paper is organized as follows. Section 2 describes psychological theories of emotions, mathematical definitions about sets and notations. Section 3 describes components of perceived behavior. Section 4 describes parameterization of

sensor-values using fuzzy sets. Section 5 describes a semantic mapping of perceived behavior to interactive emotions. Section 6 provides related work. Section 7 concludes the paper, and gives the future directions.

2 Background and definitions

There are three popular psychological theories of emotions: *James-Lange theory* [11], *Cannon-bard theory* [11] and *Schater-singer* theory [11]. *James-Lange* theory is based upon somatic feedback. The theory states that the body reacts to stimuli, the brain registers these physical reactions, and the resulting mental state is known as *emotion*. Different emotions are the perceptions of humans on different bodily reactions [11]. *Cannon-Bard* theory states that the "Autonomic Nervous System" (ANS) is activated in many emotions in anticipation rather than as a reaction to an action. The theory allows for the same bodily reactions for different emotions. For instance, increase of heartbeat occurs during anger as well as excitement [11]. *Schachter-Singer* theory states that encountering an emotion requires both a bodily response and an interpretation of the bodily response by considering the specific circumstance the individual at a specific moment [11].

Facial expressions occur both as an involuntary emotional response or voluntary social communication [1]. *Facial action coding system (FACS)* is a quantitative anatomical approach to simulate facial-expressions based on the simulation of a combination of facial muscle movements [9]. The unit of FACS is *Action Unit (AU)* that involves a segment of a muscle involved in facial expression.

Speech analysis has multiple features such as *phonemes, utterances, amplitude, frequency, pitch, quantile,* and *sound-energy*. *Phonemes* are the basic units of speech. *Pitch* consists of high-numbered harmonics that are weaker in quality than stimuli consisting of low-numbered harmonics [15]. *Rhythm* is defined as the rate of speech, and *quantile* is defined as the ratio between voiced and unvoiced frames. *Utterance* is a chain of phrases, and *sound energy* consists of changes in *pitch* and *rhythm* within a particular context [3]. During emotional interaction, pitch, amplitude, sound energy, duration of silence and utterances change significantly, and act as parameters for the recognition of interactive emotions.

Gesture [14] is a nonverbal communication using perceptible bodily actions such as body-postures and body-part movements including movements of the head, torso, hands, face and eye. *Body Posture* [8] is the orientation of the body during human interaction. Movements of body-parts is further divided into different types of movements. For example, head-movement has three degrees of freedom: 1) tilting - right or left; 2) *yaw* – rotating sideways; or 3) *pitch* – moving upward and backward [16].

Facial expressions combined with speech and gestures reduces ambiguities in deriving interactive emotions [6].

2.1 Set-theoretic definitions

A *tuple* is a collection of one or more fields that together form an element [4]. An element in a nested-tuple may be a nested-tuple, a tuple, or a singleton element. A *Cartesian-product* takes *N* sets ($N > 1$), and returns a set of N-tuples such that ith-field of an element is a member of the ith set as shown:

$$X_1 \times \dots \times X_n = \{ (x_1, \dots, x_n) | x_i \in X_i \; for \; all \; i = 1, \dots, n \} \quad (1)$$

A *finite-mapping* maps domain-element to one of the elements in the codomain. A function is a either one-to-one or many-to-one mapping. Given two sets A and B, the application of a function is denoted as *f*: A → B. Two domains can be joined using: 1) product-domain that uses Cartesian product A × B; or 2) sum-domain that uses disjoint-union A + B; 3) f*unction Domain* f: A → B; and 4) *Lifted Domain* $A_\perp$. Lifted domain $A_\perp$ of a set A contains a bottom element denoted by $\perp^A$ to make the functional mapping from domain to codomain *well-defined*. We also use top symbol "⊺" from lattice theory to take care of cases when a mapping is true for all elements of a constituent set in a Cartesian-product in semantic algebra for multimodal fusion. In that case, the presence of that set in the Cartesian-product becomes redundant.

2.2 New definitions and notations

This subsection describes new definitions related to abstract modeling of emotions using semantic algebra. Given lifted domains $A_\perp$ and $B_\perp$, Cartesian product is defined as $A_\perp \times B_\perp$. The mapping of Cartesian product on a lifted-codomain $C_\perp$ is defined as $A_\perp \times B_\perp \rightarrow C_\perp$. If a function maps a proper element of (a ∈ A and b ∈ B) to c ∈ C, then it is well defined. If either of the field in the pair (a, b) ∈ $(A_\perp \times B_\perp)$ is a bottom element, then it maps to $\perp^C$. If the mapping does not change for every element in the other set, then the other set does not play a role in mapping and therefore can be removed for the specific mapping.

$$(a \in A_\perp, b \in B_\perp) \rightarrow c \in C_\perp \text{ if } (a \in A, b \in B) \rightarrow c \in C \quad (2)$$

$$(a \in A_\perp, b \in B_\perp) \rightarrow \perp^C \text{ if } (a, b) \text{ does not map on } c \in C \quad (3)$$

$$(a \in A_\perp, b \in B_\perp) \rightarrow \perp^C \text{ if } (a == \perp^A) \vee (b == \perp^B) \quad (4)$$

$$(a \in A_\perp, \top) \rightarrow c \in C_\perp \text{ if } (a \in A, b \in B) \rightarrow c \text{ for all } b \in B \quad (5)$$

$$(\top, b \in B_\perp) \rightarrow c \in C_\perp \text{ if } (a \in A, b \in B) \rightarrow c \text{ for all } a \in A \quad (6)$$

The semantic algebra incorporates threshold values to filter out noise in sensor-values. Threshold values are donated by a symbol τ^A for the set A and τ^B for the set B. A sensor-value for an attribute is present if the sensor-value is greater than the corresponding threshold value (possibly zero depending upon the set A and B); otherwise the element is treated as absent/ill-defined. Thus the equations (2) and (4) are modified to include the threshold values as shown:

$$(a \in A_\perp, b \in B_\perp) \rightarrow c \in C_\perp \text{ if } (a \in A, b \in B) \rightarrow c \in C \wedge \text{value}(a \in A) \geq \tau^A \wedge \text{value}(b \in B) \geq \tau^B \quad (7)$$

$$(a \in A, b \in B) \rightarrow \perp^C \text{ if } (a = \perp^A) \vee (b = \perp^B) \vee \text{value}(a) < \tau^A \vee \text{value}(b) < \tau^B \quad (8)$$

An *ordered fuzzy-set* has sorted values such as {*1, 2, 3, 4*}. A *discrete fuzzy-set* has discrete values that cannot be ordered. For example, the *fuzzy-sets {straight, left-tilted, right-tilted, forward-bending, backward-bending}* modeling *body-*

posture is a *discrete fuzzy-set*. However, the set {*none*, *somewhat*, *midway*, *completely*} is an *ordered fuzzy-set* since the values can be mapped to 0..3: "none" → 0 and "completely" → 3. Inequality and value-precedence operators are applied on ordered fuzzy-sets for comparison.

Some of the notations used in this paper are as follows: E denotes the set of interactive emotions; ε denotes an element in the set E; $\wp$ denotes the set of perceived behaviors; π denotes one perceived behavior in $\wp$; $\mathbb{Z}$ denotes the integer domain, intensity domain and ordered fuzzy-set; $\mathbb{B}$ denotes Boolean domain and effectiveness; $\mathbb{R}$ denotes real number domain; Γ denotes gesture-domain; γ denotes an element in Γ; Σ denotes the facial-expression domain; σ denotes one facial expression in Σ; Ψ denotes the speech domain; ψ denotes an element in the set Ψ; $\perp^A$ denotes the bottom element in the set A.

3. Interactive emotion components

Interactive emotion involves two or more actors. In this study, one of the actors is human, and the other actor is an intelligent machine (or software system) capable of perceiving emotions through various sensors such as a microphone, camera, and software for speech and image analysis. A *perceived behavior* is the perception of sensor-values of various modes: facial-expression, speech and gesture. Different sensors provide different aspects of perceived behavior: image analysis provides facial expressions and gestures; speech analysis provides speech and silence gaps related information. Speech and body-motion is quantized to different fuzzy values to reduce computational complexity in deriving perceived behavior. *Intensity* of sensor-values facilitates finer classification of emotions. For example, *upset, anger, rage,* and *violence* show different intensity levels for the same basic emotion "anger". *Effectiveness* such as presence of tears also facilitates finer classification of interactive emotions [13]. For example, the presence of tears emphasize *sadness*.

Interactive emotion is also regulated by context such as *personality-type, age, gender, situation* and *culture*. Some of these features can be handled by having multiple knowledge bases for varying contexts. However, the current study is limited to the fusion of facial-expression, speech and gestures, their intensities and effectiveness.

The overall analysis of interactive emotion is described in Figure 1. Perceived behavior, intensity and effectiveness have been placed at the same level. Body-gestures, facial-expressions and speech utterance are shown as components of perceived behavior. The reasons for showing intensity and effectiveness at the same level as perceived behavior are: 1) All three components of perceived behavior exhibit intensities; 2) *intensity* is more global in nature; 3) *Effectiveness* is not part of any of the three components; and 4) *basic emotions* can be derived by fusing components of perceived behaviors.

3.1 Classification of perceived behaviors

There are three types of emotions: 1) basic emotions; 2) derived or mixed emotions; and 3) finer subclasses of emotions. *Derived emotions* are mixtures of two emotions. For example, "amazement" is a mixture of "surprise" and "happiness". Finer subclasses of emotions have the same basic emotion with varying intensity. For example, humans can be *elated, excited, happy, joyful*, and *relieved*. All these emotions have the same basic emotion "happiness" but different intensity.

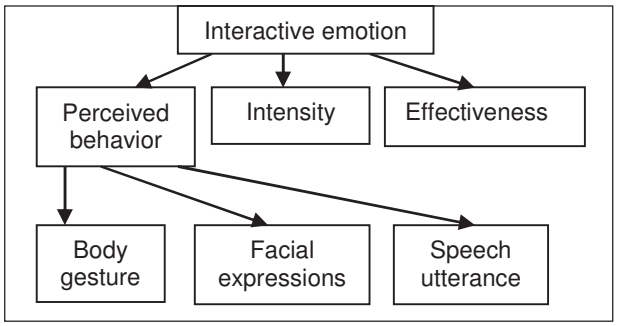


Figure 1. Components of perceived behavior

Gestures are classified into two major sub-classes: 1) postures and 2) motion. These subclasses are further divided into various body-parts such as head, shoulder, arm, torso, wrist and eyes. Posture and motion cues are correlated with various emotions and their intensities [8]. For instance, fast and jerky motions are correlated with high intensity emotions such as excitement or rage; very slow and smooth movements show sadness or depression. Movements of head and hand identify the intensity levels of emotions. The feature vectors for motion are speed, acceleration, frequency, abruptness and orientation.

Speech features are also used to recognize basic emotions. For example, a trembling voice expresses "fear"; a high pitch or tone expresses "anger" or "frustration". Longer periods of silence are associated with "sadness" or "approval" or "guilt"; a loud "no" is associated with "anger". There are phrases and speech-patterns associated with specific emotions.

For accurate identification of an emotion, a combination of two or more components and intensity levels are required. Most complex emotions, derived emotions, and emotions with extreme intensity levels are not exhibited by facial expression alone. Facial expressions, body gestures and postures, and speech features including utterances and silence must be integrated to model interactive-emotions.

Interactive-emotions are mathematically modeled as nested sets. Facial-expressions are divided into basic and derived expressions; speech is divided into utterances, phrases, and silence; and gestures are divided *into head-orientation, head-motion, posture, hand-position, hand-motion, finger-patterns*, and *finger-motion*. An *interactive-emotion* may not have values for all the components; and some of these values may not contribute to the specific emotion.

4 Fuzzy parameterization

Before mapping perceived behavior to emotions, sensor values have to be mapped to fuzzy domain [19]. The use of fuzzy domains significantly reduces number of possibilities improving the computational efficiency [12]. There are six types of parameters: 1) *facial-expressions parameters*; 2) *speech parameters*; 3) *body-posture parameters*; 4) *body-movement parameters*; 5) *intensity values*; and 6) *effectiveness*. The posture data and facial expression data are analyzed using

static images (or image frames in video clips); and speech and gesture analysis needs analysis of video-clips.

4.1 Facial-expressions parameterization

The key geometric features for facial expression analysis [9] are: *forehead, eyebrows, eye-shapes, inner space between eyes, nose, cheeks, mouth* and *lips*. The *forehead* is modeled as a discrete-fuzzy-set {*relaxed, furrowed*}. *Eyebrow* is modeled as a discrete fuzzy-set {*relaxed, curved upwards, curved-in*}. *Eye-shape* is modeled as an ordered-fuzzy-set {*partially-closed, fully-closed, relaxed, round, popping*}. Inner space between eyes is modeled as a discrete fuzzy-set {*relaxed, wrinkled*}. *Lips-shape* is modeled as a discrete-fuzzy-set {*relaxed, ajar, pinched, stretched, circular*}. *Lip-motion* is modeled as an ordered fuzzy-set {*no-motion*, *vibrating*, *opening-and-closing*}. *Cheeks* are modeled as a discrete fuzzy-set {*relaxed, stretched, pulled-back*}. *Nose* is modeled as a discrete-fuzzy-set {*wrinkled, big-nostrils, twitching, relaxed*}. Many of these values are strongly correlated, and many tuples have no emotional meaning, and map to the bottom-element. For example, a furrowed forehead and curved-eyebrows are highly correlated.

4.2 Speech parameterization

Speech is a summation of two domains: *utterances* and *silence*. Utterance has four subcomponents: *emotional-phrases, frequency-pattern, sound-intensity,* and *sound-quality*. Many times *emotional-phrases* have one-to-one mapping with emotion, and prune the search space. For example, the emotional-phrase "oh" may mean "disappointment," "surprise," or "unexpectedness". The *sound-intensity* is a triple (*pitch, loudness, duration)*. Increase in loudness and pitch shows excessive intensity level. *Sound-quality* is modeled as a triple (*rhythm-change*, *word-distortion*, *quantile*). *Rhythm* is defined as the speaking rate, and quantile is defined as *ratio between voiced and unvoiced frames.* The domains including pitch, loudness, rhythm, duration, and quantile are defined using a fuzzy set value that is relative to individual normal speech-behavior, depending on *age*, *gender*, *culture* and *personal traits*.

Silence plays an important role in emotion recognition. Excessive silence indicates "disapproval" or "suppression of emotion". However, the interpretation of silence depends upon the situation and context. Silence is parameterized by the fuzzy-set of duration.

4.3 Posture parameterization

Posture can be *body-posture*, *head-posture*, *hand-posture and gaze*. Body posture is characterized by two parameters: *body-orientation* and *body-envelope*. The parameter *body-orientation* is a value in a discrete fuzzy-set {*straight, left-tilted, right-tilted, forward-bending, backward-bending}*. The parameter *body-envelope* identifies the area of the encompassing rectangle or ellipse, and is modeled by an ordered fuzzy set { *contracted, regular, expanded*}.

Head-posture is modeled as (*head-orientation, head-tilt). Head-orientation* is described as a pair (*head-direction*, *degree*). The parameter *head-direction* takes a value from a discrete fuzzy-set {*left, right, up, down*}. The parameter *degree* takes a value from an ordered fuzzy-set {*somewhat, little, medium, completely*}. The parameter *head-tilt* takes a value from an ordered fuzzy-set {*none, somewhat, lot*}.

Hand posture uses three attributes: *left-hand-posture*, *right-hand-posture*, and *proximity*. The parameters *left-hand-posture* and *right-hand-posture* show the *orientation* and the *shape* of the hands. The parameter *proximity* describes the proximity of the fingertips to different parts of the upper body such as other hand, head, nose, mouth, etc. The parameter *proximity* takes a value from the discrete fuzzy set *{palm-touching, fingers-touching, head-touching, face-touching, eye-touching, nose-touching, separate, stretched}*. The posture for left and right hands is given by four sub-parameters: *shoulder-posture, arm-posture*, *palm-posture* and *finger-posture. Shoulder-posture* takes a value from the discrete fuzzy-set *{relaxed, dropped-down, shrugged-up}*. The parameter *arm-posture* is defined as a pair (*arm-orientation, arm-shape*). The parameter *arm-orientation* is defined as a pair of *orientation-degree* and *direction*. *Orientation-degree* is an ordered-fuzzy-set *{none, somewhat, full}*. *Arm-direction* takes a value from the discrete-fuzzy-set *{down, up, sideways, front, back}*.

Many times, palm and finger postures are combined to give a joint posture. For example, *thumbs–up* expresses "happiness" or "encouragement", and involves a combination of palm and fingers. Finger posture has been integrated with palm posture to provide palm-shape. A palm is modeled as a pair of the parameters *palm-shape* and *palm-orientation*. *Palm-shape* takes a value from the discrete fuzzy-set *{fist, open-palm, thumb-up, thumb-down, index-up, index-pointing, victory, perfect}*. *Palm- orientation* is the direction of the palm, and takes a value from the discrete fuzzy set *{bent-backward, bent-forward, facing-up, facing-down}*.

Gaze is modeled as a triple: *(left-eye, right-eye, duration). Eye* is modeled as a triple: *(iris-position, pupil-shape, eye-shape).* The parameter *iris-position* takes a value from a discrete fuzzy-set {*inward, outward, up, down, middle, invisible}*. The parameter *pupil-shape* takes a value from a discrete fuzzy-set {*constricted*, *normal, dilated*}. The parameter *eye-shape* takes value from an ordered fuzzy-set {*closed, narrow, normal, round, popping}*.

4.4 Motion parameterization

Motion of upper-body parts involves: *head-motion*, *shoulder-motion*, *arm-motion*, *hand-motion*, and *eye-motion*. These body-motions are extracted by video-analysis and skeletal analysis using Kinect [5]. Five major features describe body-movement as follows: 1) *motion speed*; 2) *motion frequency*; 3) *motion direction (Angle)*; 4) *motion attack*; and 5) *motion relaxation*. *Motion-speed* is the displacement in the unit time between the start-point and the end-point of a motion. *Motion attack* is defined as the rate-change in speed from no motion to peak speed. Conversely, *motion-relaxation* is described as the rate-change in speed to move from peak speed to no motion. *Motion-frequency* is used for repeated actions such as nodding of head, eye-blinking etc. *Motion-direction* is attached to different types of gestures, and depends upon the degrees of freedom of limbs.

Head-movement is modeled as (*start-location*, *start-posture*, *end-location*, *end-posture*, *head-motion-type*, *motion-speed, motion-frequency, motion-direction, motion-attack, motion-relaxation*). The parameters *start-location* and *end-location* take a value from the fuzzy set {*left, right, up, down, tilted-left, tilted-right*}. The parameter *head-motion-type* takes a value from the fuzzy set {*vertical, horizontal, tilt*}. The parameter *motion-direction* takes a value from the fuzzy set {*up-down, left-right, right-left, tilt-left, tilt-right*}. The parameters *motion-speed*, *motion-frequency*, *motion-attack,* and *motion-relaxation* take a value from the ordered fuzzy-set {*none, low, moderate, high, very-high*}.

Hand movement has three components: *arm, palm* and *fingers*. Arm-movement and palm-movement are modeled similar to head-movement except that the fuzzy sets for start-points, end-points and motion-type are different. In addition, some *palm-motions* are correlated to *finger-motions* and *arm-motions.* The arm-*motion-direction* takes a value from the discrete fuzzy-set {*up-down, left-right, right-up, right-down, left-up, left-down, front-back*}. Moving both hands simultaneously shows more emotions. Motion speed, frequency, attack, and relaxation take values from an ordered fuzzy-set {*none, low, moderate, high, very-high*}.

Fingers move in two ways: 1) *flexing*, such as bending, making a fist, gripping, grasping, and folding of the fingers; and 2) *extension*, such as straightening, pointing, stretching, and spreading out of the fingers [10]. The *movement-quality* itself can be *jerky* or *smooth*. The jerky movement of fingers with high speed and high frequency indicates the person is "agitated" or "nervous". *Finger-movement* is modeled as a 6-tuple: *(start-state, end-state, movement-type, movement-quality, frequency, sync-type)***.** A *finger-state* takes a value from an ordered fuzzy set {*straight, slightly-flexed*, *significantly-flexed*, and *completely-flexed*}.

Eye-movement is modeled as a quadruple: (*start-state, end-state, motion-duration, blink-frequency*). *Start-state* and *end-state* take values from the discrete fuzzy set for iris-position: {*inward, outward, up, down, middle, invisible*}. The motion-duration takes a value from an ordered fuzzy-set {none, *short, medium, long*}. The parameter blinking frequency takes a value from an ordered fuzzy set {*fixed, slow, normal, rapid*}.

5 Mapping perceived behavior

All fuzzy set of parameters are included in various domains. The major domains are *facial-expression domain, speech domain, posture domain, motion domain, intensity domain* and *effectiveness domain.* Many classes of perceived behaviors mapping into a single emotion are grouped into various subclasses. The subclasses are defined using inequality-tests on parameter intensities: *less-than* a threshold value, *greater-than* a threshold-value, or a range-test between *lower and upper thresholds*. Parameter intensities are modeled as an *ordered fuzzy set*. Intensity domain is modeled as integer domain $\mathbb{Z}$, and effectiveness domain is modeled as Boolean domain $\mathbb{B}$. Various sub-domains used in the semantic algebra are given in Figure 2.

The overall semantic algebra is given in Figure 3. It is described as a combination of product-domain (Cartesian-product) and summation-domain (disjoint-union) of constituent domain functionally mapping on the codomain.

*Facial-expression***:** product-domain
Ordered-sets: *eye-shape, lip-movement,*
Discrete-sets: *eyebrow, nose, cheek,*
Boolean sets: forehead, inner-eye-space

Speech domain: summation and product-domain
Ordered-sets: rhythm, quantile, pitch, loudness, silence, duration
Discrete-sets: emotional-phrases

Body-posture domain: product-domain
Ordered-sets: head-motion-type, head-direction, degree, eye-shape, pupil-shape, shoulder-posture, arm-orientation-degree, arm-shape-degree.

Discrete-sets: envelope-area, iris-position, pupil-shape, arm-shape-type, arm-orientation, hand-proximity, palm-shape, palm-direction

Body-motion domain: product-domain
Ordered-sets: head-speed, head-attack, head-relaxation, head-frequency, hand-speed, hand-attack, hand-relaxation, hand-frequency, finger-state
Discrete-sets: head-start-location, head-end-location, head-motion-type, hands-sync-type, arm-start-location, arm-end-location, arm-motion-direction, finger-movement-type, blinking-frequency

Intensity domain: $\mathbb{Z}$

Effectiveness domain: $\mathbb{B}$

Figure 2. Domain classifications

The overall mapping from the three components to perceived behavior is illustrated by equation in Figure 4. Interactive emotion is derived by mapping the Cartesian product of the lifted domains of perceived behavior, intensity level and effectiveness to the lifted domain of interactive emotion $E_\perp$ as follows: $\wp_\perp \times \mathbb{Z}_\perp \times \mathbb{B}_\perp \rightarrow E_\perp$ where $\wp$ is domain of well-defined perceived behaviors, $\mathbb{Z}$ is the domain of integers denoting fuzzy values of intensity levels, $\mathbb{B}$ is the truth domain showing presence/absence of tears.

The domain $\wp_\perp \times \mathbb{Z}_\perp \times \mathbb{B}_\perp$ contains the bottom element due to two reasons: 1) the presence of undefined value in the perceived behavior due to undefined or ill-defined value of the subcomponents: facial expression, speech, and/or gesture; and 2) the presence of non-correlating values in the perceived behavior triple (σ, ψ, γ) which do not map to any $\varepsilon \in E$. Another class of mapping is when a particular component does not play any role in interpreting emotion. In that case, we use top-symbol $\top$ *(top)* for the field corresponding to that component. For example, if a gesture $(\gamma \in \Gamma)$ uniquely maps to an emotion $\varepsilon \in E$ for all values of facial expressions and speech value then mapping is denoted as $(\gamma, \top, \top) \rightarrow \varepsilon$. The overall mapping of perceived behavior to interactive emotion is shown in Figure 5.

All these mappings are kept in a knowledge base, and efficient hashing function based scheme has been used to combine derived fuzzy values to retrieve the corresponding interactive emotions.

Semantic algebra for facial-expressions
Forehead × eye-shape × eyebrow × inner-eye-space × nose × cheek × lip-state → facial-expression

Semantic algebra for speech
Utterance = Emotional-phrases × intensity × quality
intensity = pitch × loudness × duration
quality = rhythm × quantile × duration
silence = {no-sound} × duration
Utterance + silence → speech

Semantic Algebra for gestures
Posture × body-motion → gestures
Body-posture × head-posture × hand-posture × gaze →Posture
body-orientation × body-shape → body-posture
head-direction × head-tilt → head-posture
shoulder-posture × arm-posture × palm-posture× finger-posture → hand-posture
arm-orientation × arm-shape → arm-posture
shape-degree × shape-type → arm-shape
palm-orientation × palm-shape → palm-posture
eye-orientation × duration → gaze
eye-shape × pupil-shape × iris-position → eye-orientation
head-motion × shoulder-motion × joint-hand-motion × eye-motion → body-motion
head-posture × head-speed × head-attack × head-relaxation × head-frequency × head-direction → head-motion
shoulder-posture × shoulder-direction → shoulder-motion
arm-motion × palm-direction × finger-motion → hand-motion
eye-orientation × motion-duration × blinking-freq. → eye-motion
arm-motion × arm-motion × sync → joint-arm-motion
arm-posture × arm-speed × arm-attack × arm-relaxation × arm-frequency × arm-direction → arm-motion
finger-state × finger-movement-type × finger-movement-quality× frequency × sync-type → finger-movement

Figure 3. Semantic algebra for multimodal fusion

$(\sigma \in \Sigma, \psi \in \Psi, \gamma \in \Gamma) \rightarrow \pi \in \wp$ (9)
$(\sigma \in \Sigma_\perp, \perp^\Psi, \gamma \in \Gamma_\perp \rightarrow \perp^\wp$ (10)
$(\sigma \in \Sigma_\perp, \psi \in \Psi_\perp, \perp^\Gamma) \rightarrow \perp^\wp$ (11)
$(\perp^\Sigma, \psi \in \Psi_\perp, \gamma \in \Gamma_\perp) \rightarrow \perp^\wp$ (12)
$(\sigma \in \Sigma, \psi \in \Psi) \rightarrow \pi \in \wp$ if $(\sigma \in \Sigma, \psi \in \Psi, \top) \rightarrow \pi \in \wp$ (13)
$(\sigma \in \Sigma, \gamma \in \Gamma) \rightarrow \pi \in \wp$ if $(\sigma \in \Sigma, \top, \gamma \in \Gamma) \rightarrow \pi \in \wp$ (14)
$(\psi \in \Psi, \gamma \in \Gamma) \rightarrow \pi \in \wp$ if $(\top, \psi \in \Psi, \gamma \in \Gamma) \rightarrow \pi \in \wp$ (15)
$(\sigma \in \Sigma) \rightarrow \pi \in \wp$ if $(\sigma \in \Sigma, \top, \top) \rightarrow \pi \in \wp$ (16)
$(\psi \in \Psi) \rightarrow \pi \in \wp$ if $(\top, \psi \in \Psi, \top) \rightarrow \pi \in \wp$ (17)
$(\gamma \in \Gamma) \rightarrow \pi \in \wp$ if $(\top, \top, \gamma \in \Gamma) \rightarrow \pi \in \wp$ (18)

Figure 4. Mapping components to perceived behavior

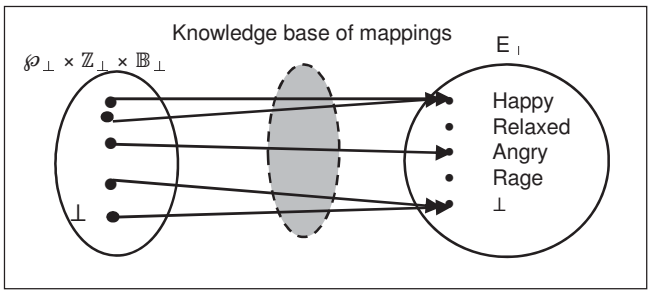


Figure 5. Schematics of perceived behavior to emotion

6 Related works

There are recent research efforts related to the emotion recognition using multimodal fusion [5, 6, 7]. However, despite the understandings facial expression analysis and speech analysis, multimodal fusion to identify emotions are limited to adhoc identification of some basic emotions.

Castellano et al. have used non-propositional movement qualities to distinguish between emotions [6]. Their study used five motion cues: 1) quantity of motion (QoM); 2) contraction index of the body (CI); 3) velocity; 4) acceleration; and 5) fluidity of hands to characterize gestures. Gestures were described by the profiles over the time of expressive motion cues [6]. The results of the study indicated that expressive motions can discriminate between different arousal states and between positive and negative emotions. However, the study limits its scope to four emotional states – *anger, joy, pleasure,* and *sadness*.

Castellano et al. [7] extended their work to multimodal framework integrating face-expression, body-gestures, and speech. A Bayesian classifier was used for feature level fusion and decision level fusion. A comparison between unimodal, bimodal, and multimodal classification showed that multimodal classification is better. This study showed that statistically derived features from each mode are not sufficient. There is additional need to identify features that are relevant to dynamics of expressive emotions. However, their study is ad hoc, and is limited to identifying eight emotions [7].

Bevilacqua et al. [5] traced upper body-parts to predict human emotions. These gestural actions, acquired using Kinect sensors [5], were mapped to seven common human emotions – *happiness, anxiety, anger, boredom, sadness, uncertainty*, and *surprise* using "Facial Action Units" (FAU). For this study, 17 FAU were defined, such as "Left Hand touching the Head", "Head tilting to the Right", "Hands Up", and "Arms Up". An emotion may correspond to one or more combinations of these action units. The results showed high scores on precision and recall parameters in both seated and upright gestures. Despite sadness and surprise being classified as indistinguishable, the overall accuracy was 80%. This study missed several important motion-features such as speed and frequency that contribute significantly to emotional intensity, and thus was incapable to separate between finer subclasses of the same basic emotions [5].

Our work describes a formal abstract model. In our study, a new way of abstract modeling of interactive emotions has been proposed. It is based upon the understanding that interactive emotions are a language to express our feelings using verbal and non-verbal cues. The integration of gesture with other modalities enhances recognition of finer subclasses of emotions and derived emotions. The use of a fuzzy classification prunes the search space significantly.

7 Conclusion and future work

In this research, we treat interactive emotions as a language to communicate feelings. We have developed a formal abstract model and semantic algebra to combine three domains: facial-expressions, speech, and gestures (both postures and upper body-motions) to map perceived behavior

to interactive emotion. A fuzzy set parameterization of postures, motion intensity, and effectiveness have been incorporated for finer classification of emotion, disambiguation of emotion, and the derivation of secondary emotions. The use of fuzzy domains significantly prunes the search space and reduces the variations caused by sensor inaccuracies and inconsistencies. The parameterization of the sensor values and archiving them in a declarative knowledge base allows for learning and dynamic update of emotional mapping. No other system has performed a formal abstract modeling of emotion based upon the semantic domains.

Based on this model, we are developing algorithms and implementation to fuse three domains to identify different shades of human emotions that are more elaborate than derived by the currently available system that derive limited set of basic emotions. We are also studying other probability and dynamic weight based fusion of different domains of emotions and efficient indexing scheme for real-time emotion recognition. We are also developing a knowledge base of gesture trajectories that can be combined with facial expressions to refine emotion recognition during real-time interaction. These trajectories will be put into an XML-based declarative knowledge bases. Based upon person's identification and cultural identification, the system will switch dynamically between different domain-specific knowledge bases to derive and generate emotions accurately. The use of declarative knowledge base will make the emotion recognition system adaptive to different personalities and contexts.

8 References

[1] R. Adolphs. "Recognizing emotion from facial expressions: psychological and neurological mechanisms"; Behav. Cogn. Neurosci. Rev., Vol. No. 1, 21-62, 2002.

[2] I. Bacivarov, M. Ionita and P. Corcoran. "Statistical models of appearance for eye tracking and eye-blink detection and measurement"; IEEE Transactions On Consumer Electronics, Vol. No. 54, 1312-1320, 2008.

[3] M. M. Bakhtin. "Speech Genres and Other Late Essays". University of Texas Press, 2010.

[4] A. K. Bansal. "Introduction to Programming Languages". CRC Press, 2013.

[5] V. Bevilacqua, D. Barone, F. Cipriani, G. D'Onghia, G. Mastrandrea, G. Mastronardi et. al. "A new tool for gestural action recognition to support decisions in emotional framework;" Proceedings of the IEEE Symposium of Innovations in Intelligent Systems and Applications (INISTA), 184-191, 2014.

[6] G. Caridakis, G. Castellano, L. Kessous, A. Raouzaiou, L. Malatesta, et el. "Multimodal emotion recognition from expressive faces, body gestures and speech"; Artificial Intelligence and Innovations: From Theory to Applications, Springer Berlin Heidelberg, 375-388, 2007.

[7] G. Castellano, S. D. Villalba and A. Camurri. "Recognizing human emotions from body movement and gesture dynamics"; Affective Computing and Intelligent Interaction, Springer Berlin Heidelberg, 71-82, 2007.

[8] M. Coulson. "Attributing emotion to static body postures: Recognition accuracy, confusions, and viewpoint dependence"; Journal of Nonverbal Behavior, Vol. No. 28, 117-139, 2004.

[9] M. Ghayoumi and A. K. Bansal. "Unifying Geometric Features and Facial Action Units for Improved Performance of Facial Expression Analysis"; Proceedings of the International Conference on Circuits, Systems, Signal Processing, Communications and Computers (CSSCC 15), Recent Advances in Electrical Engineering Series, ISBN: 978-1-61804-285-9 , Vienna, Austria, 259-266, 2015.

[10] E. Holden, R. Owens and G. G. Roy. "Hand movement classification using an adaptive fuzzy expert system"; International Journal of Expert Systems Research and Applications, Vol. No. 9, 465-480, 1996.

[11] R. Plutchik and H. Kellerman. "Theories of Emotion". Academic Press, 2013.

[12] C. H. Lim, E. Vats and C. S. Chan. "Fuzzy human motion analysis: A review"; Pattern Recognition, Vol. No. 48, 1773-1796, 2015.

[13] C. M. de Melo and J. Gratch. "Expression of emotions using wrinkles, blushing, sweating and tears"; Intelligent Virtual Agents, Zs Ruttkay et al. (Eds,), LNAI 5773, Springer Verlag, 188-200, 2009.

[14] S. Mitra and T. Acharya. "Gesture recognition: A survey"; IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, Vol. No. 37, 311-324, 2007.

[15] G. Müller and M. Möser. "Handbook of Engineering Acoustic". Springer Science & Business Media, 2012.

[16] E. Murphy-Chutorian and M. M. Trivedi. "Head pose estimation in computer vision: A survey"; IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. No. 31, 607-626, 2009.

[17] A. Paiva, I. Leite and T. Ribeiro. "Emotion Modeling for Social Robots"; The Oxford Handbook of Affective Computing, 296, 2014.

[18] M. A. Thafar, "Abstract Modeling of Gestures and Interactive Emotions Using Fuzzy Set for Social Robotics"; MS Thesis, Department of Computer Science, Kent State University, Kent, OH 44242, December 2015.

[19] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility"; Fuzzy Sets Syst., Vol. No. 1, pp. 3-28, 1978.

REACT-R and Unity Integration

Llewyn Salt[1,3], **Julian Wise**[2,3], **Charlotte Sennersten**[3], **and Craig A. Lindley**[3]
[1]ITEE, University of Queensland, Brisbane, Queensland, Australia
[2]CSIT, RMIT, Melbourne, Victoria, Australia
[3]Data61, CSIRO, Hobart, Tasmania, Australia

Abstract— *This paper presents REACT-R which is a modification of an existing cognitive architecture, ACT-R, to incorporate robot embodiment. Robot embodiment is facilitated by situating the cognitive architecture within the robot and allowing it to interact with the environment through its actuators and sensors. The REACT-R module offers flexibility by using a UDP connection to integrate with the Robot Operating System (ROS) allowing REACT-R to be used in simulations or on physical models. We have successfully integrated REACT-R with an interactive 3D Game Engine to autonomously control a simulated quadrotor flying vehicle. The cognitive model also contains varying levels of autonomy, providing a human pilot with the option of controlling different aspects or levels of robot operation.*

Keywords: UAV, Robotics, ACT-R, Cognitive Architectures, Artificial Intelligence

1. Introduction

Following great advances in robotics technology, there has been increased interest in developing cognitive robotics. The availability of systems like the Robot Operating System (ROS) [1] and perception and action systems like Tekkotsu [2] have made it possible for researchers to develop scalable levels of robotic control, from highly manual to fully automatic. Details of low level tasks can be abstracted away, such as moving each of a set of joints individually or rotating a motor to "pick up a chess piece" [3]. This allows AI researchers to concentrate upon higher level decisions rather than being bogged down in lower level path optimisation or fine motor control.

Robotics research has typically focused on optimising processes to perform specific low level tasks and integrating these lower level actions to create robots that can operate well within small problem spaces. Cognitive science and AI deal with more generalised tasks that are open-ended, knowledge intensive, and span longer time intervals [3].

Robotic optimisation processes rely on computations rather than memory for processing power and were shaped this way by the fast CPUs but slow BUS speeds available to access memory in standard von Neumann computer architectures [3]. Conversely, biological brains are capable of performing robust computations from large amounts of memory and computing elements that are slow, inhomogeneous and faulty, taking advantage from having memory and computation happening in the same location to mitigate the bottleneck found in von Neumann machines [4].

Cognitive architectures attempt to facilitate the creation and understanding of agents that have the same capabilities as humans, which is a central goal of both AI from a cognitive perspective and cognitive science [5]. Cognitive architectures like ACT-R or Soar were initially created to model human cognition but have been expanded to work in AI for simulated environments or on robots [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16].

Cognitive embodiment as a concept focuses on the limitations, affordances and features of a physical embodiment of a cognitive processor[17]. Embodied cognition is a theory that emphasises that conceptual representations that constitute our knowledge are dependent on our sensory and motor experiences.

Within the context of this paper, ACT-R is provided with a physical embodiment in the form of a UAV both within a simulated environment in a commercial 3D visualisation and physics engine, and communicating via software operating as a ROS node on a physical quadrotor. In this case, the simulation mirrors the physical UAV.

1.1 Motivations

There are many challenges faced with directly applying a cognitive model to robot embodiment, especially associated with the relationship between high level symbolic models of cognition and the meaning of computations over these models for a physical robot situated in the physical world. In particular, high level symbolic models of a verbal/textual kind have arbitrary or conventional associations between the tokens in which they are expressed and the references and meanings of those expressions that originate in the human use of tokens as signs. This means that the expressed rules do not have any intrinsic meaning for an artificial cognitive agent There are many challenges faced with directly applying a cognitive model to robot embodiment, especially associated with the relationship between high level symbolic models of cognition and the meaning of computations over these models for a physical robot situated in the physical world. In particular, high level symbolic models of a verbal/textual kind have arbitrary or conventional associations between the tokens in which they are expressed and the references and meanings of those expressions that originate in the human use of tokens as signs. This means that the

expressed rules do not have any intrinsic meaning for an artificial cognitive agent /citeLindley2013; the meaning must be engineered into the system by establishing functional relationships between symbolic structures representing rules and what those structures mean in the embodied context. That is, the relationships between symbolic expressions and sensor inputs and actuator outputs must be explicitly engineered. This results in a three-strand development process: i) crafting rules that make sense in terms of the deductive and procedural functions that they are intended to have for a human author, ii) creating the lower level sensor, behaviour and control functions of the physical platform, and iii) creating the links from high level symbolic expressions through the engineered system of embodiment that can assure correct denotational and functional meanings in the situated and embodied operation of the robot in the physical world. Hence, rule sets need to be planned, programmed, and tested over multiple iterations to ensure successful operation and management of sensory information being input into the cognitive system. Inputting sensory information requires fitting the electronics to the embodied cognitive model and fine-tuning the electronic components, both of which add layers of complexity to the cognitive modelling process in the management of electrical signals, adding more areas of potential failure to the testing and debugging process.

In the context of the project described in this paper, direct development of autonomous rule sets within the physical body confines the potential expression of productions to the behavioural space of the physical body, and the levels of abstraction provided for interfacing with the physical system. For example, the propellers are required to be perfectly tuned with all the sensors working correctly before the cognitive model could perform any missions on the drone. This means that development has an inherent bottleneck for the cognitive architecture where the body needs to operate perfectly up to the level of its control interface before cognitive productions can be tested in the embedded agent.

Using a simulation environment can aid the development process by allowing development and testing of the cognitive level in parallel with developing the platform for its physical embodiment.

Having an emulated virtual embodiment operating within a simulation engine means that complex rule-sets can be programmed and tested without the risk of bugs within the hardware development slowing down the development process.

The ability to test the cognitive architecture operating on a simulated model of physical embodiment prior to deployment into the physical embodiment minimizes bottlenecks between development of the cognitive architecture and its physical platform. The potential for developers to be productive in programming more elaborate rule sets for the cognitive architecture, without the risk of bugs within the electronic circuitry or damaging the physical embodiment with untested rule sets. Such a development process shifts the emphasis to more complex cognitive products and allows for more elaborate rule sets to be tested in a virtual environment before being deployed onto the physical model.

2. Overview

Fig. 1 shows the REACT-R module integrating with ROS which can then connect to the 3D/physics simulation or to a UAV or other robot. The decision to use ROS means that the REACT-R module has high portability and can be used with existing robotics platforms. It also allows for users to have access to the many of open-source ROS modules available.

This paper reports the integration of the REACT-R module with the 3D platform simulation as a proof of concept. Future work will involve porting this model to the UAV for operating in an underground mine environment (see [**?**]).

2.1 Cognitive Architecture

The rise of cognitive architectures emerged from Allen Newell's attempts to develop a Unified Theory of Cognition (UTC)[18], [19], defining the hallmark characteristics which underlie the representation of cognition. The UTC aims to define what a cognitive architecture includes and how it relates to human cognition. Essentially it abstracts human cognition into subsections that allows models to be created and evaluated. Furthermore it seeks to explore what it means to be cognitive and the follow on effects it has on problem solving and decision making [3].

The cognitive architecture provides a top-down structure of cognition, where manual robotic refinements can be abstracted into decisions, which the cognitive architecture can manually select.

3. Cognitive Architecture

3.1 ACT-R

ACT-R was developed by John Anderson at Carnegie Mellon University to model human cognition and the mechanism by which humans recall information and solve problems given the current state of the buffers and its procedural and declarative memory [20]. Declarative memory includes facts like *'Sydney is in Australia'*. Procedural knowledge is the how of things, like how you would grasp a cup or how

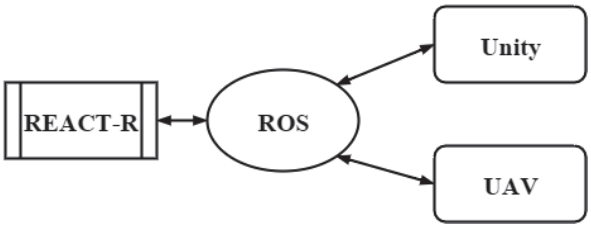


Fig. 1: System Overview of REACT-R with Simulation

you brush your teeth. The buffers are essentially how ACT-R interacts with the world. ACT-R has visual, aural, goal, motor, speech, imaginal and retrieval buffers[21]. As with any agent ACT-R requires a goal it wishes to fulfill. The goal buffer contains the current goal. The data type in ACT-R is chunks, which act like structures containing information required by the buffers. ACT-R takes an appropriate action based on its goal and buffer values.

ACT-R learns on a structural and statistical level. That is, the activation of declarative chunks will increase based on how often they are used by productions. The cost and success of productions are updated based on their observed behavior. Productions can also merge through repeated sequences of productions to create new ones [5].

To this end ACT-R/E was created to place the cognitive architecture into a physical embodiment with the goal of modeling human sensory phenomena. The cognitive architecture can accordingly have sensory phenomena as inputs into the cognitive framework from an external environment which would be taken into account during the decision making process [8].

ACT-R/E is not open source and it still incorporates many buffers that could potentially make customisation difficult.

3.2 REACT-R

The novel REACT-R variant reported in tis paper adds a module to the ACT-R 7.0 core software package to allow for a robot embodiment of ACT-R, Fig. 2 shows the REACT-R software architecture. ACT-R has been expanded upon to extend it from an abstraction of a human mind interacting with a screen using a keyboard and mouse [22] to a model that can function as a robot embodiment of cognition with any number of sensors. In this instance the embodiment was a virtual UAV sitting in a rendered model of a real mine. The mine map was formatted as a .dxf file and then voxelised, i.e. turned into multiple atomic cubes.

The UAV sensors consisted of:

- Four Sonar Sensors
- An Optical Flow Sensor
- An 6 DOF Inertial Motion Sensor

The REACT-R module takes in the sonar and optical flow sensory information and leaves the lower level control of attitude and position to the UAV control system.

REACT-R was designed to be intuitive and modular. It is easy to integrate with robotics projects that are currently using ROS. Fig. 1 shows how REACT-R can be interchangeable between a simulation and a robot.

REACT-R uses only one additional buffer that communicates using a UDP connection via ROS nodes. The buffer can be easily modified to incorporate any sensor configuration and send back any commands. ACT-R/E has visual, aural, vocal, motor, configural, and manipulative modules to emulate human cognition, as the quadrotor model requires sensory inputs which are not human-centric (sonar, lidar and optical flow). Communication through the REACT-R module was seen as preferable providing a centralised and modifiable implementation. This module consists of two buffers: Sensors and Actuators. The inputs can be any number of sensors and the outputs can be whatever is required: In our case it outputs the direction the vehicle should travel in.

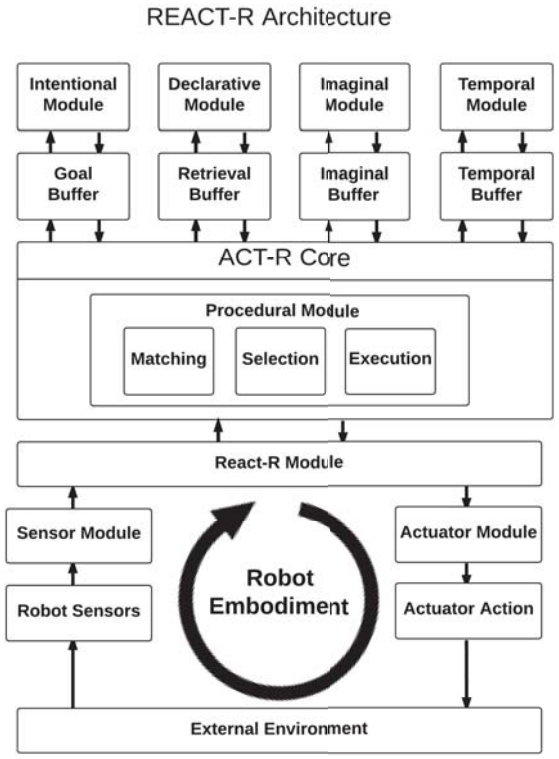


Fig. 2: Software Architecture Diagram for REACT-R Integration

The REACT-R module is designed to work as a co-pilot with a human and augment the user experience. To this end we have created the REACT-R module with varying levels of autonomy:

- None - REACT-R does not interfere at all with the pilot's flight.
- Semi - REACT-R provides hover control for the UAV and will override user control if sonar sensors reading less than a nominated distance from a surface.
- Functional - REACT-R will move autonomously to a goal location avoiding obstacles as it goes. The operator is still able to take control if they believe REACT-R is not operating as desired.

REACT-R operates as a high level decision maker, being capable of perceiving phenomena and performing a decision making process. Given the architectural topology of REACT-R within its environmental context, commands can be output through the module to a lower level language operating on the robot embodiment platform.

The newly created REACT-R module does not discriminate between whether the Cognitive Architecture inhabits a simulated environment, such as our 3D simulator, or whether the cognitive processor exists in the physical world relating to its surrounding environment over the module connection to ROS.

4. Simulation

We created a simulation of a UAV to test the REACT-R module. This was done using Unity, a 3D game engine that uses NVidia's PhysX physics calculations.

The UAV was modelled in BlendR and then put in the Unity simulation. Forces are added depending on inputs to influence UAV motion. The sensors are also simulated by taking measurements that would be provided by these sensors using either ray casting or rigid body mechanics.

A Computer-aided design (CAD) mesh rendering of an underground mine was imported into Unity as a simulated environment. A UAV was modelled with kinematics as an object within the simulated environment for ACT-R to embody.

The simulated embodiment of ACT-R is placed within a surrounding environment with virtual sensors programmed on the UAV within Unity to provide an input to ACT-R of spatial sensory recognition, passed through the REACT-R module.

For example, if the operator is operating in semi or functional autonomy, sensors on the UAV will detect the distance that it is from other objects and this data will be made available to the REACT-R module and subsequently the ACT-R productions.

Within the context of a Unity environment, the kinematics and sensory inputs of the UAV were programmed C#, using a UDP connection to REACT-R for the top level decision making process. REACT-R receives relational positional information and tells a positional control system how to move in the virtual space.

4.1 Kinematics

It was desired that the UAV be as realistic as possible, therefore a kinematics model was created. This was done by relating the roll, ϕ, pitch, θ, yaw, ψ, and desire thrust to motor speeds. To be at equilibrium, stable hovering, the kinematics must satisfy these equations:

$$\sum_{i=0}^{4} T_i = -mg \tag{1}$$

$$T_{1-4} || g \tag{2}$$

$$\sum_{i=0}^{4} M_i = 0 \tag{3}$$

$$(\omega_1 + \omega_3) - (\omega_2 + \omega_4) = 0 \tag{4}$$

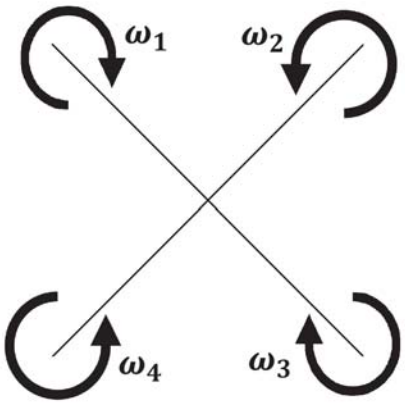


Fig. 3: X Configured Quadrotor with Propeller Labeling

Where m is the mass of the UAV, g is gravity, and T_i represents the thrust generated, M_i is the moment generated, and ω is the rotational velocity of each propeller, Fig. 3 shows which the X configuration and the labeling convention of each propeller.

To move the UAV these equilibrium values must be disturbed. That means that the sum of the thrust is greater than $-mg$ to go up and less than $-mg$ to go down.

To obtain the rotational rates we calculate:

$$\dot{\psi} = k_Y((\omega_1 + \omega_3) - (\omega_2 + \omega_4)) \tag{5}$$

$$\dot{\phi} = k_R((\omega_1 + \omega_4) - (\omega_2 + \omega_3)) \tag{6}$$

$$\dot{\theta} = k_P((\omega_1 + \omega_2) - (\omega_3 + \omega_4)) \tag{7}$$

$$F = k_F(\omega_1 + \omega_2 + \omega_3 + \omega_4) \tag{8}$$

Where k_Y, k_R, k_P, and k_F are proportional constants and F is the lift force generated. If we assume common proportionality, that is $k = k_Y = k_R = k_P = k_F$, and that $F = \sqrt{\sum_{i=0}^{4} T_i}$ then we obtain:

$$\begin{bmatrix} \dot{\psi} \\ \dot{\phi} \\ \dot{\theta} \\ F \end{bmatrix} = \begin{bmatrix} k & -k & -k & k \\ k & k & -k & -k \\ k & -k & k & -k \\ k & k & k & k \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} \tag{9}$$

The inputs from the controller of the operator give the desired T, $\dot{\psi}$, $\dot{\phi}$ and $\dot{\theta}$. Therefore to get the motor speeds, we calculate:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = K^{-1} \begin{bmatrix} \dot{\psi} \\ \dot{\phi} \\ \dot{\theta} \\ F \end{bmatrix} \tag{10}$$

Where K^-1 is the inverse of the proportional gain matrix in equation 9 [23].

To obtain the desired thrust, we obtained available data from Cobra CM-2217-20 $Kv = 950$ from [24]. This data allowed us to relate throttle to thrust and then propeller

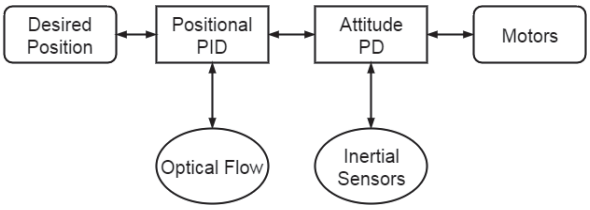


Fig. 4: Control Flow

speeds to thrust. The relationships were formalised through polynomial regression.

4.2 Control

To fully simulate how a drone would perform in a real environment it is not suitable to adjust anything other than the torque and moments created by the motors. This means that a control system has to be used rather than directly specifying the UAV's position and attitude in the environment. Fig. 4 shows the control system as it stands; positional and attitude control will be explained in the following subsections.

4.2.1 Attitude

The attitude of a UAV is the difference in the rotation of the UAV's reference frame relative to the world reference frame; Fig. 5 shows both of these reference frames. To control the attitude of the UAV a proportional and derivative (PD) control system was implemented. A PD control was chosen because it is guaranteed to be exponentially stable for most rotations [25]. The PD control system is given by:

$$R = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{11}$$

$$E = R_d - R_a \tag{12}$$

$$\dot{R}_c = -(K_P E + K_D \frac{\delta E}{\delta t}) \tag{13}$$

Where R_d is the desired attitude given by the user, R_a is the actual UAV attitude, $\dot{R}_c$ is the required rotational velocities to make the actual and desired attitudes the same. K_P and K_D are the proportional and derivative constants, and t is time.

To simulate the UAV that is anticipated to go into the mine, the rotation for roll and pitch was limited to $30°$, which is equivalent to stabilised mode, with the option to go into acrobatic mode to perform advanced aerial acrobatics such as flips and barrel rolls, if necessary.

4.2.2 Position

An external control loop was added to create the ability for the quadrotor to move to a desired position. The positional controller acts as an external loop to the attitude control. The user or cognitive model feeds the desired position to the positional controller which outputs the desired attitude that is required to achieve this position. This controller is a proportional, integral, and derivative (PID) controller and this was given by:

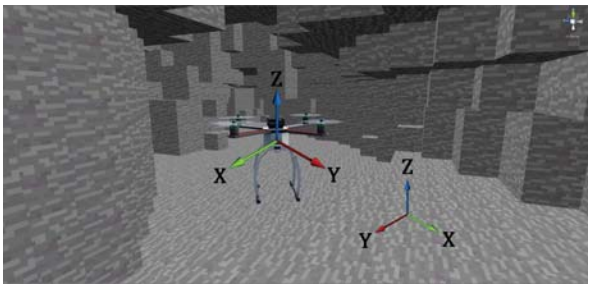


Fig. 5: UAV and World Reference

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{14}$$

$$\epsilon = P_d - P_a \tag{15}$$

$$P_c = -(K_P P + K_D \frac{\delta P}{\delta t} + K_I \int_0^t P dt) \tag{16}$$

$$\dot{R}_d = \begin{bmatrix} -P_c(z)\sin(\psi) + P_c(x)\cos(\psi) \\ P_c(x)\sin(\psi) + P_c(z)\cos(\psi) \\ R_d(y) \end{bmatrix} \tag{17}$$

Where P_a is the actual position of the UAV, P_d is the desired position of the UAV, and Pc is the initial controller output. Pc gives the desired roll and pitch rotations and throttle required to correct the x, y, and z offset. $\dot{R}_d$ is the rotations desired to be put into the attitude controller. The positional controller does not affect the yaw. The yaw is used to rotate the required roll and pitch in the UAV's reference frame relative to the world reference frame. Fig. 5 shows how the UAV's and world reference frame can differ within the voxelised mine model.

4.3 Communication

Prior research into interfacing ACT-R with Unity saw messaging passing over TCP through JSON associative arrays [13], [6].

REACT-R communicates to the robot embodiment through a UDP connection. The reasoning behind using the UDP protocol is that it is faster than TCP, and error checking can be a hindrance to the real time flow of information. There is a constant stream of packets being sent to the cognitive model and for it to react appropriately: the most recent piece of information is crucial. TCP will repeat packet requests when transmission is lost, which can create a backlog. This effectively creates a time lag between what is going on in the real world and what the cognitive model is receiving.

For example, in relation to visual perception, sometimes we fail to see or become aware of an object when it first comes into our field of view. This could be seen as

some failure of our visual processor to identify the object. However, we are not concerned with where the object was when it first came into the field of view, but where it is now that we are aware of it and can make a decision about how to react to it. Similarly, current information should take precedence within the decision making process, and as a constant feed of packets is being sent from all sensors the most recent packet is the most relevant.

The cognitive model's REACT module has two buffers, a sensor and an actuator buffer, the sensor buffer receives information from the robot embodiment and the actuator buffer relays information to the robot embodiment.

The UAV simulator uses sonar sensors to understand relative distances from it to collidable objects. These distances are sent to the cognitive model for evaluation along with a user specified status packet that determines the level of autonomy. The packet that is given to the model in this specific scenario is six bytes: one for status and five for sonar sensors. The model will then make a decision about what direction or combination of directions it should move in. The directions are up, down, left, right, forwards, and backwards. The packet that it returns to the simulation is: status and which directions it wants to move in. This also takes into account the goal destination if it is in full autonomy mode.

For a physical embodiment a ROS node is used to facilitate this communication so that it may integrate easily with other robotics platforms already using ROS. The ROS node can also be bypassed and directed by UDP connection between REACT-R and any platform to be used.

5. Discussion

5.1 Reusable Module

One of the overarching objectives within the scope of the research was to create an interchangeable ACT-R module capable of interfacing with multiple external applications. The REACT-R module programmed for this purpose requires an IP address and port over which information can be sent to any external interface receiver. When an IP address and computer port are specified, ACT-R will send and receive packets over the stream regardless of whether the embodiment is virtual or physical thus allowing for modeling scenarios, flight tests and simulated missions. The ability to first simulate before testing on a physical embodiment ensures the productions are accurate and easily testable prior to testing REACT-R productions on a physical UAV. The same rule sets of productions used by REACT-R in one scenario of embodiment can be interfaced with a simulated embodiment, or extended to physical embodiment, given the sensory input is modelled in all embodiments to reflect the same sensory phenomena. For example, the same REACT-R production set can be tested and fine-tuned on a simulated drone flight before being implemented on a physical drone. Through the use of the REACT-R module, sensory inputs and reactive outputs from ACT-R are easily modifiable, with the requirement that the external interfacing software can communicate over a UDP connection with a known port and IP address to which packets are sent and received.

5.2 Future Development

Future scope within the development of the REACT-R module will focus on the elaboration of productions within the UAV model and furthering the complexity of interfacing with the physical world through ROS.

5.2.1 Production Elaboration

Within the context of autonomous UAV surveying, productions and rule sets will be further established to provided the cognitive architecture with a greater sense of autonomy. Currently the cognitive A.I has programmed responses to allow for positional control, obstacle detection and obstacle avoidance. As time and research progresses into interfacing the physical UAV with Unity, the complexity of productions given the sensory inputs into the cognitive architecture can expand. These would include productions to allow for optimal path recognition, flight surveying with additional sensory inputs, mapping of terrain and a further refinement of responsive rule sets to sensory inputs.

5.2.2 Long Term Memory

As ACT-R's declarative memory chunks are fired in productions to complete tasks, the success rate of productions is updated in such a way that frequently coupled productions are automatically merged. The automatic merging of processes allows ACT-R to optimise the decision making process by minimising cost of time for decision calculation. While production merging can aid ACT-R in the process of self optimisation, the drawback of its current implementation is that when the program is terminated all self-created productions and optimisations are lost. Every time ACT-R is restarted, the declaration merging for optimisation needs to rediscovered within the uninterrupted software run time. Providing ACT-R with a long term memory module which saves self-established productions alongside human-programmed productions would ensure that knowledge attained from past experience within an embodiment could aid in the current and future decision making of the cognitive architecture. Future research will see ACT-R retaining self created productions through a long term memory buffer to utilise previously stored experience which would be added to the production rule set on start up and saved when ACT-R is shut down. From a research perspective it will be fascinating to see how ACT-R's self-created rule sets vary from those programmed by humans over an extended period of time.

5.2.3 ROS Optimsation and Elaboration

Within the current state of implementation, ACT-R successfully operated as a node within the Robot Operating

System suite, both sending and receiving packets to a physical UAV operating on a ROS system. Future research will include ACT-R performing the decision making process as an autonomous UAV within the physical world. As a cognitive architecture, ACT-R could perform the high level decision making for ROS, selecting which algorithms would be best to apply depending on context-based scenarios. Flight productions and rule sets would first be tested and refined within the virtual environment on the virtual UAV embodiment before being applied to the physical drone embodiment. The REACT-R module provides a streamlined method of passing messages between any client-server based architecture with the hope of modeling behaviour in a simulation before applying the cognitive architecture to a physical embodiment that can meet real world challenges.

6. Conclusion

We have developed a physics simulation of a UAV that validates the use of REACT-R as a simple and easily customisable platform for use on a robotic platform. Using ROS provides ease of integration with currently standing projects.

We have also demonstrated a simulated UAV embodiment of REACT-R that is capable of navigating to a position and avoiding obstacles, as well as assisting pilots in flight by providing hover control and obstacle avoidance. It is anticipated that AI will not replace human operators but rather augment their experience, and we have demonstrated the facilitation of this.

Since REACT-R is anticipated to be used in conjunction with other robots in a multi-agent system, as well as humans, using the core ACT-R cognitive architecture is justified because it is designed with the intent of modeling human behaviour to give robot embodiments better understanding of their human counterparts or team mates.

The elaboration of future work was done in the hope that the logical expansion of the model outlined in this paper can be realised so that we may have harmonious co-operation between human and robot workers in the future.

References

[1] M. Quigley, K. Conley, B. Gerkey, J. FAust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Mg, "ROS: an open-source Robot Operating System," *Icra*, vol. 3, no. Figure 1, p. 5, 2009.

[2] E. Tira-thompson and A. D. S. Touretzky, "Tekkotsu : A Rapid Development Framework for Robotics," *Robotics*, 2004.

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

[4] G. Indiveri and S.-c. Liu, "Memory and Information Processing in Neuromorphic Systems," *Proceedings of the IEEE*, no. 612058, 2015.

[5] P. Langley, J. E. Laird, and S. Rogers, "Cognitive architectures : Research issues and challenges," *Cognitive Systems Research*, vol. 10, no. 2, pp. 141–160, 2009.

[6] R. M. Hope, M. J. Schoelles, and W. D. Gray, "Connecting ACT-R to the World with JSON over TCP," *Proceedings of the 12th International Conference on Cognitive Modeling*, no. rm 108, pp. 354–355, 2013.

[7] W. G. Kennedy, M. D. Bugajska, M. Marge, W. Adams, B. R. Fransen, D. Perzanowski, A. C. Schultz, and J. G. Trafton, "Spatial Representation and Reasoning for Human-Robot Collaboration," *Architecture*, vol. 22, pp. 1554–1559, 2007.

[8] J. G. Trafton, A. M. Harrison, B. R. Fransen, and M. D. Bugajska, "An embodied model of infant gaze-following," *International Conference of Cognitive Modeling*, no. 2003, 2009.

[9] T. Deutsch, C. Muchitsch, H. Zeilinger, M. Bader, M. Vincze, and R. Lang, "Cognitive decision unit applied to autonomous biped robot NAO," *IEEE International Conference on Industrial Informatics (INDIN)*, pp. 75–80, 2011.

[10] P. Langley, "Intelligent Behavior in Humans and Machines," *Aaai*, 2011.

[11] J.-Y. Puigbo, A. Pumarola, C. Angulo, and R. Tellez, "Using a cognitive architecture for general purpose service robot control," *Connection Science*, vol. 00, no. 00, pp. 1–14, 2008.

[12] F. E. Ritter, D. Van Rooy, R. St. Amant, and K. Simpson, "Providing user models direct access to interfaces: An exploratory study of a simple interface with implications for HRI and HCI," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, vol. 36, no. 3, pp. 592–601, 2006.

[13] P. R. Smart, T. Scutt, K. Sycara, and N. R. Shadbolt, "Integrating ACT-R Cognitive Models with the Unity Game Engine," no. 2000, 2012.

[14] P. Smart, K. Sycara, and C. Lebiere, "Cognitive Architectures and Virtual Worlds : Integrating ACT-R with the XNA Framework."

[15] F. Tanaka and S. Matsuzoe, "Children Teach a Care-Receiving Robot to Promote Their Learning: Field Experiments in a Classroom for Vocabulary Learning," *Journal of Human-Robot Interaction*, vol. 1, no. 1, pp. 78–95, 2012.

[16] C. Wei and K. V. Hindriks, "An agent-based cognitive robot architecture," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7837 LNAI, pp. 54–71, 2013.

[17] R. A. Wilson and L. Foglia, "Embodied Cognition," 2016.

[18] J. R. Anderson and C. Lebiere, "The Newell Test for a theory of cognition." *The Behavioral and brain sciences*, vol. 26, no. 5, pp. 587–601; discussion 601–648, 2003.

[19] A. Newell, *Unified Theories Of Cognition*. Harvard University Press, 1990.

[20] J. R. Anderson, "A Simple Theory of Complex Cognition," pp. 355–365, 1996.

[21] J. Whitehill, "Understanding ACT-R âĂŞ an Outsider ' s Perspective," *Mplab.Ucsd.Edu*, pp. 1–12, 1993.

[22] J. R. Anderson, M. Matessa, and C. Lebiere, "ACT-R: A Theory of Higher Level Cognition and its Relation to Visual Attention," pp. 439–462, 1997.

[23] C. Santoro, "How does a Quadrotor fly? A journey from physics, mathematics, control system and towards a "Controllable Flying Object"," Catania, pp. 1–64, 2014.

[24] L. Miller, "How to use the Multirotor Motor Performance Data Charts," Innov8tive Designs, Inc., Vista, Tech. Rep.

[25] R. Mahony, V. Kumar, and P. Corke, "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

Experiments in Fuzzy Multi-Robot Security System

Mahmoud Tarokh [1], Malrey Lee [2] , Hongseok Chae[2], Jeongran An[2],
Gisung Jeong[3] and Dukmo Yun[3]

[1] Department of Computer Science, San Diego State University, San Diego, CA 92128-7720
mtarokh@mail.sdsu.edu

[2] 561-756, Center for Advanced Image and Information Technology, School of Electronics & Information Engineering, Chon Buk National University, 664-14, 1Ga, Deokjin-Dong, Jeonju, Chon Buk, Korea. mrlee@chonbuk.ac.kr

[3] Department of Fire Service Administration, WonKwang University, Republic of Korea

Abstract

This paper discusses a distributed fuzzy system for a set of robots engaged in securing inside or outside of a building and detecting any intruder. Each robot receives information from other robots but makes its own decision to engage in a particular activity such as patrol, inspection, intruder chasing. This decision is to best for overall objectives of the team and also to satisfy its own constraints such as its existing battery power. The performance of the system is investigated under various scenarios.

Keywords: Robotic security system; distributed systems; fuzzy logic.

1. Introduction

This paper deals with securing an area such as a warehouse during off work hours. A brief history of robotic security systems is provided in [1]. Much of the work on robotic security is based on behavioral robotics [2] and artificial intelligence robotics [3]. The work reported in [4] proposes a surveillance system intended for mobile security robots with the specific purpose of detecting changes to the environment without concern to the nature of changes. A related work uses a robotic system to detect abnormal and dangerous situations and their notification possibly through internet [5]

The work in [6] develops a team of cooperative security robots intended to work along the international border to aid the effort to prevent illegal immigration. The goal is to have the robots function essentially as a flexible sensor array having each robot configured with a suite of sensors, such as thermal imaging, to detect human presence. Functioning as a flexible sensor array, the robots uniformly space themselves in a line along the border yet are constrained within a specific domain, or region, within which the robots work. A

multi-robot system with Java software is proposed in [7] to obtain a distributed security system.

In a previous paper [8] we proposed a multi-robot security system. In this paper we focus on the performance on the system under various security events. In particular we show that the system behaves close to what a security systems consisting of human security personnel would do in similar situations.

2. Background

The goal of the robot team for this project is to cooperatively maintain security in a moderately complex environment such as that found in warehouses and laboratories after hours when no human activity is expected. The environment is moderately complex in the sense that the floor plans are known in advanced while other features of the environment, such as locations of objects, robots, and intruders are dynamic. Being physical entities, the robots move around acquiring new information and updating their knowledge which is then used as the basis for their responses.

Decision-making is the core focus of this paper and through it the cooperative security maintenance effort emerges and is seen in a robot selection of a particular activity over time as the mission progresses. A robot's decision about which security-based activity to execute considers not only what the robot agent believes it *knows* about the world but also what is *best for the team* overall, rather than desires of the individual robot, thereby yielding the *cooperative* aspect sought among the robots while permitting each to *autonomously* make its own decisions. guard, inspect, chase intruder, etc. during their mission. Deciding which activity to engage in over time in support of security depends on the robot agent decision-making process and involves several factors. These factors are remaining battery charge compared to the power need by an activity, the number of robots already engaged in an activity, the effective area coverable by the robot team, security related events, and duration a robot has been engaged in an activity. A fuzzy logic based method is used here for decision making due to its ability to establish qualitative relationships between different possible input types in terms meaningful to human reasoning. In addition, fuzzy logic methods have the ability to easily integrate and handle

information originating from multiple sources even when the information is incomplete or unreliable as it usually is in realistic situations.

2.1 Definitions of Quantities

In this paper, it is assumed that the robots are to engage in the limited set of security activities that are *patrol*, *inspect*, and *chase* the intruder with the idea of disabling it, e.g. using a TASER. A fourth activity called *Guard* is also engaged by the robots to maximize the mission duration by putting the robot in a low power usage mode while having them remain actively observant when conditions deem appropriate. For each activity the robot must determine the appropriateness of a particular activity at a particular moment such that the effort to maintain security in a cooperative manner is achieved while the needs of an individual robot are met.

Patrol and *inspect* activities for example, both of which use an identical fuzzy logic system with identical input types (but different actual inputs) use the following:

Activity power draw P_{draw}: This is the energy required for patrol or inspect activity. The higher the energy drawn by the activity, the less likely it is to be selected when energy reserve of the robot battery is low.

Battery energy E_{bat}: Battery energy is compared against estimated energy requirement by an activity.

Fraction of robots in activity F_{rob}: This is the fraction of other robots currently engaged in the patrol or inspects activity. The more robots already engaged in the activity, the less likely the current robot engages in it.

Floor Coverage: F_{cov} This is the percentage of the environment covered by the robots. The greater the floor coverage, either due to higher numbers of robots or greater coverage offered by each individual robot, the less need there is for the robot to actually move about and therefore the less likely the robot is to choose to engage in patrol or inspect.

Time $T_{in/out}$: This is the time either away from or engaged in the patrol or inspects activity. Time away, treated as a negative number, from patrol or inspect increases the likelihood that the robot will engage in these activities. Time spent in either of these activities decreases the likelihood it will continue at them.

2.2 Fuzzy Behaviors

It would be possible to create individual fuzzy logic systems that take all the above inputs simultaneously and produce the required action. However, this would lead to a high dimensional rule matrix which would not only complicate the devising of the rules but also require intensive computation. Instead, we decompose the overall fuzzy system into a number of subsystems and use only two inputs for each subsystem. The output of each subsystem is normalized and is then fed sequentially into the next fuzzy subsystem. The normalization value for each subsystem's output is set at startup by determining the range of output values that a subsystem generates and scaling that output to a minimum of zero and a maximum of 1. Fig. 1 shows the four-subsystems used by both *patrol* and *inspect* activities.

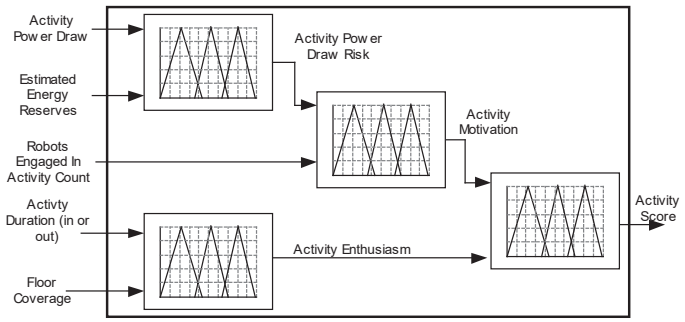


Fig. 1. Fuzzy logic system for determining *patrol* and *inspect* activity scores.

A subsystem common to all activity fuzzy systems determines *power Draw Risk* (P_{risk}) of its associated activity, as shown in Fig. 1. The output of this subsystem is based on the estimated energy remaining in the battery E_{bat} and the estimated energy requirement P_{draw} for the activity. The concept behind P_{risk} is that when faced with low remaining battery energy, a robot agent is unlikely to engage in activities that require large amounts of energy so that its duration of service for a mission is prolonged as far as possible. This is an especially useful concept when other robots and cooperation among them are considered, i.e. if other robots are present and an individual robot or robots have low energy reserves, it lets the ones with higher energy reserves engage in the more energy demanding activities. Weighing a robot's own P_{risk} against other robots present in the system is handled in the fuzzy subsystem that examines the robot's motivation to engage in an activity.

To account for imprecision on estimates of energy requirement and available battery energy, these variables are treated as fuzzy. The membership functions for P_{draw} and E_{bat} are shown in Fig 2. The associated fuzzy rule matrix that maps fuzzy inputs P_{draw} and E_{bat} into P_{risk} is shown in Table 1. The P_{risk} subsystem, as with all fuzzy logic subsystems in this project, uses the intersection operator to combine the fuzzified input into a fuzzy output.

Outputs from certain fuzzy subsystems serve as input to others. Mentioned earlier was that the P_{risk} weighed against the number of other robots N_{rob} engaged in the activity is considered when a robot decides a new activity. If a particular activity poses a large energy requirement P_{risk} and a sufficient number of them are already engaged in it, then it makes sense to choose another activity that drains less energy and permits a longer duration of the robot's mission.

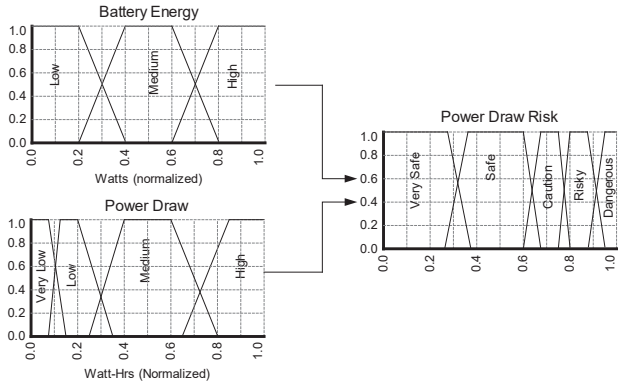


Fig. 2 Energy draw Subsystem

On the other hand, if no or few other robots are engaged or if energy risk is low then certainly the activity must be given higher consideration. Common to the fuzzy systems associated with activities *patrol*, and inspect (as well as chase to be discussed later) is the subsystem called *Motivation*. This subsystem takes as input the energy risk of an activity and the number of robots already engaged in that activity. The human reasoning behind this is that the higher activity's energy risk and the more robots already engaged in it, the less motivated a robot is to participate in that activity. On the other hand, low energy risk or low number of other robots engaged in the activity makes the robot more motivated to engage in it. The subsystem for determining an activity motivation and its associated fuzzy rule matrix are defined similaraly.

Enthusiasm to engage in an activity is a quality humans can relate to and may be affected by various issues such as mood, tiredness, and repetition. The fuzzy subsystem *Enthusiasm* is a measure of the robot agent's enthusiasm to engage in activities *patrol* or *inspect* (as well as chase to be discussed later). Enthusiasm to patrol or inspect takes as input the time in or out of the activity $T_{in/out}$ and the floor coverage F_{cov} of the environment by the robots present in the system. From the perspective of human reasoning, the robot agents are more enthusiastic to engage in an activity the longer they have been away from it. On the other hand, they are less enthusiastic if the environment is well covered by the robots and the longer they have been engaged in it. The subsystem used to determining enthusiasm to engage in an either patrol or inspect can be defined appropriately.

The fuzzy logic system for determining activity *Chase*'s score is shown in Fig. 3 and consists of four subsystems. In addition to the inputs power draw for chase P_{draw}, and remaining battery energy *Ebat* and fraction of the robots in the chase activity F_{rob} that were already defined, chase activity uses the following inputs:

Elapsed time F_{elps}: This is the time since the intruder was sighted. As time increases since a last sighting of an intruder, the more likely it is the intruder got away, making the robot less likely to chase.

		Estimated Remaining Battery Energy		
		Low	**Medium**	**High**
Activity Energy Draw	**V. Low**	Safe	V. Safe	V. Safe
	Low	Caution	V. Safe	V. Safe
	Medium	Risky	Safe	V. Safe
	High	Dangerous	Safe	V. Safe

Table 1. Fuzzy Rule Matrix.

Distance from intruder $D_{\text{int}\,rd}$: This is the distance of the robot from the location where the intruder was sighted by this or another robot. The further the intruder is from the robot the less likely the robot agent is to choose to chase it.

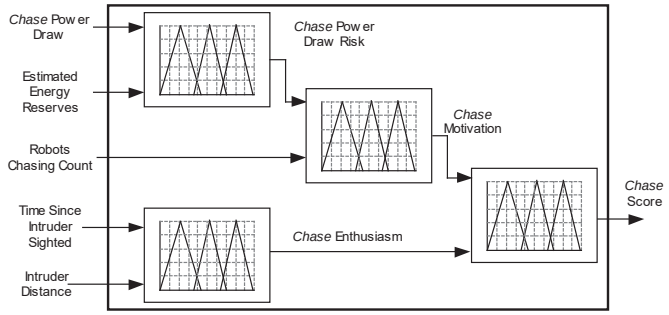


Fig. 3. Fuzzy system for determining chase activity score.

Different from inspecting and patrolling, a robot agent's enthusiasm to chase an intruder is a measure of the elapsed time since an intruder was sighted against how far away the sighting was. As an intruder roams throughout the environment sighting robot agents periodically broadcast intruder sighted messages composed of, among other bits of information, the intruder's position. These messages are repeated as long as the intruder can be "seen" by the robot agents. Robots on the team mark the time the message is received and calculate the distance to the intruder's reported position. At larger distances, robot agents are less enthusiastic to engage in *Chase* since it is less likely they can catch up to it and disable it; at closer distance the opposite is true. The time since the intruder's sighting is considered because a pursued intruder may successfully evade the robot(s) resulting in the intruder sighted broadcasts to cease. Therefore, as the time since the sighting increases, a robot agent's enthusiasm to chase decreases. This permits the robot agents to still move toward the last known location of the intruder even when it is no longer in sight yet. Enthusiasm to chase subsystem and its

corresponding fuzzy matrix can be defined using standard technique. Fuzzy subsystem for enthusiasm to chase score

The fuzzy subsystems that determine final scores for activities *Chase*, *Patrol*, and *Inspect* combine each of these activities' motivation and enthusiasm to determine those activities' final scores and is easily related to with the thinking that the more enthusiastic and motivated a robot agent is to perform a specific activity, the more likely it is to decide to do so.

Finally, the *Guard* activity is a low-power activity in which the robot remains still, only using sensors to watch for intruders. Its purpose is to prolong the duration of the robot's mission by having it conserve energy when its energy reserves is low. Unlike activities *patrol*, *inspect*, and *chase*, the robot agent is more likely to engage in this guard activity as its power draw increases. The final score for activity *Guard* considers both the power draw and the previously mentioned floor coverage. The human-based thinking is that the higher the power draw risk, the more likely the robot must conserve its reserves; yet the fewer robots available to cover the environment the less likely it is to do so. The fuzzy subsystem that determines a final score for activity *Guard* is shown in Fig. 4.

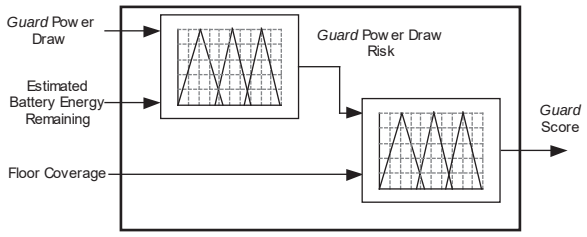


Fig. 4 Fuzzy system for determining guard activity score

3. Simulations, Experimentations and Results

Tests were carried out that examined the intelligent robot security system as a whole in both a purely simulated environment and in a real-world environment. Testing of the system as a whole in a purely simulated environment permits development and verification that the system-wide code. The code is composed not only of the decision-making process but also such other areas as networking, communications, sensory interpretation, and navigation. Testing of the system in the real world verifies that the system actually functions as expected under conditions it was meant for. Fig. 5 shows the environment for the experimental work.

Robot and intruder intelligent systems are tested from primitive to sophisticated behaviours. The trials were executed multiple times for various situations and conditions. The tests were mainly focused on the decision-making over the robot's mission that maintains security. The best way the system can achieve its goal is by performing the following tasks:

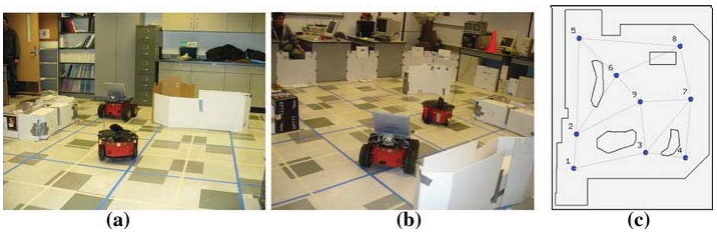


Fig. 5 Views of the environment and the waypoint graph overlaid onto the environment

- Periodically alternating behaviors like patrol and inspect of robots under normal conditions.
- Increasing patrolling duration during heightened security conditions.
- Pursuing and disabling sighted intruders.
- Maintaining diversified behaviors across the team and the environment so that too many robots are not engaged in the same behavior.
- Maximizing the duration of their mission with respect to the limited energy reserves.

Each test demonstrates key aspects of the fuzzy logic decision-making leading to the desired goal (mission) achievement. Following aspects are expected in normal conditions:

- Robots properly alternate between patrol and inspect behaviors.
- Robot appropriately responding to security-based events.
- Greater floor coverage offered by multiple robots to share work load and to extend their duration of the mission.
- Robot's respond properly to low battery conditions.
- Intruders entering and making an exit appropriately,
- Intruder's response to the robot chasing it.

3.1 Demonstrative Tests

Five tests are presented that demonstrate the robot and intruder fuzzy decision-making. Test 1 demonstrates the decision-making of a single robot that is fully charged and under normal conditions when no security events occur (no intruder in the environment). Test 2 adds security events to demonstrate their impact on decision-making of the robot and the intruder. Test 3 demonstrates how increased floor coverage offered by multiple robots translates into cooperative aspects of decision-making by the sharing the workload. Test 4 demonstrates how remaining battery energy and power required by a behaviour influence decision-making towards energy conservation to extend robot mission. Test 5 demonstrates the diversity of robot's behaviour in a multi-robot team.

Test 1: In Normal Condition with no Intruder in the Environment

This test demonstrates decision-making of a single robot that is fully charged and under normal conditions when no security events occur. When there are no security threats, the decision-making system of the robot should periodically alter between patrolling and inspecting behaviours unless and until a sufficient number of robots are present to guarantee having some robots conserve energy by guarding. Test 1 demonstrates this periodic alternation and time spent in each is consistent with the decision-making configuration. Fig.6. shows the normalized duration of the patrol and inspect behaviors in which 50 seconds of patrolling and 30 seconds of inspection.

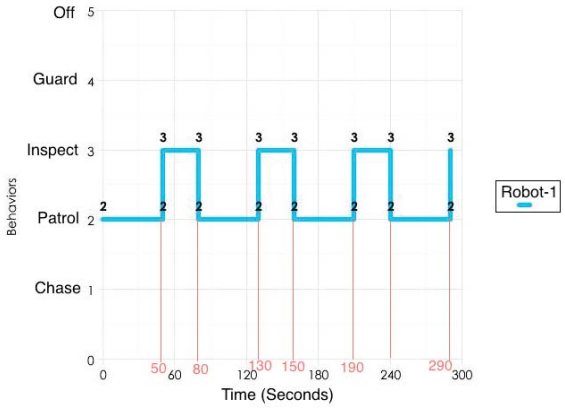


Fig. 6 Patrol and Inspect Behaviors Under Normal Conditions

Normal durations for each behaviour could be based upon various considerations, such as the amount of floor space to cover or need of the environment. For example if the robot is more prone to security concerns then it is in need of longer patrol durations or if more prone to safety concerns then it needs longer inspection durations. Two key parameters are the Patrol Duration normalizer and Inspect Duration normalizer. In security related events these normalizers reduce the duration in inspection and increase the duration of patrolling.

Test 2: One Security Event and One Intruder in the Environment

This test demonstrates the impact of security events on the robot and intruder decision-making. An intruder causes all security events such as intruder sighting, intruder disabling, intruder escaping, intruder escaping with object, intruder made an exit and object missing. Events such as intruder sighting, intruder escaping with or without object provokes the robot to chase instead of patrolling and inspecting. This is due to the fact that the robot has the intruder in its current vision and chase is the fastest behaviour to catch the intruder and to disable it. So toprevent an intruder to escape, chase is given higher priority (behaviour score) over patrol and inspection. Intruder disabling, intruder made an exit and object missing provokes the robot to patrol instead of inspection because these events indicates a possible presence of intruders in surrounding area and patrol is much faster than inspection. So to avoid giving time to intruder for hiding, patrol is given higher priority (behaviour score) over inspection.

The robot's enthusiasm is increased or decreased in order to provoke or avoid the robot to engage in a behaviour. In response to a security event, enthusiasm can be increased by decreasing the current value of behaviour's duration in or out of that behaviour. Thus decreasing the positive duration of a behaviour, the robot is engaged in brings that value closer to zero with a minimum of zero. Ultimately the behaviour duration is affected in a way such that the robot will remain in the more desired behavior longer if it is already engaged in it or switches over to it sooner if not.

The value to increase or decrease the behaviour duration of patrolling and inspecting are set by the user. Chase's enthusiasm is different from enthusiasm of patrol and inspect. Chase's enthusiasm depends on time elapsed since the last intruder sighted and the distance to the intruder at the time of that sighting Figure 7 shows the robot's and intruder's behaviours for the duration of the test. In this test two events, object missing and intruder sighting are shown. A circled "A" in Figure 7 is the object missing event and "B" is the intruder-sighting event. Intruder-1 starts in the behaviour try hiding, i.e. it enters the environment safely and hides from the robots. After its minimum hiding time of 30 seconds is over, it runs for an object and is successful in stealing it. Robot-1 starts in patrolling behaviour, changing its behaviour to inspection after 50 seconds under its normal conditions. At the end of inspection duration, Robot-1 finds that one of the objects is missing (the stolen object, event A) and changes its behaviour to patrol. At the same time Intruder-1 finds the robot in its 150-pixel radius. So intruder changes its behaviour to try hiding.

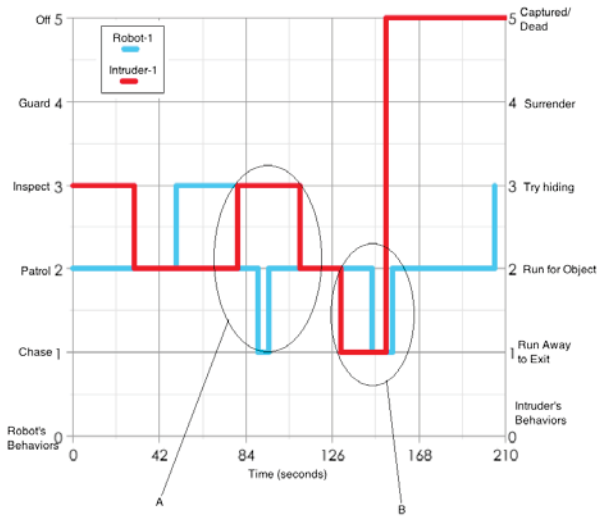


Figure 7 Robot's and intruder's behavior duration

After successfully hiding from the robot, Intruder-1 again changes its behavior to steal an object while the Robot-1 is still patrolling. As the Intruder-1 approaches the object to

steal, it is sighted by the Robot-1 (event B). Robot-1 continues to run behind the Intruder in patrol behavior for minimum commit time period of patrol. As soon as Intruder-1 finds the robot in its 150-pixel radius, instead of pursuing to steal it changes its behaviour to run away to exit. And as the minimum commit time of Robot-1's patrol behaviour is over, it changes its behaviour to chase and captures Intruder-1.

Test 3: Robot's Response to Varying Floor Coverage

Test 3 demonstrates how floor coverage offered by multiple robots changes its cooperative aspects of decision-making about sharing of the overall load. Floor coverage is the amount of surface area possible to cover by the team of robots if each individual robot in it was spread out evenly. Floor coverage and behavior duration impacts a robot's enthusiasm to patrol and inspect.. If floor coverage is high then the enthusiasm to patrol and inspect is low provoking the robots to guard for conserving their energy. The enthusiasm subsystem of robots is used as a means for achieving cooperation and to increase the mission duration by making the robots do behaviours with less energy consumption if more robots are present. The following Figures 8 and 9 with two and three robots respectively, show that as the number of robots in the team increases, the floor coverage increases and robots are made to guard.

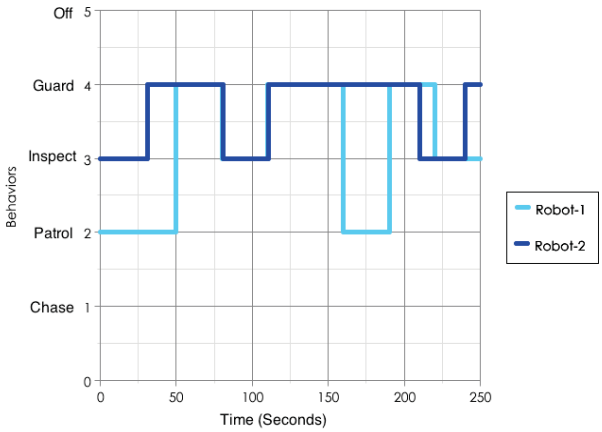


Figure 8 Two robots are being used in Test 3.

Figure 8 shows two robots in the environment. The floor coverage is close to high (moderate or dense), due to which the robots are more prone to guard. Since the floor coverage is not too high, the robots oscillate between guard and inspect behaviours, and occasionally patrol. Figure 9 shows three robots in the environment. The floor coverage is high (dense or crowded), due to which the robots are more prone to guard. Since the floor coverage is high, the robots sometimes fluctuate between guard and inspect behaviours (sometimes patrol also). In such a case, one of the robots engages in patrol and other robots perform only guard or may change to inspection and perform guard again for long time. This case is similar to security guards at the entrance, i.e. to keep watch standing at one point.

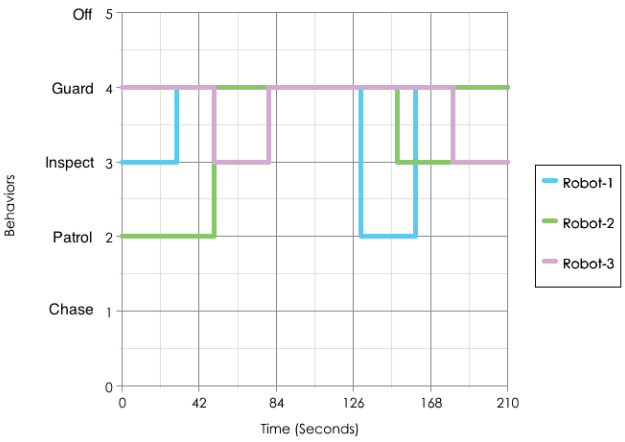


Figure 9 Three Robots are being used

6. Conclusions

The main emphasis of this paper is to maintain security through robot's fuzzy logic decision-making system that is balanced between individual robot's desires and cooperative robot team needs. Maintaining security means the decision-making system has to handle normal conditions when there are no threats and at times when there are threats. It must do this across variable conditions such as areas the team can effectively cover, normal and low battery conditions, etc. Behaviour variation during normal periods by an individual robot was demonstrated, which alternates between patrol and inspection, and occasionally guarding to conserve energy. Indication of an intruder by events such as a missing object leads the robot to become more inclined to patrol or to extend its current patrol duration to vigilant about the environment. The direct sighting of an intruder leads the robot to pursue and attempt to disable the intruder.

When several robots were present the floor coverage was increased so that the robots were able to decide to conserve energy and extend the mission duration by guarding. Also, when energy reserves were low, the robots were seen to become more likely to guard to extend the mission as well. Diversity in behaviour selection across the team was demonstrated. If all robots perform the same behaviour, it will cause a potential hazard. For example, if all robots pursue a sighted intruder, a second intruder could sneak in while the first intruder distracts the robot team. Information sharing by each robot about its current task, allows all other robots to determine which behaviour they should engage in.
In the proposed security system, no robot is told what to do, where to go, or how to proceed yet the behaviours they choose to engage reflects a cooperative effort. Several cases were observed where such decision-making did not always lead to the best results, as when the robot closest to an intruder ceased chasing first in order to diversify the behaviours of the team, however this behaviour was later changed and was eventually corrected when another team member entered the scene.

Acknowledgement
This work (Grants No: 1401001175) was supported in part by Business for Academic-industrial Cooperative establishments funded Korea Small and Medium Business Administration in 2015.

References

[1] H. Everett, "A brief history of robotics in physical security", Space and Naval Warfare Systems Center, San Diego, Robotics Publications, available at http://www.spawar.navy.mil/robots/land/robart/history.html.

[2] R. Arkin, *Behavior-Based Robotics*, MIT Press, Cambridge, Massachusetts, 1998.

[3] R. Murphy, *Introduction to AI Robotics*, MIT Press , Cambridge, Massachusetts, 2000.

[4] Castelnovi, M., Musso, P., Sgorbissa, A., & Zaccaria, R. "Surveillance robotics: analyzing scenes by colors analysis and clustering," Proceedings of the IEEE, vol. 1, pp. 229 – 234, 2003.

[5] K. L. Su, T. L. Chien, J. H. Guo, "Design of low cost security robot applying in family,", Int. Conf. Autonomous Robots and agents, pp. 367-372, Palmerston North, New Zealand, 2004.

[6] T. Born, G. Ferrer, A.M. Wright, and A.B. Wright, "Layered mode selection logic control for border security," Department of Physics, Hendrix College, Conway, AR, and Department of Applied Science, University of Arkansas at Little Rock, Little Rock, AR, 2007.

[7] R. B. Rusu, "Robotux, a multi-agent robot based security system," Int. Conf. Automation, Quality and Testing, Robotics, May 2004, Cluj-Napoca, Romania, 2004.

[8]. M. Tarokh, M. Cross and M. Lee, "Fuzzy logic decision making for multi-Robot security systems," DOI 10.1007/s10462-010-9168-8, May 2010.

SESSION

LOGIC, FUZZY LOGIC, FUZZY CONTROL, BAYESIAN METHODS, AND APPLICATIONS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

On the Possibility of an Event

Daniel G. Schwartz
Department of Computer Science
Florida State University
Tallahassee, Florida 32306, USA

Abstract— *This paper proposes a computational methodology for determining the possibility of a real-world event. It is argued that the notion of possibility for an event is context dependent, where the context of the event comprises prerequisites and/or constraints. Prerequisites enable occurrence of the event, and constraints inhibit its occurrence. It is proposed to compute the possibility of an event as a function of the probabilities that the prerequisites are satisfied and/or the constraints are not satisfied, where the prescribed functions employ the well-known logic of possibility theory. The proposed method is thus a composite of probability and possibility theories.*

Keywords: Possibility theory, Real world event, Prerequisites, Constraints

1. Introduction

Possibility theory was introduced by Lotfi A. Zadeh [10] and subsequently developed at length by Didier Dubois and Henri Prade [2]. Other related works have appeared in the collection edited by Ronald Yager [9]. Zadeh himself has returned to the topic several times, and the subject now enjoys a rich literature.

This work consists primarily of theoretical studies, however, with virtually no applications. A reason for this seems to be that there currently is no established procedure for computing the degree of possibility for a real-world event. Existing approaches use only subjective evaluations. This may be contrasted with probability theory which provides both subjective and computational methods, where the latter are based on the notion of statistical sampling. The objective in this paper is to fill this void.

2. Intuitive Rationale

The Merriam-Webster dictionary gives a simple definition of "possible" as: "(1) able to be done, (2) able to happen or exist, (3) able or suited to be or to become something specified" (http://www.merriam-webster.com/dictionary/probable). For comparison, their simple definition of "probable" is: "(1) the chance that something will happen, (2) something that has a chance of happening, (3) a measure of how often a particular event will happen if something (such as tossing a coin) is done repeatedly" (http://www.merriam-webster.com/dictionary/probability). Regarding the idea of a *degree* of possibility, Zadeh has suggested in messages on the BISC (Berkeley Initiative in Soft Computing) email list that this should be a measure of the "ease" by which something can occur or be done.

Note that these notions of possibility include both things that can be done as acts of human volition and things that occur as a result of forces of nature. Thus any procedure for measuring a degree of possibility should encompass both types of events.

The key idea to be argued here is that the notion of possibility for a real-world event is context dependent, where the context of the event consists of some prerequisites that must be satisfied for the event to occur and/or some constraints that serve to inhibit its occurrence.

It is proposed to compute the degree of possibility for an event as a function of the probabilities that the prerequisites will be satisfied and/or the constraints will not obtain, where this function employs the established logic of possibility theory. In this manner, the proposed computational procedure entails a hybrid of probability theory and possibility theory that exploits the computational features of the former for the purposes of the later.

To illustrate, consider the possibility that Jane will travel to Europe next summer, and suppose that her doing this depends on her having sufficient time and money. Then time and money are prerequisites. It is proposed to compute the degree of possibility for Jane's travel as a function of the probability that she has the necessary time and money, to wit,

$$\mathrm{Poss}(\mathrm{travel}) = \mathit{min}[\mathrm{Prob}(\mathrm{time}), \mathrm{Prob}(\mathrm{money})]$$

where Prob is a standard probability measure such as given by the well-known Kolmogorov axioms [3]. The *min* operation represents the logical "and" in possibility theory.

While the probabilities in this example might not lend themselves to evaluation by statistical samplings, it is clear that in many applications they would. Consider for example the following weather prediction from a local newspaper [1]:

"Severe storms possible ahead of Christmas Day"
"There is a possibility of 2-3 inches of rain."

These possibilities clearly are based on probabilistic models involving such factors as temperature, humidity, cloud formations, and wind currents.

In this spirit, the proposed approach uses the methods of probability theory to determine the likelihood that the prerequisites will be satisfied and then uses the logic of possibility theory to give the degree of possibility for the event. Note that in the example regarding Jane we are here considering only the possibility that she may choose to travel to Europe and not the probability that she actually will go.

Now suppose Jane has learned that a relative is ill and might need her assistance during the same time that Jane plans to travel. This would be a constraint. In this case the foregoing computation becomes

$$\begin{aligned}\text{Poss}(\text{travel}) = \mathit{min}[&\text{Prob}(\text{time}), \text{Prob}(\text{money}),\\ &\text{Prob}(\neg\text{assistRelative})]\end{aligned}$$

or equivalently

$$\begin{aligned}\text{Poss}(\text{travel}) = \mathit{min}[&\text{Prob}(\text{time}), \text{Prob}(\text{money}),\\ &1 - \text{Prob}(\text{assistRelative})]\end{aligned}$$

using the probability theory representation of the logical "not". In effect, $1 - \text{Prob}(\text{assistRelative})$ measures the degree to which the constraint "assistRelative" is mitigated. Note that this amounts to regarding the mitigation of the constraint as a precondition. In general, any constraint c can be construed as specifying a corresponding precondition having the form $\neg c$

These examples illustrate an intuitively plausible means for working with conjunctions of contextual elements (prerequisites and constraints). For a disjunction example, suppose that a certain football coach has two star quarterbacks, Bill and Bob, on his team and believes it is possible to win the next game as long as at least one of these is available to play. Then it would be natural to set

$$\text{Poss}(\text{win}) = \mathit{max}[\text{Prob}(\text{Bill}), \text{Prob}(\text{Bob})]$$

The *max* operation represents the logical "or" in possibility theory. Note that here the probability for each quarterback is only that the player will be available and is not that he actually will play. Whether a particular quarterback plays is a decision resting with the coach.

Now suppose that Bob was injured in a previous game and may not be ready to play for the next one. This introduces a constraint that may be incorporated by revising the foregoing as

$$\text{Poss}(\text{win}) = \mathit{max}[\text{Prob}(\text{Bill}), \text{Prob}(\neg\text{BobInjured})]$$

or equivalently

$$\text{Poss}(\text{win}) = \mathit{max}[\text{Prob}(\text{Bill}), 1 - \text{Prob}(\text{BobInjured})]$$

This same rationale may be extended to more complex combinations of prerequisites and constraints. For example, in the second case of Jane's trip to Europe, suppose that she has decided additionally that she does not want to travel alone and will be willing to go only if at least one of her friends Jill and John agree to accompany her. Then the related formula would be

$$\begin{aligned}\text{Poss}(\text{travel}) = \mathit{min}[&\text{Prob}(\text{time}), \text{Prob}(\text{money}),\\ &1 - \text{Prob}(\text{assistRelative}),\\ &\mathit{max}[\text{Prob}(\text{Jill}), \text{Prob}(\text{John})]]\end{aligned}$$

Note that from the standpoint of pure logic there is no difference between regarding a constraint as such and regarding the negation of the constraint as a prerequisite, for which reason the very consideration of constraints might seem superfluous. Such do play a role however, as a means of making the constraints explicit in the context of the event. In the case of Jane's relative, for example, there would be no reason to even mention the relative if the person was not sick and there was no concern that the relative might need attention.

3. Formalization

These considerations motivate the following formal definitions. For an event E, any proposition p can serve as a prerequisite, and any proposition c can serve as a constraint. Let us define the *contextual constructs* for event E by:

1) If p is a prerequisite for E, then p is a contextual construct for E.
2) If c is a constraint for E, then $(\neg c)$ is a contextual construct for E.
3) If C_1 and C_2 are contextual constructs for E, then $(C_1 \wedge C_2)$ is a contextual construct for E.
4) If C_1 and C_2 are contextual constructs for E, then $(C_1 \vee C_2)$ is a contextual construct for E.
5) Nothing is a contextual construct for E except as required by the items 1 through 4.

Outermost surrounding parentheses will be omitted when not required for disambiguation. A contextual construct either of the form p where p is a prerequisite or of the form $\neg c$ where c is a constraint will be an *atomic* contextual construct.

Given an event E, we may define the *possibility valuation* v for contextual constructs for E by:

1) If C is an atomic contextual construct for E, then $v(C) = \text{Prob}(C)$.
2) If C is of the form $(C_1 \wedge C_2)$ where C_1 and C_2 are contextual constructs for E, then $v(C) = \mathit{min}(v(C_1), v(C_2))$.
3) If C is of the form $(C_1 \vee C_2)$ where C_1 and C_2 are contextual constructs for E, then $v(C) = \mathit{max}(v(C_1), v(C_2))$.

Let us say that a contextual construct for an event E is *complete* if it is considered to be a full description of the relevant context for E in terms of the event's prerequisites and constraints. Then if C is a complete contextual construct for E, set

$$\text{Poss}(E) = v(C)$$

Let us now illustrate this formalism with a return to two of the foregoing examples.

4. Examples Revisited

For a simple case, consider E as the initial event of Jane traveling to Europe next summer. The prerequisites are $p_1 = \textit{sufficient time}$ and $p_2 = \textit{sufficient money}$ and both are required, so the complete contextual construct for E is

$$C = (p_1 \wedge p_2)$$

and the foregoing definitions give

$$\begin{aligned}\mathrm{Poss}(E) &= v(C)\\ &= min(v(p_1), v(p_2))\\ &= min(\mathrm{Prob}(p_1), \mathrm{Prob}(p_2))\end{aligned}$$

Thus one obtains the same result as described in the intuitive rationale.

As a further example, consider the final, more complex case of Jane's travel to Europe. Again call the event E. Here let p_1 and p_2 be as above, let $c = \textit{must assist relative}$, let $p_3 = \textit{Jill goes too}$, and let $p_4 = \textit{John goes too}$. Then a complete contextual construct for E is

$$C = ((p_1 \wedge p_2) \wedge (\neg c)) \wedge (p_3 \vee p_4)$$

Note that there can be more than one complete contextual construct depending on the manner in which these are built up from atomic constructs. For example

$$C' = (((p_4 \vee p_3) \wedge (\neg c)) \wedge (p_2 \wedge p_1))$$

also satisfies our definition of a complete contextual construct.

5. Formalization Continued

Given that there can be more than one complete contextual construct for the same event, the question arises whether all such constructs will evaluate to the same possibility degree. It turns out that there is no guarantee that this will be the case, even given that they employ the same prerequisites and constraints, since the formation of the complete contextual construct depends on how a particular user envisions the logical interrelationships between the prerequisites and constraints.

For this reason it would make sense to define a *context* for an event E as a complete contextual construct for E. In this way, the notion of context is taken to include not only the prerequisites and constraints for E, but also the manner in which these are viewed as being interrelated. Thus the above question becomes one of determining what conditions might be placed on contexts that would ensure that they evaluate to the same possibility degree.

Note that contextual constructs may be regarded as propositions of the classical propositional calculus (CPC) by taking prerequisites and constraints as propositional variables. In this respect it might be reasonable to expect that such contexts should be logically equivalent when thus regarded as propositions of CPC. It happens, however, that this in itself is not sufficient. The reason is that it is not generally true that, if two propositions of CPC are equivalent with respect to that logic, they will also be equivalent (evaluate to the same degree) when interpreted in possibility theory, i.e., where "and", "or", and "not" are interpreted as *min*, *max*, and $1-$. [1] This accordingly raises the question of what additional conditions on the constructs might ensure that they evaluate to the same possibility degree. The following proposes an answer.

First let us review some well-known facts regarding CPC. The language of CPC uses *propositional variables* $\mathbf{p}_1, \mathbf{p}_2, \ldots$, which may be denoted generically by p, q, r, etc., some *connectives* denoted $\wedge$, $\vee$, and $\neg$, and parentheses (and). The *propositions* are defined by:

1) Propositional variables are propositions.
2) If p and q are propositions, then so are $(\neg p)$, $(p \wedge q)$, and $(p \vee q)$.
3) Nothing is a proposition except as required by items 1 and 2.

Parentheses will be dropped when the intended grouping is immaterial or clear; associativity is assumed to be to the right. A *literal* is a proposition of the form p or $\neg p$ where p is a propositional variable. A *clause* is a disjunction of literals, i.e., a formula of the form of $p_1 \vee \cdots \vee p_n$, where the p_i are literals. A *conjunctive normal form* (CNF) is a conjunction of clauses, i.e., a proposition of the form

$$(p_1 \vee \cdots \vee p_{m_1}) \wedge \cdots \wedge (p_n \vee \cdots \vee p_{m_n})$$

where the p_{i_j} are literals. It is well-known that every proposition of CPC has an equivalent CNF. Semantic arguments proving this may be found in [6] and [4] . There also is a well-known algorithm for converting a proposition into a CNF, e.g., cf. [5]. This algorithm may be given in the form of a recursive procedure *conv* as follows. For arbitrary proposition π of CPC,

1) If π is a literal, stop.
2) If π is of the form $\neg\neg p$, replace π with $conv(\pi)$.
3) If π is of the form $\neg(p \wedge q)$, replace π with $conv(\neg p) \vee conv(\neg q)$.
4) If π is of the form $\neg(p \vee q)$, replace π with $conv(\neg p) \wedge conv(\neg q)$.
5) If π is of the form $p \wedge (q \vee r)$, replace π with $(conv(p) \wedge conv(q)) \vee (conv(p) \wedge conv(r))$.
6) If π is of the form $p \vee (q \wedge r)$, replace π with $(conv(p) \vee conv(q)) \wedge (conv(p) \vee conv(r))$.

[1] This was pointed out to me in some exchanges on the BISC (Berkeley Initiative on Soft Computing) email list with Vladik Kreinovich and Dana Scott. Kreinovich notes that the propositions $p \vee \neg p$ and $q \vee \neg q$ are equivalent in classical logic because they are both tautologies, but if p and q are assigned different possibility degrees then the two propositions typically will evaluate differently, e.g., setting $v(p) = 0.4$ and $v(q) = 0.8$ give $v(p \vee \neg p) = 0.6$ and $v(q \vee \neg q) = 0.8$. Scott's example consists of the propositions $(p \vee \neg p) \wedge q$ and $(p \wedge \neg p) \vee q$, which are equivalent in classical logic but evaluate to 0 and 0.5 respectively in possibilistic logic if one sets $v(p) = 0.5$ and $v(q) = 0$.

Item 2 applies the Idempotence Law, items 3 and 4 apply De Morgan's Laws, and items 5 and 6 apply the Distributive Laws. The fact that this conversion process produces an equivalent proposition is established by the fact that these laws are all valid in CPC. It is also known that CPC validates the Commutative and Associative Laws.

Let us say that two propositions p and q of CPC are *strongly equivalent* if $conv(p)$ and $conv(q)$ can be converted into one another using only the Commutative and Associative Laws. In effect, this says that, not only are p and q logically equivalent, but their CNFs differ from one another essentially only in the order of their constituent clauses and the orders of the literals within the clauses. Similarly let us say that two contextual constructs C and C', are *strongly equivalent* if they are strongly equivalent when considered as propositions of CPC.

Next let us note the following well-known facts about possibilistic logic. First, the Commutative and Associative Laws hold true. To wit, for all propositions p, q, r with possibility values $v(p), v(q), v(r)$, we have

$$\begin{aligned} v(p \wedge q) &= min(v(p), v(q)) \\ &= min(v(q), v(p)) \\ &= v(q \wedge p) \end{aligned}$$

and

$$\begin{aligned} v((p \wedge q) \wedge r) &= min(min(v(p), v(q)), v(r)) \\ &= min(v(p), min(v(q), v(r))) \\ &= v(p \wedge (q \wedge r)) \end{aligned}$$

and similarly for $\vee$ with *max*. In addition, the Distributive Laws hold. Consider

$$p \vee (q \wedge r) \text{ and } (p \vee q) \wedge (p \vee r)$$

By examining the six possible arrangements of p, q, and r with respect to their numerical values (i.e., $v(p) \leq v(q) \leq v(r)$, $v(p) \leq v(r) \leq v(q)$, $v(q) \leq v(p) \leq v(r)$, $v(q) \leq v(r) \leq v(p)$, $v(r) \leq v(p) \leq v(q)$, and $v(r) \leq v(q) \leq v(p)$) one can see that in all cases both of the above formulas evaluate to the same number. Similarly for

$$p \wedge (q \vee r) \text{ and } (p \wedge q) \vee (p \wedge r)$$

We are now in a position to establish the following result.

Theorem. If two contextual constructs C and C' are strongly equivalent, then $v(C) = v(C')$.

Proof. Let C and C' be two strongly equivalent contextual constructs. Consider the CNFs $conv(C)$ and $conv(C')$. Note that, because of the rules for building contextual constructs, none of items 2, 3, or 4 in the foregoing algorithm *conv* will need to be applied. In other words, only items 1, 5, and 6 will be employed by the conversion process, and these apply only the Distributive Laws, which we have just seen are valid in possibilistic logic. Thus $v(conv(C)) = v(C))$ and $v(conv(C')) = v(C'))$.

By the definition of strongly equivalent for contextual constructs, each of $conv(C)$ and $conv(C')$ can be converted into the other using only the Commutative and Associative Laws. Since these laws are valid in possibilistic logic, this means that $v(conv(C)) = v(conv(C'))$. This together with the foregoing gives the desired result.

6. Complex Events

The foregoing has defined a notion of possibility for an event whose occurrence is predicated on a set of prerequisites and/or constraints. This in turn gives rise to the issue of an event whose possibility is predicated on the possibilities of some precursor events. More exactly, we here consider an event E whose possibility depends on the possibility of some Boolean combination E' of precursor events $E_1, \ldots, E_n$, where it is assumed that possibility values for $E_1, \ldots, E_n$ are known. The former type of event may be referred to as a *simple event* and the latter as a *complex event*. At issue is how to compute $\mathrm{Poss}(E)$ given $\mathrm{Poss}(E_1), \ldots, \mathrm{Poss}(E_n)$.

It is natural to compute the possibility of the Boolean combination E' in terms of the possibilities of $E_1, \ldots, E_n$ using the standard possibility theory interpretations of $\wedge$, $\vee$, and $\neg$ as the operations *min*, *max*, and $1-$ as discussed in the foregoing. The question, then, is how to determine the possibility of E given the possibility of the Boolean combination E'.

For this it is important to note that the connection between E and E' is a logical relationship. This may be contrasted with probability theory where the conditional probability of E given E' may be interpreted as a causality relation. In possibility theory there is no role for the notion of causation. The possibility relation between E and E' is simply that, if E' is possible, then so is E, i.e., it is a purely logical inference.

Because of this it is natural compute the possibility of E in terms of the possibility of E' by means of some possibilistic extension of the classical rule of Modus Ponens. This amounts to selecting a suitable inference operator, i.e., a functional representation of the logical $\rightarrow$. In the present context it is reasonable to assume that the inference $E' \rightarrow E$ is true, in which case its truth value in a multivalent logic would be 1. Given this, then a value for $\mathrm{Poss}(E)$ can be computed from the value for $\mathrm{Poss}(E')$ according to the function specified for $\rightarrow$.

To illustrate, consider the well-known Łukasiewicz operator (e.g., cf. [7]) defined by

$$v(E' \rightarrow E) = min(1, 1 - \mathrm{Poss}(E') + \mathrm{Poss}(E))$$

In this case, if $v(E \rightarrow E') = 1$, then $1 - \mathrm{Poss}(E') + \mathrm{Poss(E)} \geq 1$, so that $\mathrm{Poss}(E) \geq \mathrm{Poss}(E')$. Thus one obtains a range of values for $\mathrm{Poss}(E)$. Under the current circumstances, however, there is no loss in simply assuming that $\mathrm{Poss}(E) = \mathrm{Poss}(E')$.

But this is only one of many available choices for an inference operator. Numerous other such operators have been discussed throughout the literature. In particular nine different extensions of Modus Pones are discussed in [2] (pp. 130–131) and ten are discussed in [8]. Which one to employ depends on the preferences of the user and the needs of the application.

Note that with the Łukasiewicz operator under the foregoing assumptions the degree of possibility propagates unchanged through chains of inferences. With other versions of logical inference these degrees may degrade as the chain grows longer. In some cases the possibility value can degrade rather quickly toward 0.5, indicating complete uncertainty. This may be suitable for modeling some situations, and not for others.

7. A Note Regarding Applications

Probability theory is concerned with prediction. In this context, one wants to know, given certain conditions, what is the likelihood that a given event will occur. In contrast, possibility theory has no concern with prediction. Rather this addresses the situation where one wants to know, given certain preconditions, what is the possibility that a given event may occur.

Thus, while probability theory lends itself to scientific studies, possibility theory lends itself to planning. In order to plan a course of action leading to some goal, it is of interest to know which plans are possible and, among these, which are more possible than others. To this author's knowledge the opportunities for employing possibility theory in the area of planning have yet to be explored. It seems reasonable that this could have use in organizational planning as well in robotics for automated planning of motion control. The ideas presented in this short paper may be a step in this direction.

8. Conclusion

This paper has proposed a simple but intuitively plausible procedure for computing the degree of possibility of an event. The plausibility rests on the observation that the notion of possibility for an event is context dependent, where the context consists of prerequisites that must be satisfied in order for the event to occur and/or constraints that may inhibit or prevent the event's occurrence. The prerequisites may be satisfied, and the constraints may manifest, with specific numerical degrees of probability. Thus is seems reasonable to compute the possibility of the event in terms of these probabilities.

This has the advantage that one can use the computational methods of statistical sampling to determine the indicated probabilities, and then use these probabilities to determine the event's possibility. Thus the overall method is computational. This augments the currently existing subjective approaches and may serve as the foundation for future applications of possibility theory in practical real-world settings.

It is argued that possibility theory can play an essential role in planning applications inasmuch as it provides a tool for evaluating to possibilities of alternative plans. This opens opportunities to employ possibility theory in areas such as organizational planning and robot motion control.

9. Acknowledgement

Here I would like to express my appreciation to the anonymous reviewer of this paper for the unusually positive remarks and, especially, for requesting that I provide more explanation of the main theorem. It turns out that in attempting to do so I discovered an important oversight in the proof, which led me to revise that section so as to deal with this error. I believe that the present version is in this respect a significant improvement over the previous one, and hopefully this time is also mathematically correct.

References

[1] J. Burlew, Severe storms possible ahead of Christmas Day, *Tallahassee Democrat*, Wednesday, December 23, 2015, p. 3A.
[2] D. Dubois and H. Prade, *Possibility Theory*, Plenum, New York, 1988.
[3] A.N. Kolmogorov, *Foundations of the Theory of Probability*, Chelsea Publishing, New York, 1950.
[4] A. G. Hamilton, *Logic for Mathematicians, Revised Edition*, Cambridge University Press, Cambridge, UK, 1988.
[5] MathStackExchange, How to convert to conjunctive normal form?, http://math.stackexchange.com/questions/214338/how-to-convert-to-conjunctive-normal-form, 2016.
[6] E. Mendelson, *Introduction to Mathematical Logic, Third Edition*, Wadworth and Brookes/Cole, Pacific Grove, CA, 1987.
[7] N. Rescher, *Many-valued Logic*, McGraw-Hill, New York, 1969.
[8] D. G. Schwartz, Fuzzy inference in a formal theory of semantic equivalence, *Fuzzy Sets and Systems*, 31 (1989) 205–216.
[9] R. R. Yager (ed.), *Fuzzy Set and Possibility Theory: Recent Developments*, Pergamon Press, 1982.
[10] L. A. Zadeh, Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets and System*, 1 (1978) 3-28.

Towards an Automated Home Interior Designer System

Aakanksha Bapna[1] **and G. Srinivasaraghavan**[2]
[1,2]Department of Computer Science, IIIT Bangalore, Karnataka, India

Abstract— *We present an almost completely automated intelligent system that produces realistic and aesthetically appealing interior designs for homes. We present the results from our current implementation that generates interior designs for rectangular living rooms. The particularly striking feature of our system is that it generates multiple plausible options for an empty room. The relationships between different elements of a room and items placed in the room are represented as Bayesian networks. The causal relationships defining the network structure are derived from standard thumbrules of interior designing. The parameters for every node in the network are learnt from information extracted semi-automatically from the top view images of furnished living rooms and conversation areas. New layouts based on user inputs are generated upon inferencing from this learnt network. We have successfully dealt with living rooms having (i) dimensions varying from 9 to 25 ft, (ii) 1-3 doors, (iii) 2 windows, (iv) TV or/and fireplace as focal points, (v) dining area, (vi) second conversation area.*

Keywords: Interior designing, Intelligent system, Bayesian networks

1. Introduction

Interior designing is a curious mix of established thumbrules and creativity to shape the experience of the space inside a home, office or even a single room. It requires making the space functional, clutter-free, aesthetically pleasing and comfortable by a combination of creative spatial alignment and cosmetic touches through texture, lighting and colors. All of these are indeed subjective and personal tastes and preferences of the occupants do play a significant role in interior designing. It is therefore essential in interior designing for the designer/architect to work closely with the client and arrive at a design that meets the client (occupant's) needs best. The typical process followed in interior designing is: (i) the user gives loose, high level preferences and requirements, (ii) the architect generates multiple plausible designs that more-or-less meet the stated requirements, while clarifying why some of the preferences if any as stated by the client are inappropriate for various reasons (high cost, violating established best practices, feasibility, etc.), (iii) client makes choices on his/her preferred options, (iv) the architect iterates on the designs by using the client responses and choices as fresh inputs, (v) the final design is arrived at iteratively through a series of interactions between the client and the architect. Much of the discussion in the process described above is visual in nature, involving pictures, drawings and graphical illustrations.

We propose a system that is intended to work a lot like what has been described above and effectively play the role of an interior designer/architect. The system takes basic information such as dimensions and shape (including location of doors, windows, staircase etc.) of the space for which the user wants the interiors designed. It then takes structured input from the user on preferences and constraints (may be an old antique furniture that needs to be accommodated, TV and/or fireplace as focal point for the room, dining area, second conversation area, etc.). The designer's 'knowledge base' is encoded as a Bayesian network learnt from several layouts that have been generated by human designers. The iterative process described above seems to lend itself naturally to a 'Bayesian' interpretation. The user inputs form the evidence for driving nodes of the network with which the system infers the distributions of the locations and attributes of the other nodes in network. Multiple options are generated by sampling from these distributions, with some minimum probability threshold for feasibility. This process can be iterated till the user is happy with a design of his/her choice. We have currently implemented such a system to generate the interiors for a single room (the living room). Our algorithm however can be extended to more complex layouts and multiple spaces naturally.

The main contributions of our work are -

1) object extraction along with their positions and orientations semi-automatically (supported by little manual annotation) from 2d-layout diagrams.
2) a modeling framework entirely in terms of Bayesian networks that creates interior designs only from user requirements.
3) parameter learning in a hybrid Bayesian network with both continuous and discrete nodes.
4) a generic algorithm that can be easily extended to find shape, size, color, texture etc. of the objects and also to generate designs for other rooms.

The third one is, we believe, of independent interest. Hence we focus on the modeling and learning part of the problem which is of primary relevance in the current context.

The rest of the paper is organized as follows. Section 2 is a brief survey of similar attempts at building automated interior designer systems, in the past. Section 3 gives an overview of our system. Sections 4 through 7 describe each of the key conceptual modules in our scheme, an overview of each of which along with a bird's-eye-view of the entire

system is explained in Section 3. Our results are summarized in Section 8. We conclude with some extensions and scope for future work that we foresee in Section 9 followed by a list of references.

2. Previous Works

In the graphics domain, researchers have applied optimization algorithms to the problem of constrained layout synthesis. For example, a parallel tempered Metropolis-Hastings algorithm has been used to synthesize furniture layout [1]. A similar layout problem has been solved using simulated annealing [2]. In these two systems, the designers need to specify the number and types of objects in advance. Another major drawback we see of these systems is that it can handle only spatial arrangement of pieces of furniture selected by the user. In contrast our system generates multiple suggestions for the user even in the choice of furniture, other than the ones that the user wants to preserve. Our system is lot more flexible and can include other features of interior designing such as colors, texture etc. Moreover the rule extraction process is largely manual and is essentially hard-coded into the system. We have implemented a rule extraction algorithm from designs created by human designers and architects, instead. In [2], distance and orientation of every object w.r.t. wall seems to be fixed and specified prior, but that may not always be the case (shown in Figure 1)

In [3] Merrell and others also proposed a method to generate residential building layouts. We also employed Bayesian networks for modeling relationships between different rooms but we derive its structure from standard thumbrules of interior designing (details can be found in section 4). and learn the parameters of the network. [4] developed a tool called *FurnIt* that automatically places furniture into the given floor plan using some predefined hierarchical templates which have certain functionality. This system also has the limitation that it cannot by design handle anything other than spatial arrangements of objects. Also the templates are manually created and constrains the user to a small set of choices implied by the templates.

Apart from the optimization field, many works have been done in the object synthesis domain to render functional designs for indoor scenes. Example-based [5] and activity-centric [6] scene synthesis have been described by Fisher et. al. in recent years. In example based scene synthesis, their algorithm consisted of occurrence model (Bayesian Networks), arrangement model (Mixtures of Gaussians) and contextual categories (clusters). In our system the Hierarchical Bayesian networks handle everything from occurrence, arrangement and interchanging of objects. Also when location and orientation of few objects govern the presence, location , orientation of other objects the results are better.

In activity-centric scene synthesis, when modeling a scene, they first identify the activities supported by a scanned

Fig. 1: Few training images depicting that the distance and orientation of sofas w.r.t. wall and w.r.t. each other is not always same. It is largely dictated by room size and location of doors.

environment and then determine semantically-plausible arrangements of virtual objects. In these two works Bayesian networks were just used to create occurrence models. That implies they have just utilized Bayesian networks to model relationships between discrete variables. Whereas, we used Bayesian networks to model relationships between both discrete and continuous variables. Learning such networks from data is in itself a significant contribution of our work. Also using their algorithm for arranging furniture within a room may lead to only subtle changes in the interior designs. Our work actually creates many functional scenes for empty spaces. After extending our work from 2-d to 3-d layouts, their method can be used to create stylistically different scenes by scanning the different furniture arrangements suggested by our system.

An algorithm to model the structural relationships between objects using graph kernels was given in [7]. Objects or group of objects form the nodes and relationships between them such as enclosure, horizontal, vertical or oblique support form the edges. It is mainly used for scene matching and search purposes. Our system aims to exactly mimic the interior designing process. Their approach of forming a network to represent relationships is good for arranging tables etc. but in case of living rooms when all furniture items are kept on floor,which implies no surface contact and no enclosure of furniture objects, the edges in the graph would be very few. Hence matching could be difficult. Also, we don't just model dependencies between objects, in our case properties of room govern properties of subspaces and then properties of subspaces govern properties of objects. We derive the design in hierarchical manner, hence they are less brittle.

A new algorithm called locally annealed reversible jump MCMC (Markov-chain Monte Carlo) that generates samples from transdimensional distributions encoding complex constraints is described in [8]. This however is fit for

open worlds only where good layouts not only satisfy the constraints of physical plausibility and functionality, but also crowded-ness and appropriateness of the furniture. In open world layouts where the number of objects are not fixed and optimal configurations for different numbers of objects may be drastically different. Also, sampling can make the system very slow as mentioned in [6].

Our proposed system is clearly lot more flexible and generic in its ability to generate a number of 'creative' suggestions for the user.

3. Overview

Figure 2 give a bird's-eye-view of our system. The rectangular boxes represent the major conceptual modules comprising our system and arrows represent dependencies between these modules in the overall workflow. The blue line separates the parts of the system that work offline (one time). The portion below the blue line represents recurring use which essentially involves inferencing from the Bayesian networks with evidence from the user and other constraints to generate design options. The red arrow indicates possible iterations between the inference engine and the user.

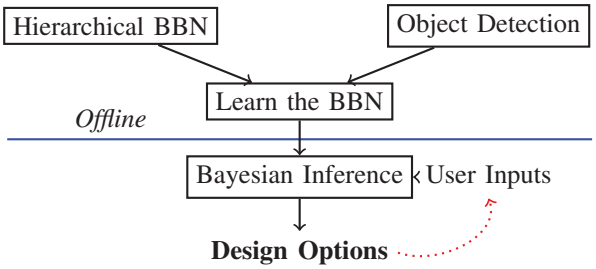


Fig. 2: System Overview

We now describe each of the components shown in the above figure. The system consists of a set of hierarchically defined set of Bayesian Belief Networks (BBN). The structure of these networks is fixed and is derived from assumed causal relationships between the different elements in a living space. The structure and the hierarchy of these BBNs are derived from accepted thumb rules of interior design and interviews with professional interior designers. However the structure of these BBNs encodes only the basic causal relationships between the different elements occupying the space. The causal arrow between a node A and a node B only says that A will have a direct influence on B. The exact nature of this influence is learnt from layouts designed by human architects. The assumption about causal relationships, which is the only set of hard-coded assumptions in the system, we believe are in some sense essential and in any case are much milder than those in the ones the earlier systems we have surveyed.

The object detection module facilitates automatic parameter learning for the BBNs. It detects different object types from standard interior layouts and annotates them with attributes of the objects such as locations and orientations.

The Learning module carries out parameter learning for the BBNs. This required a rather non-trivial extension of existing parameter learning algorithms for Hybird BBNs. The BBNs in our context are hybrid in the sense that some nodes have continuous values (location, orientation) and others take discrete values (yes/no, colors from a set of choices, etc.).

At the end of the learning phase, we have a large hierarchically arranged BBN whose parameters (conditional probability densities) have been learnt. The Inference engine uses the user constraints and preferences as evidence on the network and carries out an inference propagating the evidence through the network to arrive at the target distributions for each node. We describe each of these in detail in the sections that follow.

Note: In all the layout diagrams, the arrows represent doors and the direction of the arrow tells whether it is an 'entry-door' or 'exit-door'. L,T,O,w,l stand for location, type, orientation, width and length of an object respectively.

4. Hierarchically Organized Bayesian networks

The BBN was organized as a hierarchical collection of smaller BBNs and were intended to capture a set of generally accepted causal relationships that mimic the process a typical designer/architect follows during interior design. Some patterns that we observed in the layouts that we studied in the process had a significant influence on the final structure of the network.

Patterns observed in Conversation Area: We observed certain shapes of conversation areas repeating in the training images as shown in Figure 3. We considered 18 different possibilities for conversation area shapes. In all shapes the one edge represents the big sofa. Other edges may be formed by any of the 3-seater, 2-seater, 1-seater or a chair or a combination of any of these.

Patterns observed in room: The images in the Figure 4. show some observations made about the location of doors affecting the conversation area shape in medium sized living rooms. The images in Figure 5 show some observations made in large living rooms which may contain dining area or second conversation areas. Their most occurring locations and sizes are drawn out in the figure.

4.1 BBN Construction

The observations illustrated above played an important role in deciding the structure of Bayesian networks. It also helped in discarding few nodes. The idea is to categorize the living spaces into *conversation area* types. Each conversation area would then drive a different network. For each conversation area type we start with the largest objects (typical

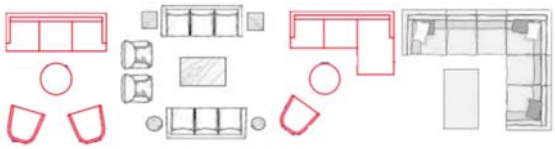


Fig. 3: Some training images of conversation area. (Left to right) Smile shape, C shape, tilted-hut shape, big L shape.

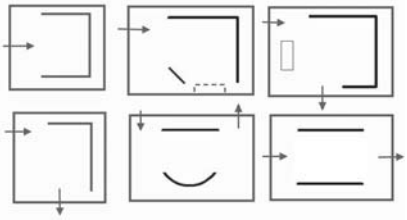


Fig. 4: Examples of different conversation area types based on location of doors observed in living room images.

example would be big sofas). There would be different networks for cases where such large objects exist and those where such large objects do not exist. In case large objects exist, these would drive the placement of the other objects relative to them. Multiplicities are handled by duplicating the nodes appropriately.

We walk through a detailed example of a BBN for a living room. We needed 2 different networks one for describing the relationships between main areas and objects in the room and another for describing the relationships between objects within the conversation area. The following steps are taken to construct the complete network- (i) We first form 2 template networks depicting dependencies between different objects of a room and conversation area based on domain knowledge gained from designers, different websites and observations made in the training set. (a network having nodes- room, conversation area, sofa, table etc.) (ii) Every node in these networks is split into multiple nodes each representing a property of that object (one node each for location, orientation, and size of the object). Therefore all the parents of an object will become the parents of every node representing a property of that node. (iii) It was checked if

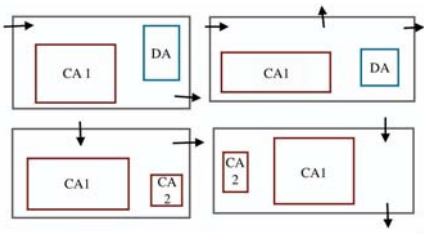


Fig. 5: Examples of different sizes and locations of main conversation area (CA1), dining area (DA) and second conversation area (CA2) observed in living room images.

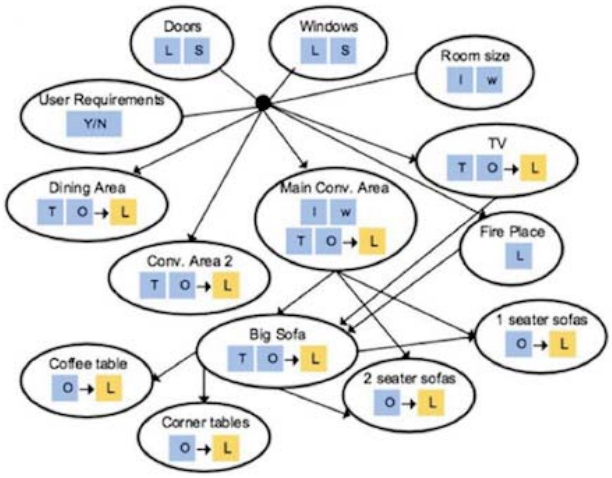


Fig. 6: A model Bayesian network for living room. (conv. means conversation. The squares within a node represent properties of an object. L,T,O,w,l stand for location, type, orientation, width and length of an object respectively. Properties modelled using discrete variables are shown in blue squares and those modelled by continuous variables are shown in yellow.)

some property of a node was independent of some parent (like orientation of sofa wont affect type of table). All properties for all parent and child pairs which intuitively seemed to be unrelated were checked and the corresponding edges were removed. (iv) If a property of an object affected some other properties of the same object then edges were added between them (e.g. orientation of sofa affects its location). (v) If any object in these networks can occur multiple times in a room then there should be multiple instances of it in the network. (like multiple 1-seater sofas). The connection to other nodes, the inter connections between the properties of a node are also copied for all instances.

The variable big sofa type belongs to the conversation area but its value is determined by a few elements of the room. Therefore we created a different network for deciding the big sofa type. After all these steps we obtained 3 networks which are visited in hierarchical top down manner to arrive at the results. First one for capturing the relationships between properties of subspaces and objects of room, second for deciding the type of big sofa. And third for the objects within main conversation area. A condensed form of these 3 networks together is shown in Figure 6.

4.2 Object Nodes and their Properties

Column 1 of Table 1 gives the list of all the objects used to furnish the living room in our system, maximum number of instances of each object which were encountered in the training images is shown in column 2 and the distinct types of each object considered (based on its size or shape) is

Table 1: List of objects used to furnish the living room.

Object	Number of instances	Types (based on shape/size)
Door	4	2
Window	1	1
Fireplace	1	1
TV	1	1
L-shape sofa	1	6
3-seater sofa	2	2
2-seater sofa	2	2
1-seater sofa	4	2
Chair	2	1
Coffee table	1	3
Corner table	4	1
Dining unit	1	2

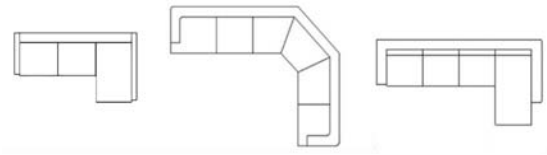


Fig. 7: Small, square and long L-shape sofas used in our system.

shown in column 3. The types of each object class could be based on their shape:

- coffee table — square, rectangular or round
- L-shape sofa — small, square or long (Figure 7)
- dinning table — square or rectangular

or on their size as is the case for doors and sofas. The size of the door matters, because if there's a big door on a small wall hardly anything else can be accommodated, or if the big door is on a larger wall the size of the conversation area decreases.

For all objects except doors and fireplace location is stored using a continuous variable (x,y) and orientation is stored using a discrete variable. We checked the range (out of 8 ranges from $0°$ to $180°$) to which actual orientation belonged to find its discrete orientation. For objects which may or may not appear in the room and that is decided by the user- the property called Y/N is also stored. For other objects it is combined with the discrete variable for orientation, i.e. the last value of orientation representing object is not present. Doors, windows and fireplace are present on the wall therefore their orientation can be determined from their

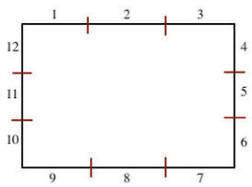


Fig. 8: Discrete locations for doors and fireplace on the wall.

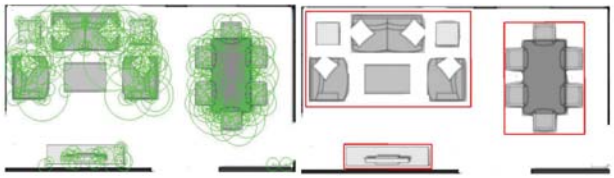


Fig. 9: (a) SURF features detected in a living room image. (b) Conversation area, dinning area, TV matched from templates.

position on the wall and for storing their location just one discrete variable was used. As shown in Figure 8 we divided the wall into 12 sections: one at left, right and centre of every wall. The doors can lie at any location on the wall, but to store its discrete location, it is determined into which of these 12 sections that location falls. The number of the sections was independent of room size. The discrete location was given by one of these 12 numbers. The sizes of room and conversation areas were also discretized.

Note: In our entire procedure the sofas and tables belong to conversation area. The remaining objects and conversation area belong to the room. Big sofa can be any of the 6 L-shape sofas or a 3 or 4-seater sofa.

5. Object Detection and Property extraction from Manual Designs

Figure 1. shows some sample top view images of the living room used for training the Bayesian network. We used top view images of living room as these best exhibit the location and orientation of objects with respect to each other. In 3-d images automatic detection of various kinds of objects was difficult, even if we annotate the objects and bounding boxes manually, automatic extraction of exact relationships between objects was not possible. Nearly 65 living room top view images were gathered from various websites from web. Also, few living room layouts were found in the book 'Time-saver standards for interior designing' [9].

5.1 Extraction within Room

From every living room image main conversation areas, TV and second conversation area, dining area (if present) were cropped and saved separately. Matching these templates (rotated object images) to living room images using SURF (Speeded-Up Robust Features)[10] gives an affine transformation matrix, which can be used to find the orientation (rotation angle $-\pi$ to $-\pi$) and location (transformed centroid-(x,y)) of these objects in the living room. Figure 9 (a) shows the SURF features computed for a living room image, Figure 9 (b) shows the bounding boxes for conversation area, dining area and TV drawn after matching these with the templates. For rest of the objects- doors, windows and fireplace, their discrete location was set manually for training images.

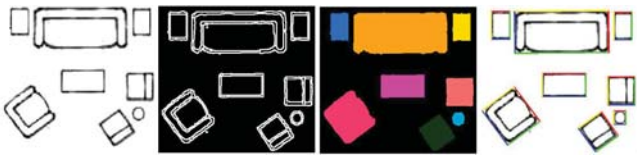


Fig. 10: Outputs after performing different steps of image segmentation on conversation area image.

5.2 Extraction within Conversation Area

The data for conversation areas consisted of images of these cropped from the living room images as mentioned above. We also collected many more images of conversation areas from different sources. The book titled 'Design Rules: The Insider's Guide to Becoming Your Own Decorator' by Elaine Griffin [11] contained many common conversation areas. SURF could not be used here because the objects like sofa and table in the line drawings lack the minimum number of features required for matching. Therefore we performed image segmentation on these images to get the individual objects within a conversation area. Figure 10, (a) shows the actual image of a conversation area from Griffin's book, (b) shows the enhanced image after thresholding and dilation, (c) shows the output image after performing hole-filling, erosion and connected component labelling (every connected component represents an object in the conversation area).

The properties such as centroid, minimum bounding box, length, width, area, aspect ratio, minor and major axis length, perimeter and solidity were extracted for every object. With these region properties as features we used logistic regression to classify the objects into different classes such as 3-seater sofa, 2-seater sofa, 1-seater sofa, square table, rectangular table, round table, tiny table, triangular L-shape sofa, square L-shape sofa etc. We achieved an accuracy of nearly 97% for classifying objects within conversation area when we had only 20 images (training+testing) and 10 object classes. However it dropped to 80% when a total of 108 images were used for training and testing and 17 object classes were considered. It happened mainly because the shapes of same object varied a lot in the bigger data set. The last image in Figure 10. shows the bounding boxes for every object with every edge colored differently. For sofas and chairs, for annotating their front edge, we specified which colored edge is their front edge. Then the angle front edge makes with the positive x-axis was calculated and discretised. To the Bayesian network, centroid and angle are given for training.

6. Learning the Bayesian Networks

All networks elicited in the last section were implemented in *Matlab* using the *Bayes Net Toolbox* (BNT) by Kevin Murphy [12]. BNT was used as it allows multivariate Gaussian (continuous) nodes and mixed (discrete+continuous) networks. It also supports inferencing using junction tree inference engine. But Bayesian parameter learning was available only for pure discrete network. Hence we extended BNT to include generic Bayesian parameter learning for mixed networks.

As per the details about the properties of objects mentioned in the section 4.2 the corresponding nodes were initialised as discrete or continuous. Every continuous node storing the location property was assumed to have a multivariate Gaussian distribution in order to store (x, y) together at a node. The initial means and weights were set to zeros and covariance was set to identity matrix. All other discrete nodes were assumed to have a Dirichlet distribution initialised with random counts. For learning parameters the input data were the values of the properties of every node extracted from the images as explained in section 5. For objects having just one instance in the network the data was given directly, but for objects having multiple instances some criteria were adopted to decide that the data for an object should go to which instance of that object in the network. Some of them are as follows - if 2 instances of 3-seater sofa are present in a room, the first one is always the one forming the top edge in conversation area another one is assigned to second instance in the network. Corner tables for big sofa were checked whether they appeared on the right side or the left side of it and then assigned to respective nodes. For 2 instances of 1 or 2-seater sofas if they have the same x-coordinate then the first one from left is given to first instance of that sofa and the other one to second. And if they have the same y-coordinate then the first one from top is given to first instance of that sofa and the other one to second.

7. Inferring Realistic Designs

Having learnt the networks, the task of filling an empty room is accomplished by performing inference on these networks in a hierarchically top-down manner. The user inputs the dimensions of a room, location of doors and windows in it and a few more requirements. The properties of the objects/subspaces in the room such as TV, dinning area, conversation areas are inferred as distributions conditional on the user choices and the actual placement of the objects (and potentially other stylistic attributes) are decided by sampling from these distributions, with a threshold minimum probability. The inferred properties of main conversation area are passed on to the sub-network. The sub-network predicts type of big sofa which will be most suitable for the given area.

Big sofa type is passed on to the next network for conversation area. And the properties (location, orientation and type) of other sofas and tables that form the conversation area are inferred. If the inference results in none of the big sofas to be present then we use the different version of the third network where the 2-seater (or a similar smaller object) is the main sofa to infer the properties of other

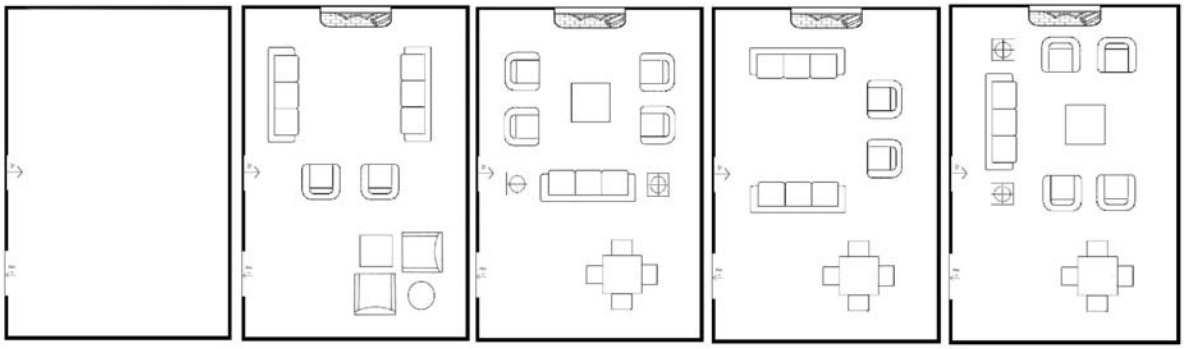


Fig. 11: User inputs: room size -16.7 ft x 24.6 ft, doors at location shown in first image, fire place as focal point, wants to have dining or second conversation area in extra space. 4 different suggestion generated by our system.

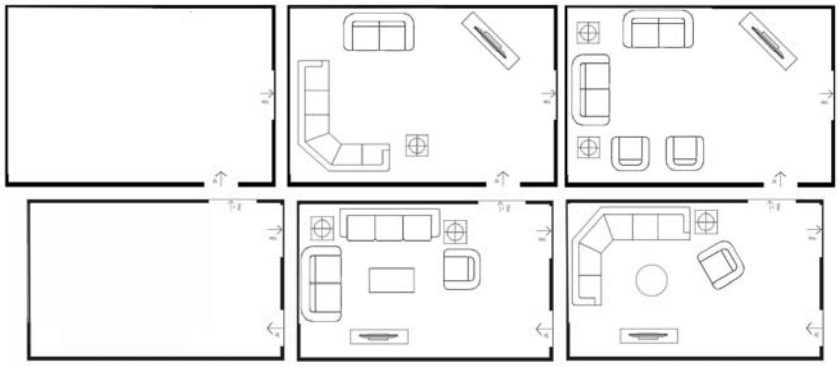


Fig. 12: User inputs: for the room shown in first row: room size-19.4 ft x 13.2 ft and for the one in second row: room size-17.7 ft x 11 ft, doors at location shown in first image, TV as focal point, 2 different suggestion generated by our system for both.

objects. If the inference results in multiple big sofas to be equally possible then different properties of other objects are inferred for every big sofa type to generate multiple designs. Similarly, if different kinds of conversation ares are inferred to be equally probable then different designs are generated upon inferencing from remaining networks with different properties of conversation area as input. If the room is large and the user wants a dining area or second conversation area, it is checked whether that will fit into the room and then its location and orientation are found.

8. Results and Conclusion

Our automated interior designing system successfully carries out the tasks of furniture selection and its positioning at the best possible location and orientation. It was carefully done so as to maintain a balance in the room and leave sufficient space for pathways and circulation within in the room. Our outputs reflect the robustness of our system as it creates aesthetically sound interiors for living rooms having (i) dimensions varying from 9 to 24 ft, (ii) 1-3 doors, (iii) 1-2 windows, (iv) TV or/and fireplace as focal points, (v) dining area, (vi) second conversation area. Figure 11 exhibits multiple interior designs for a big room generated by our system, having different type and orientation of conversation area. Figure 12 exhibits different designs for an average size room having different types of big sofa. All of this is done automatically by our system (as explained in section 7).

These designs generated were completely realistic. The user can readily use our outputs to buy the furniture of appropriate sizes and furnish their living room themselves. Comments by Interior Designer Navni after looking at outputs produced by our system-

"An intelligent system is introduced which is time saving, easy to operate, requires no user intervention or knowledge of interior designing principles, and outputs are multiple. User can discuss with his family various options in just one click. As an Interior Designer, while discussing with the client, the tool is helpful as the layouts generated can be understood well by the client. I really liked the outputs where the same conversation area type is being realized in different rooms by selecting different objects or changing the distance and angle between the objects slightly. Another good point is that the user also comes to know about the seating capacity as he inputs the size of the living room. Problem -The sofas should be at a less distance from the walls so as to get maximum space for movement and circulation in centre. "

We also plan to undertake a more extensive and systematic user study to validate the designs generated by our system and to improve on it.

9. Extensions and Future work

Our current implementation (i) works only for rectangular living rooms, (ii) did not consider all possible objects in a living room. Objects such as showcase, armoire, flower vase, bookcase etc. occur often in living room but were not used in our system since these were not essential elements. Their instances in the training set were scarce, (iii) Number of persons living in that house was not taken into account to calculate the number of seats as no data for that was available, (iv) Sizes were considered only for sofas (only 2 sizes) and shapes for tables only, (v) Color and texture were not considered at all, (vi) no room other than living was considered. (vii) ours is currently a 2-d system that works on floor plans.

Most of these can however be handled with our framework described in this paper. A node to store room shape can be added to deal with non-rectangular rooms. New nodes can be added to store new furniture objects, capacity of the room in terms of the number of persons that can comfortably be accommodated in the room, stylistic choices such as genre of furniture, colors, texture etc., can be considered by adding more properties to the nodes in the network. We believe creating interior designs for multiple rooms would also be a simple extension largely because many of the attributes of the items across different rooms would be almost completely decoupled. Hence conditioned on a common set of user defined stylistic and functional choices that are inherited for the entire design, each room can be more-or-less designed independently of the others.

We plan to extend our work described in this paper to a complete working interior designer system. Though conceptually it seems straightforward, we do anticipate some challenging problems we might encounter in the process, such as: (i) automated detection of objects in a layout and extraction of relationships between these, from a much wider class of available interior layout diagrams, (ii) reducing the level of manual annotation in these diagrams before learning, (iii) handling the computational difficulty associated with much larger networks, particularly hybrid networks of the kind we require, (iv) simulating an iterative interaction between the designer (our system) and the user to arrive at the design most appealing to the user. (v) extending our algorithms to realistic interior designing in 3-d.

Acknowledgements

We would like to thank Prof. Dinesh and Sowmya for their guidance in the field of object detection and recognition. Interior Designer Navni for enlightening us with various interior designing principles and making us aware of popular books on interior designing. People at *livspace.com* for some the living room layouts they provided us.

References

[1] P. Merrell, E. Schkufza, Z. Li, M. Agrawala, and V. Koltun, "Interactive furniture layout using interior design guidelines," *ACM Transactions on Graphics (TOG)-Proceedings of ACM SIGGRAPH 2011*, vol. 30, no. 4, p. 87, 2011.

[2] L.-F. Yu, S.-K. Yeung, C.-K. Tang, D. Terzopoulos, T. F. Chan, and S. J. Osher, "Make it home: automatic optimization of furniture arrangement," *ACM Transactions on Graphics (TOG)-Proceedings of ACM SIGGRAPH 2011, v. 30, no. 4, July 2011, article no. 86*, 2011.

[3] P. Merrell, E. Schkufza, and V. Koltun, "Computer-generated residential building layouts," in *ACM Transactions on Graphics (TOG)*, vol. 29, p. 181, ACM, 2010.

[4] K. A. H. Kjølaas, *Automatic furniture population of large architectural models*. PhD thesis, Massachusetts Institute of Technology, 2000.

[5] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan, "Example-based synthesis of 3d object arrangements," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 135, 2012.

[6] M. Fisher, M. Savva, Y. Li, P. Hanrahan, and M. Nießner, "Activity-centric scene synthesis for functional 3d scene modeling," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 6, p. 179, 2015.

[7] M. Fisher, M. Savva, and P. Hanrahan, "Characterizing structural relationships in scenes using graph kernels," in *ACM Transactions on Graphics (TOG)*, vol. 30, p. 34, ACM, 2011.

[8] Y.-T. Yeh, L. Yang, M. Watson, N. D. Goodman, and P. Hanrahan, "Synthesizing open worlds with constraints using locally annealed reversible jump mcmc," *ACM Trans. Graph.*, vol. 31, pp. 56:1–56:11, July 2012.

[9] J. De Chiara, J. Panero, and M. Zelnik, *Time-saver standards for interior design and space planning*. McGraw-Hill Companies, 1991.

[10] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer vision–ECCV 2006*, pp. 404–417, Springer, 2006.

[11] E. Griffin, *Design Rules: The Insider's Guide to Becoming Your Own Decorator*. Penguin Publishing Group, 2009.

[12] K. P. Murphy, *Bayes Net Toolbox for Matlab*, 1997-2002.

[13] K. P. Murphy, "Inference and learning in hybrid bayesian networks," Tech. Rep. 990, University of California Berkeley, Dept. of Comp. Sci., 1998.

[14] K. P. Murphy, "A variational approximation for bayesian networks with discrete and continuous latent variables," in *In UAI*, pp. 457–466, Morgan Kaufmann, 1999.

[15] R. E. Neapolitan, *Learning Bayesian Networks*. Northeastern Illinois University Chicago, Illinois: Prentice Hall, 2004.

[16] D. Geiger and D. Heckerman, "Learning gaussian networks," in *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, UAI'94, (San Francisco, CA, USA), pp. 235–243, Morgan Kaufmann Publishers Inc., 1994.

[17] D. Heckerman, "A tutorial on learning with bayesian networks," Tech. Rep. MSR-TR-95-06, Microsoft Research, March 1995.

[18] R. Daly, Q. Shen, and S. Aitken, "Review: Learning bayesian networks: Approaches and issues," *Knowl. Eng. Rev.*, vol. 26, pp. 99–157, May 2011.

[19] S. Bottcher, "Learning bayesian networks with mixed variables," *Proceedings of the Eighth International Workshop in Artificial Intelligence and Statistics*, 2001.

[20] R. G. Cowell, "Local propagation in conditional gaussian bayesian networks," *J. Mach. Learn. Res.*, vol. 6, pp. 1517–1550, Dec. 2005.

[21] S. L. Lauritzen and F. Jensen, "Stable local computation with conditional gaussian distributions," *Statistics and Computing*, vol. 11, no. 2, pp. 191–203, 2001.

Improved ABC and Fuzzy Controller Based on Consonant FACTS Devices

H. A. Shayanfar[1*], O. Abedinia [2], N. Amjady[3], Saman Rajaei[2]
[1]Center of Excellence for Power System Automation and Operation, School of Electrical Engineering, Iran University of Science & Technology, Tehran, Iran
[2]Department of Electric Power Engineering Budapest University of Technology and Economics, Budapest, Hungary
[3]Department of Electrical Engineering, Semnan University, Semnan, Iran
hashayanfar@gmail.com, oveis.abedinia@gmail.com, n_amjadi@yahoo.com, saman_persis_rj@yahoo.com

Abstract – *Flexible AC Transmission System (FACTS) devices equipe the probability to control voltages and power flows in a power system and hence to enhance the security of the system. Accordingly, to creat of this possibility, the set amounts of the FACTS controllers have to be selected by appropriate way. In this paper, combination of fuzzy lead-lag control and Improved Artificial Bee Colony (IABC) in multi-machine power system has been proposed which equipped with Thyristor-Controlled Series Capacitor (TCSC) and a Static Var Compensator (SVC). Furthermore, the free parameters of fuzzy controller have been optimized with IABC by considering the time consuming and improving the application of proposed controller structure. To demonstrate the effectiveness of the proposed method multi-machine power system has been considered.*

Keywords: FACTS Devives, Improved Artificial Bee Colony, Fuzzy Control.

1 Introduction

A FACTS is a system composed of static tools used for the AC transmission of electrical systems. It is meant to improve controllability and grow power transfer capability of the network. In this system, the power system stability control is one of the improtant factors in operation. So, once the controller parameters of a dynamic device are optimized to provide the best performance, control conflicts that arise amoung various FACTS controllers may cause the onset of variations [1-9]. Therefore, the coordinated control of the mentioned devices are so important [4]. In this regad, thyristor-controlled series compensators (TCSCs) and static var compensators (SVCs), have been widely studied in the technical literature for improving the power system stability [6]–[8]. Accordingly, in this paper the mentioned devices and proposed controller based on intelligent algorithm has been used to improve the dynamic response of a multi-machine system.

For this purpose, several research works have been presented by researchers. One of the old damping control models is using the washout filter followed by an *m*th order lead-lag controller [10]–[14]. Hence, the pole-zero location algorithm is used for providing the free parameters of a lead lag controller [11], [12], [14]. However, these old models are based on linearized models which can be a big problem for power system solution. To tacke of the mentioned problem, the FACTS control based on fuzzy controller has been presented [16].

In this work, fuzzy controller based on IABC has been proposed which its free parameters have been obtained by proposed intelligent algorithm. Also, the coordination of TCSC and SVC devices in a multi-machine system has been considered. The performance advantage of the FACTS devices bundled with the fuzzy lead-lag device is verified based on comparison with classical lead-lag controllers.

The main advantages of the proposed method are that it is not sensitive to initial parameter values and also not affected by increasing the dimension of a problem. Also this method is strong robustness, fast convergence and high flexibility, fewer setting parameters. The classic version of this algorithm (ABC) has been used for solving multidimensional and multimodal optimization problems. However, the premature convergence in the later search period and the accuracy of the optimal value which cannot meet the requirements sometimes is some of disadvantages of this classic version. For this purpose the interactive ABC is proposed to preventing of this problem. The effectiveness of the proposed method is verified through comparisons with various optimization algorithms.

2 Power System Model

The single-line diagram of the proosed test case including of three machines, three static loads, and an interconnecting network has been presented in Fig. 1. This system includes mutually-coupled transmission lines which quickly connected to an infinite bus [17]. In this system, IEEE type 1 is the excitation system for the synchronous generator. The dynamic behaviors of the generators are defined by the nonlinear one-axis model

* Correspanding Author: hashayanfar@gmail.com (H. A. Shayanfar)

with a rotor reference frame. At a nominal operating point, the nonlinear dynamic equations are first

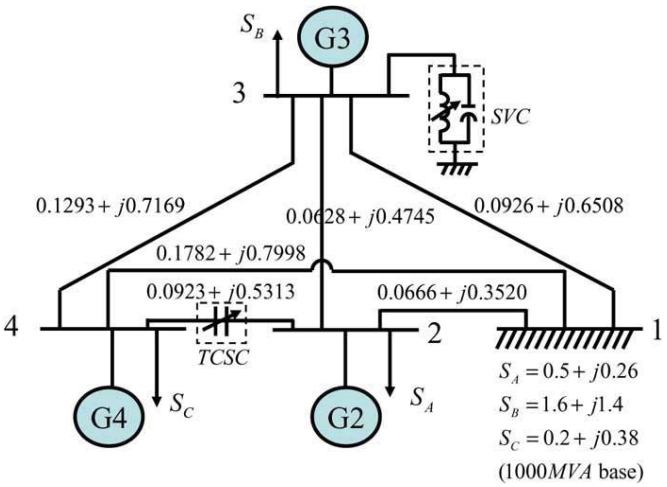


Fig. 1. Three machine power system

linearized to provide a set of linear state relations so we will be able to test the small-signal stability of the system through the eigenvalue analysis. The electromechanical modes are close to the imaginary axis of the complicated plane. The settlment for the three dominant modes needs to be increased, and the requirement for suitable stability equipment to enhance the stability in multi-machine power system is obvious. The TCSC is inserted into the transmission line amoung bus 2 - 4 for controlling the power flow. The SVC is located at the generator bus 3 for obtaining the instantaneous reactive power modulation in voltage keeping.

For presenting the good continuous operating specifications and their suitable application, a fully TCSC system is based on several single-module TCSCs in series to complete the gaps amoung operation modes, is considered. Furthermore, in Fig 3, the model of TCSC has been presented. In this model, three main inputs has been considered for; an auxiliary signal, $X_{auxiliary}$, which can be the input from an external power flow controller, the reference, $X_{reference}$, that is the initial operating point, and small-signal modulation input, $X_{modulation}$, to obtain the stability. Moreover, the block diagram of SVC has been presented in Fig. 4. So, the equal admittance of the TCR can be provided from;

$$B_{L(\alpha)} = \left[\frac{2\pi - 2\alpha + \sin 2\alpha}{\pi}\right] B_{LMAX} \quad (1)$$

Where, B_{LMAX} defines the total admittance of the reactors relative for its rating in MVA. Also, the fix capacitors and the currents of TCR can be shown in two-axis components.

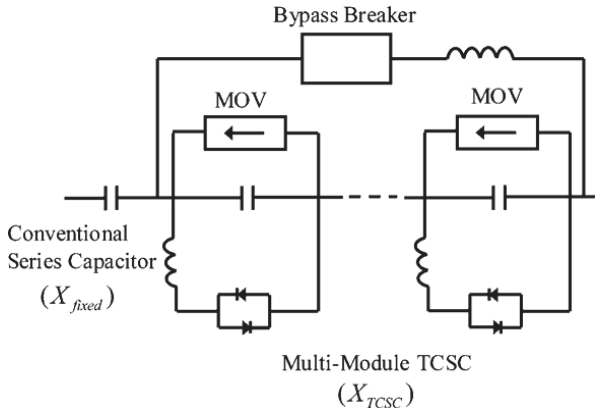


Fig. 2. The TCR SVCs figure of fixed capacitor

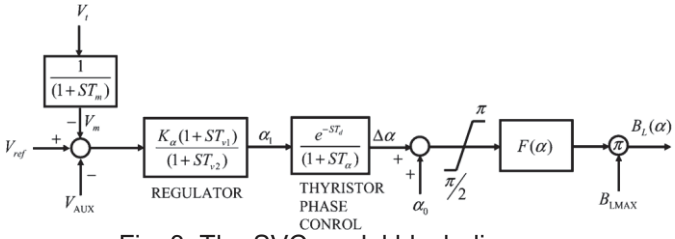


Fig. 3. The SVC model block diagram

The ancillary controllers can give suplementary control signal to the voltage regulator terminal. For an interconnected system with a big number of generators, detailed modeling the system of each individual generator is far also complex to design a controller simply. Simpler models with lower dimension provide more useful and easy undrestanding design process and results.

3 Proposed Fuzzy Control

In wide area control strategy, the selection of control inputs are very important where, the most often used method to select locations and stabilizing signals for power system stabilizer and FACTS block is controllability/observability calculation and analysis according to a linearized time-invariant system model through an obtained operating situation [18], [19]. According to this fact that the proposed controller is applied to the nonlinear system model directly, classical generator rotor speed deviations are considered as control currents of TCR can be shown in two-axis inputs conservatively18].

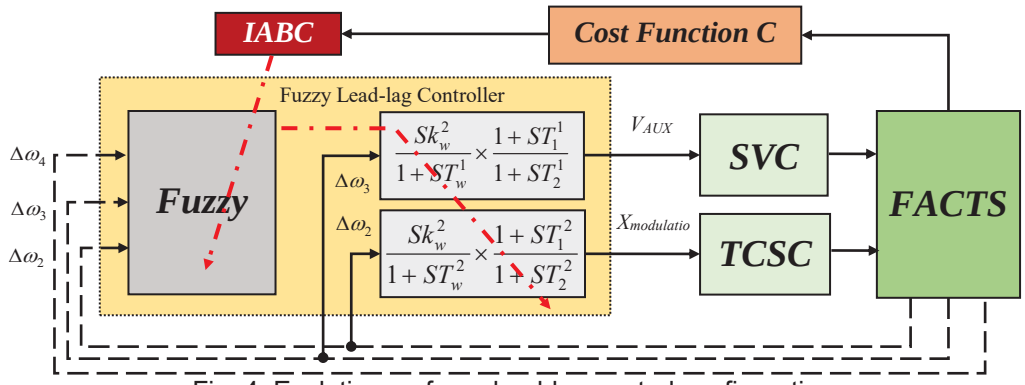


Fig. 4. Evolutionary fuzzy lead-lag control configuration

3.1 Lead-Lag Controller

As mentioned before, two lead-lag controllers have been applied in this work to obtain the suitable control signals to the TCSC and the SVC. The mentioned controllers defined in Fig 4 by the following equations;

$$\frac{V_{AUX}}{\Delta\omega_3}=\frac{SK_\omega^1}{1+ST_\omega^1}\times\frac{1+ST_1^1}{1+ST_2^1} \quad (2)$$

$$\frac{X_{Modulation}}{\Delta\omega_2}=\frac{SK_\omega^2}{1+ST_\omega^2}\times\frac{1+ST_1^2}{1+ST_2^2} \quad (3)$$

Where, the speed deviation in this equation can be described by $\Delta\omega_h(t)=\omega_h(t)-\omega_{ref}(t)$ for generator rotor to the reference speed $\omega_{ref}(t)$. The lead-lag controllers help to the oscillation enhancement and damping ability obtained by TCSC and SVC, thereby providing much higher levels of stable power transfer. To enhance the control application, instead of using fixed free parameter sets such as; $\{T_1^1,T_2^1,T_\omega^1,K_\omega^1\}$ and $\{T_1^2,T_2^2,T_\omega^2,K_\omega^2\}$, in this work we used an intelligent algorithm for providing the optimzal values of these parameters. In the next section the proposed IABC algorithm will be presented.

3.2 Fuzzy Lead-Lag Controller

Actually, three main valiables are important in fuzzy controller such as; $\{\Delta\omega_2(t)\}$, $\Delta\omega_3(t)$, and $\Delta\omega_4(t)$, that defines the speed deviations of rotors in G2, G3, and G4, respectively. This controller is consists of zero-order Takagi-Sugeno (TS)-type fuzzy IF-THEN rules by the following model:

Rule i: IF $\Delta\omega_2(t)$ is A_1^i AND $\Delta\omega_3(t)$ is A_2^i
AND $\Delta\omega_4(t)$ is A_3^i,
THEN T_1^1 is $a_1^i(t)$, T_1^2 is $a_2^i(t)$, T_ω^1 is $a_3^i(t)$, K_ω^1 is $a_4^i(t)$, T_1^2 is $a_5^i(t)$, T_2^2 is $a_6^i(t)$, T_ω^2 is $a_7^i(t)$,
K_ω^2 is $a_8^i(t)$, i=1,…,R

Where, $a_m^i, m=1,...,8,$ is a crisp value, is the total number of rules, and each fuzzy set A_j^i uses a Gaussian membership function, μ_j^i, that is described by

$$\mu_j^i(\Delta\omega_k)=\exp\{-(\Delta\omega_k-m_j^i)^2/(b_j^i)^2\} \quad (4)$$

Where, m^i_j and b^i_j defines the center and width of the fuzzy set A^i_j, respectively. The firing strength which is presented by $\phi^i(\vec{x})$ of the i^{th} rule is computed by;

$$\mu_j^i(\Delta\omega_k)=\exp\{-(\Delta\omega_k-m_j^i)^2/(b_j^i)^2\} \quad (5)$$

The mth output of the proposed fuzzy controller defined as; y_m,m = 1,…,8, and is evaluated by the following weighted average defuzzification equation:

$$y_m=\left(\sum_{i=1}^{R}\phi^i.a_m^i\right)/\sum_{i=1}^{R}\phi^i \quad (6)$$

The rotor speed deviations of all generators are used as controller inputs in order to calculate the damping effect and obtaine the parameters of lead-lag controllers in a same time. Furthermore, a decreased-order nonlinear system model can be calculated from the original power system by association techniques [20]. So, the proposed strategy can be drived.

4 Interactive Artificial Bee Colony

4.1 Review of Artificial Bee Colony

ABC is one of the population based alforithm which inspired from foraging behavior of the honey-bees has been presented by Karaboga and Basturk [21] for optimization problems. The proposed algorithm simulates the intelligent foraging behavior of the honey bee swarms. Furthermore, this algorithm is very simple, robust technique [22].

The colony of artificial bees contains three groups of bees as: employed bees, onlookers and scouts. A bee expecting on the dance section for creation of decision for selecting a food source that is called an onlooker and a bee going to the food source found and selected (as location and etc) by it previously is named an employed bee. In this algorithm, half of the colony conclude of employed artificial bees and the

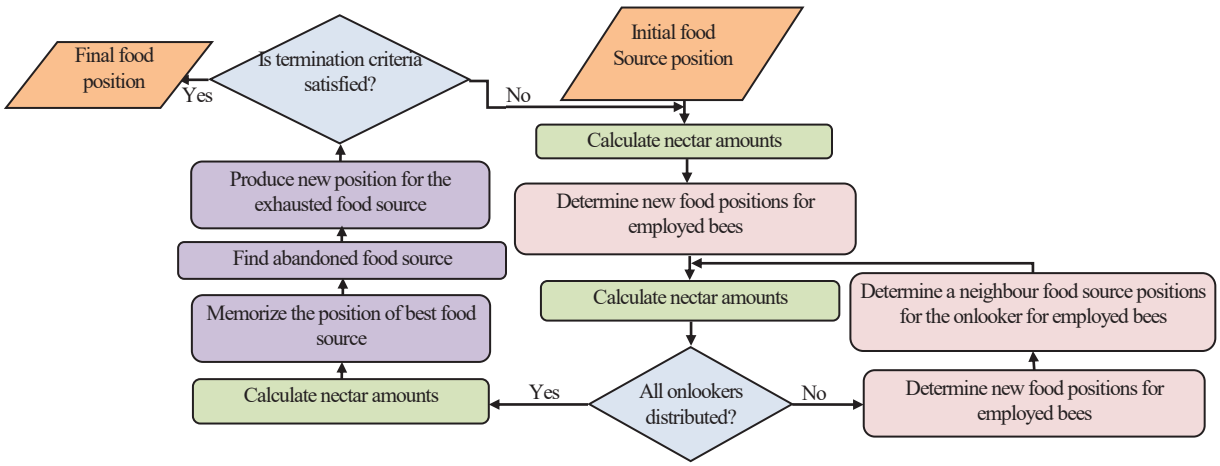


Fig. 5. The flowchart of the proposed ABC algorithm

other half constitutes the onlookers. The main stages of the method can be summarized as [22, 21]:

- *Initialize*
- *REPEAT*
- *Place the employed bees on the food sources in the memory*
- *Place the onlooker bees on the food sources in the memory*
- *Send the scouts to the search area for discovering new food sources*
- *UNTIL (requirements are met)*

Also, the overview of this algorithm is summarized in Fig 5.

4.2 Interactive ABC

An interactive ABC optimization algorithm for stability problem enhancement and finding the optimal values of fuzzy controller has been proposed in this research paper. The model maps the forager bee's path improvement mechanism to pick new coordinates. The forager bee is directed by scout bee which calculates the fitness values of all probable neighboring coordinates. But, in ABC the original design of the onlooker bee's transferring only considers the effects among the employed bee, that is opt by the roulette wheel selection which can not be very strong and accurate, and the one selected randomly. So, it is not strong enough to maximize the exploitation capacity. The IABC is proposed by applying the Newtonian law of universal gravitation for the menioned problem [23]. The universal gravitations among the onlooker bee and the selected employed bees are exploited which is described as:

$$F_{12} = G\frac{m_1 m_2}{r_{21}^2}\hat{r}_{21} \tag{7}$$

Where,

F12 = the gravitational force heads from the object 1 to the object 2

G = the universal gravitational constant

m1, m2 = the masses of the objects

r21 = the separation between the objects

$$\hat{r}_{21} = \frac{r_2 - r_1}{|r_2 - r_1|} \tag{8}$$

In this technique, the mass *m1* is substituted by the parameter. The mass, *m2*, is substituted by the fitness value of the randomly selected employed bee and is denoted by the symbol, *fiti*.

4.3 Fuzzy Mechanism

Upon having the Pareto-optimal set of non-dominated solution, the proposed approach presents one solution to the decision maker as the best compromise solutions. Because of the imprecise nature of the decision maker's judgment, the i^{th} objective function is defined by a membership function μi defined as [22]:

$$\mu_i(p_{gi}) = \frac{f_i^{\max} - f_i(p_{gi})}{f_i^{\max} - f_i^{\min}} \tag{9}$$

Where, f_i^{max} and f_i^{min} are the maximum and minimum values of i^{th} objective, respectively.

$$FDM_i(p_{gi}) = \begin{cases} 0 & \mu_i(p_{gi}) \le 0 \\ \mu_i(p_{gi}) & 0 < \mu_i(p_{gi}) < 1 \\ 1 & \mu_i(p_{gi}) \ge 1 \end{cases} \tag{10}$$

For each non-dominated solution k, the normalized membership function FDM^k

$$FDM^k = \left[\frac{\sum_{i=1}^{2} FDM_i^k(p_{gi})}{\sum_{j=1}^{M}\sum_{i=1}^{2} FDM_i^j}\right] \tag{11}$$

The best compromise solution of stability problem in this pape ris the maximum value of FDM^k as fuzzy decision making function. Where, *M* is the total number of non-dominated solutions. So, all the solutions are evaluated in descending order according to their membership function values which will guide the decision makers with priority list of non-dominated solutions in view of the current operating conditions.

5 Simulation Results

In this section the obtained numerical results is presented which provided by Intel (R) Core i5-2430M CPU 2.40 GHz, running the Windows 7 operating system. Also, sampling interval in all examples was set to 0.001 s.

5.1 Fuzzy Lead-Lag Control

For generating the online training data for application of proposed controller, 0.2 s, a sudden increment of 0.1 p.u. mechanical torque have been considered in this work which occurs simultaneously at all three generators. Then, it can increment disappears after 0.3 s. The whole control duration is 0-9 s. In Fig. 6 the dynamic response of the speed deviations for all generators and the angles of the rotors without application of the TCSC and SVC devices has been presented. The result demonstrate the unstable mood of power system in the lack of the controller.

Beside of this comparison and consideration, application of porposed intelligent technique over fuzzy controller has been drived which the rule number, *R*, was set at 5, the pop size, of this algorithm set to 30 and *T* was set at 10. For considering the trend of proposed algorithm, 1000 iteration has been considered for this procedure. Therefore, there were a total of performance evaluations per run. For statistical analysis, learning was repeated for 30 runs which can be tackle to premature convergence in optimization process. For real-time operation, the proposed controller should send control outputs to the power system within each sampling interval. Hence, the proposed approach is feasible for real-time training. The cost function, *C*, for performance evaluation was calculated based on the root-mean-squared deviations of generator rotor

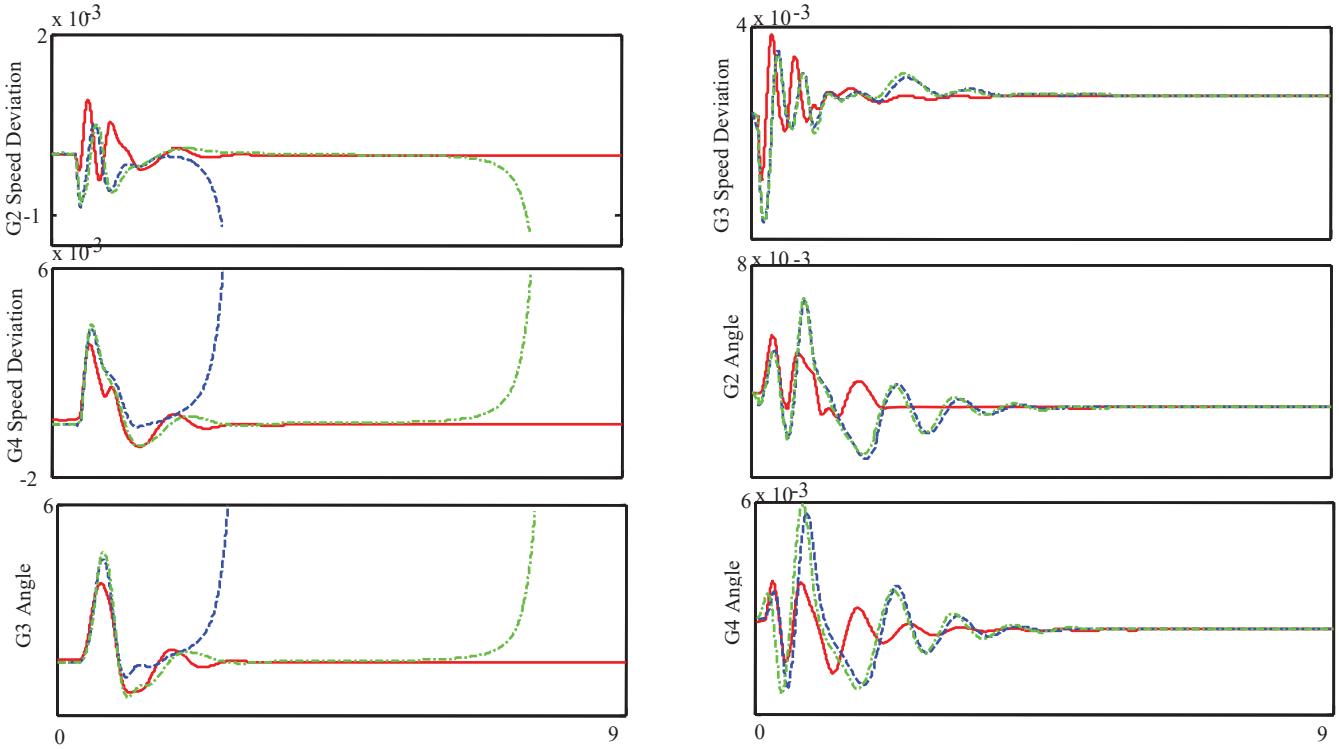


Fig. 6. Dynamic response of rotor speed deviations and angles in G2, G3, and G4 using fuzzy lead-lag controller (solid), lead-lag controller (dashes-dotted), and without control (dashes), where a sudden increment of 0.1 p.u. mechanical torque occurs in G2, G3, and G4 at 0.2 s.

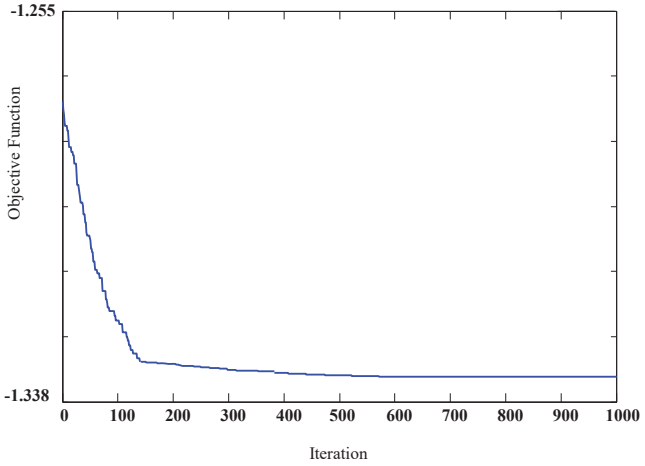


Fig. 7. Averaged best-so-far cost value at each evaluation for evolutionary fuzzy lead-lag control through the porposed algorithm

speeds, $\Delta\omega_i$, and angles, $\Delta\delta_{i,j} = 2,3$ and 4, over 9000 time steps. Regarding to the figure six, angle deviation is about one thousand times the scale of the speed deviation; so it can be said that, weightings c_1 and c_2 were set to a large value of 10^4 to improve system performance especially at low speed deviation amounts, such as 10^4 p.u. The speed deviation of generator G4 depends heavily on those of generators G2 and G3; so, the weighting c_3 was simply set to 1. The weightings c_4, c_5, and c_6 were all originally set to 1 and then reset to 100 for the last 3000 control time steps to enhance the steady state response of the rotor angles. The average average and STD of the cost value, *C*, over the 30 runs has been presented in Table 1.

The result shows that the fuzzy lead-lag controller successfully damps the system oscillations. For quantitative analysis of the control result. Table 2 presents the sum of absolute deviation (SAD) values for both $\Delta\omega_i$ and $\Delta\delta_i$ (i=2, 3, and 4) over the control period which is nine second.

Table. 1. Learning performance of proposed algorithm in comparison with other methods based fuzzy lead-lag controller

Algorithm	ACO_R [24]	ACO_R [24]	RCACO [25	HSA	Proposed
Average	0.04802	0.04743	0.04669	0.04476	0.04324
STD	0.001445	0.00067	0.00044	0.00043	0.00020
t-value	5.636	7.164	1.989	2.25	1.744

Table. 2. Learning performance of proposed algorithm and various modified intelligent techniques based fuzzy lead-lag controller

Algorithm	HPSO-TVAC [26]	HGAPSO [27]	IPSO [28]	Proposed
Average cos	0.047457	0.048321	0.047514	0.04423
STD	0.00780	0.000998	0.000559	0.000214
t-value	6.387	9.687	9.070	6.124

6 Conclusion

This paper presents the combination of two FACTS devices, as SVC and TCSC, for providing the stabilization of amulti-machine in power system. Also, combination of fuzzy lead-lag control and improved artificial bee colony in multi-machine power system has been proposed. Obtained numerical results in various conditions with different torque perturbations verify the oscillation damping ability of the proposed fuzzy lead-lead control. Comparison resulys of porposed method with other well-known algorithms, demonstrate the validity of proposed algorithm.

7 References

[1] O. Abedinia, B. Wyns, A. Ghasemi, Robust Fuzzy PSS Design using ABC, Proceeding of the IEEE International Power & Energy Society Power Systems Conference and Exposition, submitted, Italy, Rome (EEEIC), pp. 100-103, 2011.

[2] O. Abedinia, Mohammad. S. Naderi, A. Jalili, B. Khamenepour, Optimal Tuning of Multi-Machine Power System Stabilizer Parameters Using Genetic-Algorithm, Proceedings of International Conference on Power System Technology, Page(s): 1-6, 24-28 October, 2010.

[3] JM. Call, "Genetic algorithms for modeling and optimization", Journal of Computational and Applied on Mathematics, vol. 184, pp. 205-222, 2005.

[4] H.A. Shayanfar, H. Shayeghi, O. Abedinia, A. Jalili, Design rule-base of fuzzy controller in multi-machine power system stabilizer using genetic algorithm, In: Proceedings of the international conference on artificial intelligence, Las Vegas, Nevada, USA, pp. 49-43, July 2010.

[5] P. Kundur, Power System Stability and Control, 2nd ed. New York, NY, USA: McGraw-Hill, 2008.

[6] Q. Zhao and J. Jiang, "Robust SVC controller design for improving power system damping," IEEE Trans. Power Syst., vol. 10, no. 4, pp. 1927–1932, Nov. 1995.

[7] K. Seethalekshmi, S. N. Singh, and S. C. Srivastava, "Wide-area protection and control: Present status and key challenges," in Fifteenth National Power Systems Conference, Bombay, India, pp. 169–175, Dec. 2008.

[8] B. Singh, N.K. Sharma, A.N. Tiwari, K.S. Verma and S.N. Singh, Applications of phasor measurement units (PMUs) in electric power system networks incorporated with FACTS controllers, International Journal of Engineering, Science and Technology, Vol. 3, No. 3, pp. 64-82, 2011.

[9] G. Shahgholian, A. Rajabi, B. Karimi, "Analysis and Design of PSS for Multi-Machine Power System Based on Sliding Mode Control Theory", International Review of Electrical Engineering (I.R.E.E.), vol 5, Issue. 5, pp: 2241-2250, 2010.

[10]K. R. Padiyar, "Power System Dynamics-Stability and Control", BS Publications, Hyderabad, India, 2002.

[11]H.A. Shayanfar, H. Shayeghi, O. Abedinia, A. Jalili, "Design rule-base of fuzzy controller in multi-machine power

system stabilizer using genetic algorithm", In: Proceedings of the international conference on artificial intelligence, Las Vegas, Nevada, USA, pp: 49-43, July 2010.
[12]E. Mahmoodi, M.M. Farsangi, "Design of stabilizing signals using model predictive control," International Journal on "Technical and Physical Problems of Engineering" (IJTPE), Issue2, Vol. 2, No. 1, pp: 1-4, Mar. 2010.
[13]N.M. Tabatabaei, M. Shokouhian Rad, "Designing power system stabilizer with PID controller", International Journal on "Technical and Physical Problems of Engineering" (IJTPE), Issue. 3, Vol. 2, No. 2, pp: 1-7, June 2010.
[14]O. Abedinia, Mohammad. S. Naderi, A. Jalili, B. Khamenepour, "Optimal tuning of multi-machine power system stabilizer parameters using genetic-algorithm", Proceedings of International Conference on Power System Technology, Hangzhou, China, pp: 1-6, October 2010.
[15]J. Barati, S. S. Mortazavi, A. Saidian, "Analysis and Assessment of FACTS-Based Stabilizers for Damping Power System Oscillations using Genetic Algorithms", International Review of Electrical Engineering (I.R.E.E.), vol 5, Issue. 6, pp: 2819-2827, 2010.
[16]O. Abedinia, Mohammad. S. Naderi, A. Jalili, A. Mokhtarpour, "A novel hybrid GA-PSO technique for optimal tuning of fuzzy controller to improve multi-machine power system stability", International Review of Electrical Engineering (I.R.E.E.), vol. 6, no. 2, pp: 863-873, March-April 2011.
[17]H.B. Duan, D. B. Wang, J. Q. ZHU, X. H. Huang, "New development on ant colony algorithm theory and its applications". Control & Decision, vol. 19, no. 12, pp. 1321 - 1326, 2004.
[18]G. Ma, H, Duan, S. Liu, "Improved Ant Colony algorithm for global optimal trajectory planning of UAV under complex environment", International Journal of Computer Science & Applications, vol. 4 Issue 3, pp: 57-68, 2007.
[19]T. Niknam, "A new HBMO algorithm for multiobjective daily Volt/VAR control in distribution systems considering distributed generators", Applied Energy, vol. 88, pp: 778–788, 2011.
[20]H. Yassami, A. Darabi, S.M.R. Rafiei, "Power system stabilizer design using Strength Pareto multi-objective optimization approach", Electric Power Systems Research, vol. 80, pp: 838-846, 2010.
[21]C. Öztürk, D. Karaboğa, B. Görkemli, "Artificial bee , 2012colony algorithm for dynamic deployment of wireless sensor networks", Turk. J. Elec. Eng. & Comp. Sci., vol. 20, pp. 255-262, 2010.
[22]A. Ö. Bozdoğan, A. E. Yilmaz, M. Efe, "Performance analysis of swarm optimization approaches for the generalized assignment problem in multi-target tracking applications", Turk. J. Elec. Eng. & Comp. Sci., vol. 18, pp. 1059-1078, 2010.
[23]P. W. Tsai, J. S. Pan, B. Y. Liao and S. C. Chu, "Interactive Artificial Bee Colony (IABC) Optimization", ISI2008, Taiwna, Dec. 2008.
[24]K. Socha and M. Dorigo, "Any colony optimization for continuous domain," Eur. J. Oper. Res., vol. 185, pp. 1155–1173, 2008.
[25]C. F. Juang and P. H. Chang, "Designing fuzzy-rule-based systems using continuous ant-colony optimization," IEEE Trans. Fuzzy Syst., vol. 18, no. 1, pp. 18–149, Feb. 2010.
[26]A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," IEEE Trans. Evol. Comput., vol. 8, no. 3, pp. 240–255, Jun. 2004.
[27]C. F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 34, no. 2, pp. 997–1006, Apr. 2004.
[28]F. J. Lin, L. T. Teng, J. W. Lin, and S. Y. Chen, "Recurrent functional-link-based fuzzy-neural-network-controlled inductiongenerator system using improved particle swarm optimization," IEEE Trans. Ind. Electron., vol. 56, no. 5, pp. 1557–1577, May 2009.

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 520 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 his Ph.D from Semnan University, Semnan, Iran in 2015. At present, he is a postdoctoral researcher and adjunct faculty in Budapest University of Technology and Economics. His major research interests are Application of Artificial Intelligence to Power System and Control Design, Forecasting in power system, Distribution Generation, Restructuring in Power Systems, and Optimization.

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.

Saman Rajaei is a Senior year Bachelor's student of Electrical Engineering and informatics with the specialisation of Power Engineering, at Budapest university of Technology and Economics (BME). He has been living in Hungary for the past 4 years where he pursues his degree and is planning to go Canada for his Master's studies. During his Bachelor's studies he has worked with technology companies and won an award for innovation in Wearable technologies in Munich, Germany. His upcoming diploma work is studying the Impact of Electric Car Charging on Low Voltage Smart Grids.

Brain-actuated Control of Wheelchair Using Fuzzy Neural Networks

Rahib H.Abiyev, Nurullah Akkaya, Ersin Aytac, Irfan Günsel, Ahmet Çağman, Sanan Abizade
Near East University, Applied Artificial Intelligence Research Centre, Lefkosa, Mersin-10, North Cyprus
Emails:rahib.abiyev@neu.edu.tr;nurullah@nakkaya.com;

Abstract - *In this paper, a brain-actuated control of the wheelchair for physically disabled people is presented. The design of the system is focused on receiving, processing and classification of the brain signals and then performing control of the wheelchair. The number of experimental measurements of brain activity has been done using human control commands of the wheelchair. Using obtained data including brain signals and control commands the design of classification system based on fuzzy neural networks (FNN) is performed. The structure and learning algorithm of FNN used for brain-actuated control are presented. The training data is used to design the system and then test data is applied to measure the performance of the control under real conditions. The approach used in the paper allows reducing the probability of misclassification and improving the control accuracy of the wheelchair.*

Keywords: Brain-computer interface, wheelchair, fuzzy neural networks, electroencephalogram signal.

1 Introduction

The measuring human brain signal and converting it into control signals needs the development of the interface between the brain and computer and then implementing the control of devices. A brain computer interface (BCI) provides communication between computer and mind of pupils. This interface can be based on brain activity during muscular movements or the changes of the rhythms of brain signals. These brain activities can be detected using electroencephalographic (EEG) signals. BCI transforms the EEG signals produced by brain activity into control signals which can then lasted be used for controlling the wheelchair without using muscles. Since the brain signals are very weak we need to apply some spatial and spectral filters and amplifiers to the EEG signals in order to extract characteristic features of these signals. Several EEG signals can be detected, resulting in different types of BCI. These signals are based on the change of frequencies, change of amplitudes. For example during voluntary thoughts the frequencies of signals are modified, during movement a synchronisation/ desynchronisation 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 wheelchairs. BCI is a control interface that translates human intentions into appropriate motion commands for the wheelchairs, robots, devices, etc. [1] 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. [2] describe a BCI system which control the wheelchair that moves in only one direction- move forward. In [3] a simulated robot is designed that performs two actions- 'turn left then move forward', or 'turn right then move forward'. [4,5] uses three possible commands turn left, turn right and move forward. In [6] BCI is designed using EEG signal captured by eight electrodes. Wavelet transform was used for feature extraction and the radial basis networks were used to classify the predefined movements. In [7] controller based on the brain-emotional-learning algorithm is used to control the omnidirectional robot. [8] presents the design of an asynchronous BCI based control system for humanoid robot navigation using an EEG. [10] 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. [10-15] consider the design of brain controlled wheelchair. The constraction of viable brain–actuated wheelchair that combines brain computer interface with a commercial wheelchair, via a control layer, is considered. BCI allows improving the quality of life of disabled patients and letting them interact with their environment. The processes of feature extraction and classification is very important in BCI design and has a great affect to the performance of the BCI system. Set of research have been done for improvement of the feature extraction and classification algorithms [16-19]. [16,17] considers feature extraction algorithms for Brain-Computer Interfaces.

Recently different clustering algorithms based on support vector machine, linear discriminant analysis, neural networks are applied for classification of brain signals [18]. [19] used features, optimised in the sense of statistically significant and potentially discriminative coherences at a specific frequency and applied linear discriminant for classification purpose. SVM based classification [20] and linear discriminant analysis (LDA) [21] are used for classification purpose of brain signals. [22] uses fuzzy logic and [23] uses neural networks with fuzzy particle swarm optimisation for BGI design. In [24] continuous wavelet transform is used to extract highly representative features and an Adaptive Neuron-Fuzzy Inference System (ANFIS) is used for classification. Fuzzy

logic provides a simple way for determining a conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. Fuzzy Logic's approach to control problems mimics how a person would make faster decisions.

As shown feature extraction and classification plays an important role in designing brain based control for obtaining of high classification accuracy. In BCI design, 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. Different clustering algorithms based on support vector machine, linear discriminant analysis, neural networks are applied for classification of brain signals [18]. Fuzzy classification represents knowledge more naturally to the way of human thinking and is more robust in tolerating imprecision, conflict, and missing information. In this paper, fuzzy neural network is used for the design of BCI in order to achieve efficient brain based control of wheelchair. Signal processing, feature extraction and classification algorithms are designed for brain-actuated control of wheelchair.

2 BCI system architecture

Fig.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 micro-controller that controls the movement of motors. A BCI based control system is usually composed of six main units: signal acquisition unit, signal preprocessing unit, feature extraction unit, classification unit, control action unit and wheelchair motors unit. The brain signals are captured using an emotive headset utilizing 14 channels. These input signals are sent to the signal processing unit. The signals in preprocessing block after filtering and scaling are entered to the feature extraction block. The basic features are extracted and send to the classification system. The output signals of the classification block are motor signals (clusters) that are sent to the wheelchair.

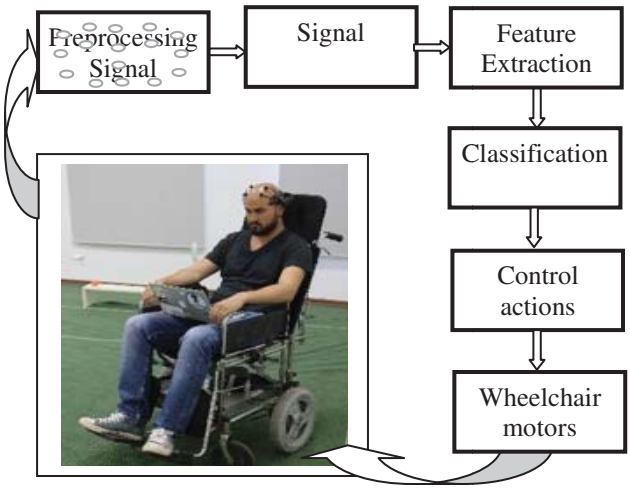


Fig. 1. The BCI based control of the wheelchair

In signal acquisition block the EEG signals are captured using the Emotiv headset. Emotiv EPOC is an EEG Headset which supplies 14 channel EEG data (Fig.2) 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.

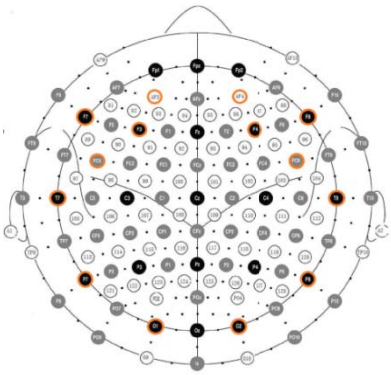


Fig.2. Emotiv's sensor Layout compared to standart 72 sensors layout. The distribution of EEG electrodes. Fourteen channels are marked for data acquisition.

The measured signals are sent to the system input. The signals are very long for processing. Therefore, the feature extraction technique is applied in order to decrease the signal size and extract more important features for classification. In the paper, two different approaches are used for processing of the input sensor signals: With Fast Fourier Transform (FFT) and without FFT. The input signal received from the headset is divided into windows having 2 sec time interval with 50% overlap (Fig.3). The use of overlapping windows allows us to increase the accuracy of the classification. Each two second window corresponds to 256 samples of data. Each second headset returns 128 data samples. The obtained signals from the channels, stored as windows, are then sent to normalisation block. Each channel is normalised 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 normalisation, 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. The expected information gain is the change in information entropy (H) from a prior state to a state that takes some information as given:

$$InfoGain(Class,Frequency) = H(Class) - H(Class \mid Frequency)$$

Information Gain, selects a subset of the original representation attributes according to Information Theory quality metric, Information Gain. This method computes the value of the metric for each attribute, and rank the attributes. Then simply decide a threshold in the metric and keep the attributes with a value over it.

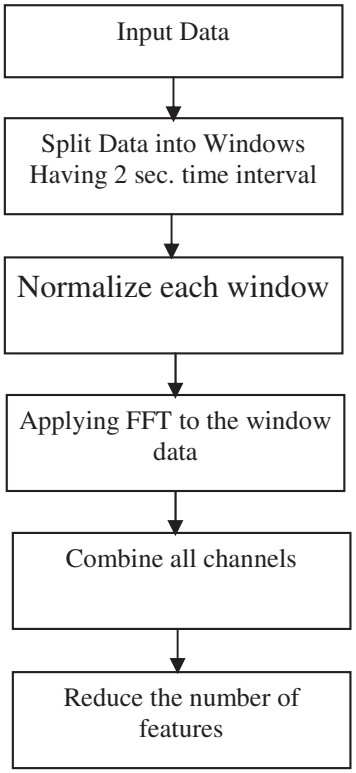


Fig.3. Signal Preprocessing unit.

After frequency representation, all channels in the window are combined in to a single unit so as to apply classification on all channels at once. 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 Fig. 3. In the second approach the acquired brain signal after windowing, normalisation and combining operations are used for classification purpose.

These signals are input for the classification. After the classification the signals the 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. The output of the state machine drives the microcontroller which controls the motors on the wheelchair. The number of classes is equal to the number of control actions.

3 FNN Based Classification

The features extracted from the EEG signals are used for classification and determining control action. In this paper, we propose a novel approach for the classification of brain signals using FNN based classifier. The extracted features are input signals of the FNN based classifier. The classifier based on the above features classifies the signals into the six classes: move forward, move backward, switch on, stop, turn left and turn right. The fuzzy neural system combines the learning capabilities of neural networks with the linguistic rule interpretation of fuzzy inference systems. The design of FNN includes the generation of IF-THEN rules [25-28]. Here, the problem consists in the optimal definition of the premise and consequent part of fuzzy IF-THEN rules for the classification system through the training capability of neural networks, evaluating the error response of the system. There are two basic types of IF-THEN rules used in fuzzy systems. These are Mamdani and Takagi-Sugeno-Kang (TSK) type fuzzy rules. The first one consists of rules, whose antecedents and consequents parts utilize fuzzy values. The second type fuzzy system uses the rule base that has fuzzy antecedent and crisp consequent parts. The second type of fuzzy system approximates nonlinear system with linear systems and has the following form.

If x_1 is A_{1j} and x_2 is A_{2j} and … and x_m is A_{mj} Then

$$y_j = b_j + \sum_{i=1}^{m} a_{ij} x_i \qquad (1)$$

Here x_i and y_j are input and output signals of the system, $i=1,...,m$ is the number of input signals, $j=1...r$ is the number of rules. A_{ij} are input fuzzy sets, b_j and a_{ij} are coefficients.

The structure of fuzzy neural networks used for the classification of EEG signals is based on TSK type fuzzy rules and is given in Fig. 4. The FNN includes six layers. In the first layer, the x_i $(i=1,...,m)$ input signals are distributed. The second layer includes membership functions. Here each node corresponds to one linguistic term. Here for each input signal entering the system, the membership degree to which input value belongs to a fuzzy set is calculated. To describe linguistic terms, the Gaussian membership function is used.

$$\mu 1_j(x_i) = e^{-\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}}, \quad i = 1,...,m, \;\; j = 1,...,r \qquad (2)$$

where m is a number of input signals, r is a number of fuzzy rules (hidden neurons in the third layer). c_{ij} and σ_{ij} are centre and width of the Gaussian membership functions, respectively. $\mu 1_j(x_i)$ is membership function of i-th input variable for j-th term.

The third layer is a rule layer. Here number of nodes is equal to the number of rules. Here R_1, R_2,...,R_r represents the rules. The output signals of this layer are calculated using t-norm min (AND) operation.

$$\mu(x) = \prod_i \; \mu 1_j(x_i), \;\; i=1,..,m, \; j=1,...,r \qquad (3)$$

where Π is the min operation.

These $\mu_j(x)$ signals are input signals for the fifth layer. Fourth layer is a consequent layer. It includes n linear systems. Here the values of rules output are determined.

$$y1_j = b_j + \sum_{i=1}^{m} a_{ij} x_i \tag{4}$$

In the fifth layer, the output signals of the third layer are multiplied by the output signals of the fourth layer. The output of j-th node is calculated as

$$y_j = \mu_j(x) \cdot y1_j$$

In the sixth layer, the output signals of FNN are determined as

$$u_k = \frac{\sum_{j=1}^{r} w_{jk} y_j}{\sum_{j=1}^{r} \mu_j(x)} \tag{5}$$

Here u_k are the output signals of FNN, (k=1,..,n). After calculating the output signal, the training of the network starts.

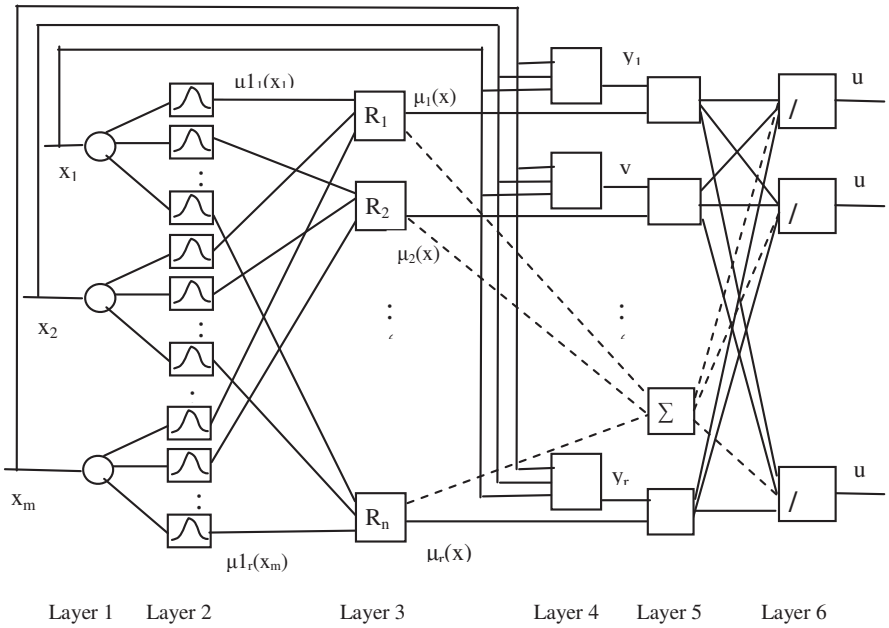


Fig. 4. FNN based identifier

4 Parameter Learning

The design of FNN (Fig. 4) includes determination of the unknown parameters that are the parameters of the antecedent and the consequent parts of the fuzzy if-then rules (1). In the antecedent parts, the input space is divided into a set of fuzzy regions, and in the consequent parts the system behaviour in those regions is described [25-28]. In this paper, the fuzzy clustering is applied to design the antecedent (premise) parts, and the gradient algorithm is applied to design the consequent parts of the fuzzy rules. Fuzzy c-means clustering is applied in order to partition input space and construct antecedent part of fuzzy if-then rules. In the results of partitioning the determined cluster centers will correspond to centers of the membership functions used in input layer of FNN. The width of the membership function is determined using distance between cluster centers. After the design of the antecedents parts by fuzzy clustering, the gradient descent algorithm is applied to design the consequent parts of the fuzzy rules. At the beginning, the parameters of the FNN are generated randomly. To generate a proper FNN model, the training of the parameters has been carried out. For generality we have given the learning procedure of all parameters of FNN using gradient descent algorithm. The parameters are the membership function of linguistic values in the second layer of the network and the parameters of the fourth and fifth layers. Training includes the adjusting of the parameter values. In this paper, we applied gradient learning with adaptive learning rate. The adaptive learning rate guarantees the convergence and speeds up the learning of the network. In addition, the momentum is used to speed-up the learning processes.

At first, on the output of the network, the value of cost function is calculated.

$$E = \frac{1}{2} \sum_{k=1}^{n} (u_k^d - u_k)^2 \tag{6}$$

Here n is the number of output signals of the network, u_k^d and u_k are desired and current output values of the network (k=1,..,n), respectively. The parameters w_{jk}, a_{ij}, b_j, (i=1,..,m, j=1,..,r, k=1,..,n) in consequent part of network and

the parameters of membership functions c_{ij} and σ_{ij} (i=1,..,m, j=1,..,r) of in the premise part of FNN structure are adjusted using the following formulas.

$$w_{jk}(t+1) = w_{jk}(t) - \gamma\frac{\partial E}{\partial w_{jk}} + \lambda(w_{jk}(t) - w_{jk}(t-1));$$

$$a_{ij}(t+1) = a_{ij}(t) - \gamma\frac{\partial E}{\partial a_{ij}} + \lambda(\mathrm{a}_{ij}(t) - a_{ij}(t-1)); \quad (7)$$

$$b_j(t+1) = b_j(t) - \gamma\frac{\partial E}{\partial b_j} + \lambda(\mathrm{b}_j(t) - b_j(t-1));$$

$$c_{ij}(t+1) = c_{ij}(t) - \gamma\frac{\partial E}{\partial c_{ij}} + \lambda(c_{ij}(t) - c_{ij}(t-1));$$

$$\sigma_{ij}(t+1) = \sigma_{ij}(t) - \gamma\frac{\partial E}{\partial \sigma_{ij}} + \lambda(\sigma_{ij}(t) - \sigma_{ij}(t-1)); \quad (8)$$

$$i = 1,...,m;\ \ j = 1,...,r;\ \ k = 1,...,n.$$

Here γ is the learning rate, λ is the momentum, m is the number of input signals of the network (input neurons) and r is the number of fuzzy rules (hidden neurons).

Using equations (7) and (8) the correction of the parameters of FNN is carried out.

Convergence is very important problem in learning of FNN model. The convergence of the learning algorithm using gradient descent depends on the selection of the initial values of the learning rate. The derivation of the convergence is given in [33, 34].

5 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 utilized 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 Fig.5. Fig.5(a) depicts a neutral pose, patient relax not doing anything. Fig.5(b) depicts a positive gesture. As shown in figures, the EEG signals with positive gesture pose are changing more frequently than a neutral pose. In the paper, the FFT is applied to extract important features of the signal. After preprocessing stage, given in section 2, the important features of these signals are extracted and used for classification purpose. The number of extracted features was determined as 100. These signal are inputs for FNN system. Outputs of FNN model are clusters. Six clusters are used in the experiment: Move Backward, Move Forward, Switch on, Stop, Turn Left, and Turn Right. For each cluster, the system recorded 10 seconds of data.

The classification of the EEG signals is performed using FNN model. To synthesis classification model the FNN structure with hundred input- and six output neurons is generated first.

.

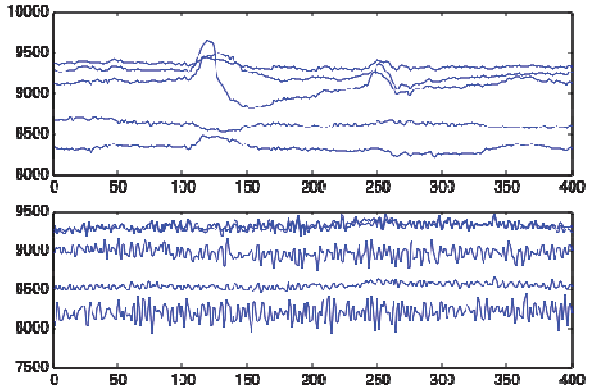


Fig.5. EEG signals for five channels: a) neutral pose, b) positive gesture pose

Fuzzy classification is applied in order to partition input space and select the parameters of the premise parts, that is the parameters of Gaussian membership functions used in the second layer of FNN. Fuzzy c-means clustering is used for the input space with 5 clusters for each input. 5 fuzzy rules are constructed using different combination of these clusters for 100 inputs. After clustering input space gradient decent algorithm is used for learning of consequent parts of the fuzzy rules, that is parameters of the 4-th layer of FNN. In learning of FNN 10 fold cross validation is used for separation the data into training and testing set.

The initial values of the parameters FNN are randomly generated in the interval [-1, 1] and, using the gradient algorithm derived above, they are updated for the given input-output training pairs. As a performance criterion, RMSE is used.

The training is carried out for 1000 epochs. The values of the parameters of the FNN system were determined at the conclusion of training. Once the FNN has been successfully trained, it is then used for the classification of the EEG signals. During learning, the value of RMSE was obtained as 0.223264 for training data, and 0.241625 for evaluation. After learning, for the test data the value of RMSE was obtained as 0.257986 with 100% accuracy of classification. Fig. 6 depicts RMSE values obtained during training. The design of FNN model is performed using a different number of rules. Table 1 includes results of simulations wit 5, 6, 9 and 16 rules respectively. As shown accuracy of FNN classification model are 100%.

For comparison purpose, we test the system using different classification techniques. In the result of classification, the following results are obtained (Table 2). As shown the simulation results demonstrate the efficiency of application FNN model in the classification of EEG signals. These clusters activate the corresponding control signal which is then used to actuate the motors of the wheelchair..

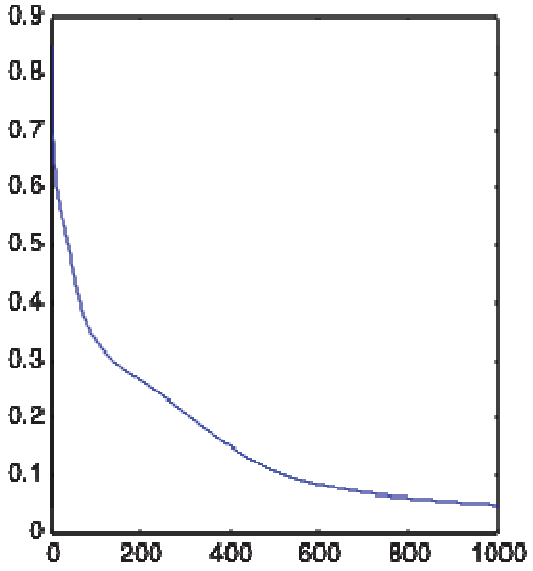


Fig. 6. Training of FNN

Table 1. Classification results.

Number of Rules	Correctly Classified Instances	Incorrectly Classified Instances	Training RMSE	Evaluation RMSE	Test RMSE
5	92%	3	0.465492	0.464918	0.476516
6	100%	0	0.223264	0.241625	0.257986
9	100%	0	0.152714	0.153688	0.153874
16	100%	0	0.047268	0.048324	0.048262

Table 2. Classification results

Method	Correctly Classified Instances	Incorrectly Classified Instances	Mean absolute error	Root mean squared error
SVM	96%	4%	0.2424	0.32
MLP (NN) (6 hidden neurons)	100%	0	0.048	0.0958
Bayesian	94%	6%	0.024	0.1549
Random tree	74%	26%	0.104	0.3225
FNN	100%	0	1.824	0.258022

6 Conclusions

The paper presents BCI based on FNN for the wheelchair. The emotional and muscular states of the user are evaluated for control purpose. The design of BCI has been done to actuate a brain controlled wheelchair using six mental activities of the user: Move Backward, Move Forward, Switch on, Stop, Turn Left and Turn Right. For classification of EEG signals FNN with 10 fold cross validation data set is used. The design of the FNN system is implemented using fuzzy c means classification and gradient descent algorithm. 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 of wheelchair and decrease detection time of the EEG signal used for measuring brain activities and design efficient brain controlled wheelchair.

7 References

[1] 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)

[2] 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)

[3] 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)

[4] 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)

[5] 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

[6] 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.

[7] 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

[8] Yongwook Chae, Jaeseung Jeong, 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

[9] 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

[10] Tom Carlson, Robert Leeb, Ricardo Chavarriaga and Jos´e del R. Mill´an. The Birth of the Brain–Controlled Wheelchair. IEEE/RSJ International Conference on Intelligent Robots and Systems October 7-12, 2012. Vilamoura, Algarve, Portugal, pp. 5444-5445

[11] 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

[12] 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.

[13] 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

[14] 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

[15] Lei Cao, Jie Li, Hongfei Ji, Changjun Jiang. A hybrid brain computer interface system based on the neurophysiological protocol and brain-actuated switch for wheelchair control. Journal of Neuroscience Methods. Volume 229, 30 May 2014, Pages 33–43

[16] F. Lotte, C.T. Guan, "*Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms*", IEEE Transactions on Biomedical Engineering, vol. 58, no. 2, pp. 355-362, 2011

[17] Xiaomu Song, Suk-Chung Yoon. Improving brain–computer interface classification using adaptive common spatial patterns. Computers in Biology and Medicine 61 (2015) 150–160

[18] Lusheng Bi, Xin-an Fan, Yili Liu. EEG-based brain controlled mobile robots: a survey. IEEE Tran. on Human-Machine Systems, V:43,No2,2013.

[19] Rocio Salazar-Varas, David Gutiérrez. An optimized feature selection and classification method for using electroencephalographic coherence in brain–computer interfaces. Biomedical Signal Processing and Control. 18 (2015) 11–18

[20] E. Hortal, D. Planelles, A. Costa, E. Iáñez, A. Úbeda, J.M. Azorín, E. Fernández. SVM-based Brain–Machine Interface for controlling a robot arm through four mental tasks. Neurocomputing, Volume 151, Part 1, 3 March 2015, Pages 116–121

[21] Yonghui Fang, Minyou Chen, Xufei Zheng. Extracting features from phase space of EEG signals in brain–computer interfaces. Neurocomputing 151 (2015) 1477–1485

[22] Mandeep Kaur & Poonam Tanwar. Developing brain computer interface using fuzzy logic. International Journal of Information Technology and Knowledge Management July-December 2010, Volume 2, No. 2, pp. 429-434

[23] Chai R, Ling SH, Hunter GP, Tran Y, Nguyen HT.Brain-computer interface classifier for wheelchair commands using neural network with fuzzy particle swarm optimization. IEEE J Biomed Health Inform. 2014 Sep;18(5):1614-24. doi: 10.1109/JBHI.2013.2295006.

[24] Darvishi S, Al-Ani A. Brain-computer interface analysis using continuous wavelet transform and adaptive neuro-fuzzy classifier. Conf Proc IEEE Eng Med Biol Soc. 2007; 2007:3220-3.

[25] Rahib H.Abiyev. Fuzzy Wavelet Neural Network Based on Fuzzy Clustering and Gradient Techniques for Time Series Prediction. Neural Computing & Applications, Vol. 20, No. 2, pp. 249-259, 2011

[26] Rahib Hidayat Abiyev, Controller based on Fuzzy Wavelet Neural Network for Control of Technological Processes. In proceeding of IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, IEEE CIMSA 2005, pp.215-219, Giardini Naxos - Taormina, Sicily, ITALY , 20-22 July 2005.

[27] Rahib H. Abiyev, Time Series Prediction Using Fuzzy Wavelet Neural Network Model. ICANN-2006. Lecture Notes in Computer Sciences, Springer-Verlag, Berlin Heidelberg, 2006. pp.191-200.

[28] Abiyev R.H, Abiyev V, Ardil C. Electricity Consumption Prediction Model using Neuro-Fuzzy System. Proceedings of the World Academy of Science Engineering and Technology, Vol.8,pp.128-131, Oct 26-28, 2005, Budapest, Hungary.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

AGENT TECHNOLOGIES AND APPLICATIONS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Towards adaptive ex ante circuit breakers in financial markets using human-algorithmic market studies

John Cartlidge
Department of Computer Science, University of Nottingham Ningbo China, Ningbo, Zhejiang, China
john.cartlidge@nottingham.edu.cn

Abstract— *Position paper introducing an AI research methodology for investigating dynamics and stability of financial markets. Controlled experiments using an experimental financial marketplace containing humans and adaptive trading agents are performed; with brain activity of human participants captured using an EEG headset. Machine learning is applied to brain and trading data to develop an adaptive multi-agent model of markets that can be optimised for stability using a co-evolutionary GA framework. Specifically, financial market circuit breakers—mechanisms for limiting or halting trading on an exchange—are addressed; a pertinent real-world problem of trans-national importance. Ex post circuit breakers that halt a market after a sudden price swing will be analysed and optimised. Further, ex ante circuit breakers, which are triggered before problems emerge, will be investigated and explored. The development of reliable "anti-crash" trading technology would have globally significant economic, social, and political impact.*

Keywords: Agent Based Modelling, Brain Computer Interaction, Coevolution, Financial Circuit Breakers, Trading Agents

1. Introduction

In 2012 the UK Government's Office for Science published their final *Foresight* report on the future of computer trading in financial markets [1]. The two-year research project, involving 150 leading academics from more than 20 countries, concluded that introducing coordinated circuit breakers may help reduce extreme adverse price instability; such as witnessed during the "flash crash" of May 6, 2010, when the Dow Jones Industrial Average plummeted 5% (wiping approximately $1 trillion in value) in just 5 minutes [2]. Furthermore, it was suggested that the development of new types of ex ante circuit breakers that are triggered *before* problems emerge would be particularly advantageous [1, p.105].

In September 2015, after a volatile August period that saw widespread losses on China's stock markets, the introduction of an ex post circuit breaker was announced [3]. The breaker, designed to halt trading for half an hour when indexes fall by 5%, went live on January 4, 2016. After a sustained market rout, causing the breaker to trip twice in the space of four days, the breaker was suspended; leading directly to the resignation of the chairman of the China Securities Regulation Commission [4].

These events highlight a fundamental problem facing market regulators. Current understanding of the dynamics of financial systems is woefully inadequate; there is simply no sound theoretical way of knowing what the systemic effect of a structural change will be [5]. Therefore, introducing new regulatory policy amounts to little more than trial and error testing. This is a symptom of the dominant economic modelling paradigms of rational expectations and oversimplified equilibrium models; but a solution is possible. It has been compellingly argued that economic systems are best considered through the paradigm of complexity [6]. Agent-based models present a way to model the financial economy as a complex system [7], while the converging traditions of behavioural [8] and experimental [9] economics can address non-rational human behaviours such as overconfidence and fear using controlled laboratory experiments.

This position paper proposes a novel methodological framework for a hybrid approach to modelling financial systems. Section 2 briefly reviews modelling paradigms and the impact of computerised trading on financial markets. Section 3 introduces the proposed framework. Section 4 outlines financial market circuit breakers as an application area for future work. Finally, Section 5 concludes.

2. Background

2.1 Modelling Economic Systems

Classical economics relies on assumptions such as market efficiency, simple equilibrium, agent rationality, and Adam Smith's invisible hand. These concepts have become so strong that they tend to supersede empirical evidence. Consequently, no classical framework exists to understand "wild" market dynamics such as crashes. To ensure more long-run stability, therefore, it is necessary to develop "a more pragmatic and realistic representation of what is going on in financial markets, and to focus on data, which should always supersede perfect equations and aesthetic axioms" [5].

Disturbingly, there are no mature models used to understand and predict issues of systemic risk in the financial markets [10]; leaving policy makers in the dark. While sophisticated mathematical models exist for modelling potential profit and risk of individual trades, there is "no attempt

to assemble the pieces and understand the behaviour of the whole economic system... When it comes to setting policy, the predictions of... models aren't even wrong, they are simply non-existent" [7]. "Two particularly illuminating questions about priorities in risk management emerge... First, how much money is spent on studying systemic risk as compared with that spent on conventional risk management in individual firms? Second, how expensive is a systemic-risk event to a national or global economy (examples being the stock market crash of 1987, or the turmoil of 1998 associated with the Russian loan default, and the subsequent collapse of the hedge fund Long-Term Capital Management)? The answer to the first question is 'comparatively very little'; to the second, 'hugely expensive"' [6].

Fuelled by disillusionment with classical economics, there is a trend toward alternative economic modelling paradigms designed to overcome the problems of classical economics: (1) *Agent-based models* potentially present a way to model the financial economy as a complex system, while taking human adaptation and learning into account [5], [7], [6], [11]; (2) *Behavioural economics* addresses the effects of social, cognitive, and emotional factors on the economic decisions of individuals. The focus is on non-equilibrium processes and actions of diverse agents with bounded rationality who may learn from experience and interactions [8]; (3) *Experimental economics* is the application of experimental methods to study economic questions. Data collected in experiments are used to estimate effect size, test the validity of economic theories, and illuminate market mechanisms [9].

2.2 Computer Trading in Financial Markets

In recent years, the financial markets have undergone a profound transformation from a highly regulated human-centred system, to a less-regulated and more fragmented computerised system containing a mixture of humans and automated trading systems (ATS)—computerised systems that automatically select and execute a trade with no human guidance or interference. Today, the majority of trades are executed electronically and anonymously at computerised trading venues where human traders and ATS interact. Homogeneous human-only markets have evolved into heterogeneous human-ATS markets, with recent estimates suggesting that ATS now initiate between 30% and 70% of all trades in the major US and European equity markets [1].

As computerisation has altered the structure of financial markets, so too the dynamics have changed. In particular, trading velocity (the number of trades that occur in unit time) has dramatically increased [1]; stocks and other instruments exhibit rapid price fluctuations ("fractures") over sub-second time-intervals [12]; and wide-spread system crashes occur at astonishingly high speed (e.g., the flash crash saw US market indexes plunge 5% in 5 minutes, and then largely recover over the next 20 minutes [2]). The speed and scale of such system dynamics were unprecedented in the pre-computerisation era. To accurately model financial systems, therefore, it is now no longer sufficient to consider human traders only. It is also necessary to model ATS.

3. A New Modelling Methodology

Here, a novel framework is proposed to model financial systems using a synthesis of experimental economics, behavioural economics, and agent based modelling approaches.

3.1 Real-time human-agent markets

Experimental human-only markets have a rich history dating back to Vernon Smith's seminal—and ultimately Nobel-prize winning—1960's research [9]. Smith demonstrated that financial markets can reach theoretical equilibrium through the market mechanism alone, without further assumptions. Several decades later, it was shown that identical markets containing only adaptive ZIP trading agents can also reach equilibrium [13], and can do so *more efficiently* than human only markets [14]. Using a relatively simple reinforcement learning rule to maximise expected profit, ZIP serves as a minimal model of human trading behaviour and has become a standard tool for financial trading simulations. However, despite this long history, relatively few experiments have been performed using mixed markets containing software trading agents *and* humans; and those that have tend to be discrete in nature (for an extensive review, see [15]).

Real-time human-agent markets add a critical dimension of realism that is missing from discrete trading experiments. ATS act at speeds much quicker than the limit of human reaction times. This disparity has been shown to produce a phase transition such that at time scales faster than human reaction times the market enters a regime of computer-only trading that exhibits ultra-fast fractures [12].

Previous work by the author and collaborators has demonstrated that using real-time markets containing a mixture of human participants and financial trading agents holds promise for understanding real-world financial markets. In particular, it has been shown that moving from discrete-time to real-time trading alters the behaviour and efficiency of markets, suggesting that all previous studies using discrete-time experiments may need to be re-evaluated [15]. It has also been demonstrated that faster financial trading agents can reduce market efficiency; a result that may have significant real-world implications [16]. Finally, it has been shown that when trading agents act at speeds quicker than human reaction times, the market starts to fragment towards agent-only interactions and human-only interactions [17]; a result that supports real-world data [12].

3.2 Brain-data for human behavioural models

To extend the methodology used for previous real-time human-agent markets, brain activity of human traders will be captured during trading experiments using an EEG headset. This technology quantifies real-time electrical brain activity

using two dimensions: attention and meditation, and has been used in applications such as controlling adaptive film [18]; and for neuro-adaptive human-computer interfaces [19]. Brain activity data will be captured during times when the market is relatively stable, and during times of extreme volatility (market shocks). Supervised learning methods will be applied to data to build a model of human trading that transitions from logical to more emotional trading behaviour, as market dynamics shift. This approach, akin to the behavioural economics paradigm, will enable a more realistic human-trading model to be developed; and subsequently used within an agent-only market model.

3.3 Coevolution of agents and market

Once an agent-only model has been developed, a competitive coevolutionary genetic algorithm [20], [21] will be used to co-adapt agent models, market structure, and potential regulatory mechanisms. This coevolutionary approach has been successfully applied in discrete agent-only markets; demonstrating the intriguing result that the traditional exchange mechanism (where buyers and sellers have an equal chance of trading) may not be optimal in agent-only markets [22]; although a later result cast some doubt on this [23].

3.4 Open Source Platform

To implement this approach, the author will extend "ExPo: the Exchange Portal", an open-source real-time trading platform for controlled human-agent financial trading experiments [24]. ExPo is a web-application enabling human participants and trading agents to trade over a network. An admin interface, accessible through a web browser, enables the operator to configure experimental variables; including maximum time, supply and demand schedules, number of participants, and agent trading models. ExPo has previously been used to implement real-time versions of discrete trading agents from the literature, and to explore novel Assignment Adaptive (ASAD) trading agents [25]. (For a description of ExPo, see [26].) ExPo will be wrapped in a coevolutionary framework, enabling market mechanisms and trading agents to be automatically adapted using a competitive coevolutionary GA. The extended framework will be released open source; a practice that the author strongly advocates [27].

4. Application: Circuit Breakers

Financial circuit breakers that limit or halt an exchange during times of extreme volatility are used in many major markets. By halting trading during a rapid swing in price, the breaker is designed to stop further movement by "calming" the market. However, this is manifestly not always the outcome. In January 2016, China introduced a circuit breaker that would halt the market for 30 minutes when indexes fall by 5%. The breaker tripped on the first day of trading. But rather than calm the market, the halt in trading caused widespread panic. When trading resumed, further dramatic falls ensued. After a subsequent trip and market halt, the breaker was abandoned and the regulator resigned.

This episode highlights the "irrational" behaviour of market participants and the need for better models of financial markets to test regulatory mechanisms *before* they are introduced into real markets. To achieve this, a series of agent-human financial market experiments will be run under three conditions: (1) a stable market with fixed equilibrium P_0, (2) a shocked market where equilibrium P_0 shifts, and (3) a shocked market containing a simple ex post circuit breaker that halts trading for t seconds after a price swing of ρ percent. Data will be used to test the null hypothesis h_0; that circuit breakers have no effect on market equilibration (defined using Smith's α metric [28]—the root mean squared deviation of transaction prices from theoretical equilibrium, P_0). If h_0 is rejected, then the relationship between t, ρ, and α will be empirically established. Such understanding will immediately offer novel insight into the causal relationship between circuit breakers and market dynamics.

Brain activity data of human participants will be investigated for a change in participants' emotional state—in particular attempting to identify a phase shift in attention and meditation signals—during times of market stress, and during a breaker-triggered market halt. Any changes in trading behaviour that are correlated with a shift in emotional state will be identified and measured. In particular, trading aggressiveness (measured using the proportion and frequency of market orders—those that take the current price; rather than passive limit orders—those that set a target price and wait) will be monitored. Insights into the effects of circuit breakers on the emotional state and corresponding behaviour of traders will be a significant theoretical advance.

To model the emotional behaviour of traders, a reactive "Emo" trading agent will be developed. Emo agents will extend ZIP, such that prevailing market conditions will vary internal ZIP parameters, or shift between "emotional" and "logical" ZIP states. An ensemble of machine learning techniques (neural networks, decision tree learning, reinforcement learning) will be used to fit the adaptive Emo model over data generated during the previously described human-agent experiments. For validation, comparative market experiments will be re-run, replacing human traders with Emo traders. If successful, Emo will be a novel contribution to the literature on experimental and behavioural economics.

Once a validated agent-only (ZIP-Emo) market is available, this will be used within a coevolutionary GA framework to optimise breaker parameters. Using α (and/or other standard metrics, such as profit dispersion and market efficiency [15]) as a metric for market fitness, evolving breakers will be rewarded with high fitness for *low* α (i.e., for equilibration performance), while the market will parasitically co-evolve (exploring agent types, agent parameters, supply and demand schedules, etc.; and rewarded with high fitness for *high* α), in order to pose difficult market scenarios for the

breaker to encounter. To counter-balance any coevolutionary asymmetry present within the system, virulence moderation will be employed [21], [29]. This framework should encourage the exploration and discovery of robust breakers.

Finally, ex ante circuit breakers will be explored. It has previously been shown that there is an intriguing correlation between ultra-fast price swing fractures and the 2010 flash crash [12]. Previous experiments have been able to replicate some of the underlying features of the market that may be linked to fractures [17]. This result holds great promise and prompts further investigation. If a causal link between these phenomena can be identified and understood, then it is possible that an early warning system, or adaptive ex ante circuit breaker can be developed. A further line of investigation will consider neuro-adaptive trading interfaces—graphical approaches that use real-time brain data to warn a trader that they are trading on emotion—and their ability to act as an ex ante circuit breaker.

Circuit breakers in financial markets have been studied using agent-based models elsewhere in the literature, e.g., [30]. However, as far as the author is aware, no other study is using real-time agents, and the behavioural and experimental economics methodology proposed here.

5. Conclusion

A coevolutionary multi-agent methodology for modelling financial markets using real-time human-agent financial market experiments has been proposed. An EEG headset captures human brain activity to identify a phase change during periods of market shock. These data are used to fit adaptive behavioural models of human traders, which are integrated into dynamic agent-based market models to explore and optimise ex post and ex ante circuit breakers using a coevolutionary optimisation approach. This research program has potential for great scientific, policy, and financial impact.

References

[1] GO-SCIENCE, "The future of computer trading in financial markets: An international perspective," UK Government Office for Science, London, UK, Foresight Report, Oct. 2012.

[2] SEC & CFTC, "Findings regarding the market events of May 6, 2010," U.S. Commodity Futures Trading Commission and the U.S. Securities and Exchange Commission, Washington, DC, Report, Sept. 30th 2010.

[3] J. Duggan, "China plans stock market 'circuit breaker' to curb volatility," *The Guardian*, 8 Sept. 2015. [Online]. Available: https://www.theguardian.com

[4] Reuters, "China's stock market regulator 'resigns' after circuit-breaker fiasco," *The Telegraph*, 18 Jan. 2016. [Online]. Available: https://www.thetelegraph.co.uk

[5] J.-P. Bouchaud, "Economics needs a scientific revolution," *Nature*, vol. 455, no. 7217, p. 1181, Oct. 2008.

[6] R. M. May, S. A. Levin, and G. Sugihara, "Complex systems: Ecology for bankers," *Nature*, vol. 451, pp. 893–895, Feb. 2008.

[7] J. D. Farmer and D. Foley, "The economy needs agent-based modelling," *Nature*, vol. 460, no. 7256, pp. 685–686, Aug. 2009.

[8] D. Kahneman, *Thinking, Fast and Slow*. New York, NY: Farrar, Straus and Giroux, 2011.

[9] V. L. Smith, *Papers in Experimental Economics*. Cambridge, UK: Cambridge University Press, 2006.

[10] D. Cliff and L. Northrop, "The global financial markets: an ultra-large-scale systems perspective," UK Gov. Office for Sci., London, Foresight Driver Review—The Future of Computer Trading in Financial Markets—DR4, Aug. 2011.

[11] J. D. Farmer and S. Skouras, "An ecological perspective on the future of computer trading," UK Gov. Office for Sci., London, Foresight Driver Review—The Future of Computer Trading in Financial Markets—DR6, Aug. 2011.

[12] N. Johnson *et al.*, "Abrupt rise of new machine ecology beyond human response time," *Scientific Reports*, vol. 2, no. 2627, pp. 1–7, Sept. 2013.

[13] D. Cliff and J. Bruten, "Minimal-intelligence agents for bargaining behaviours in market-based environments," Hewlett-Packard Labs., Tech. Rep. HPL-97-91, Aug. 1997.

[14] G. Tesauro and R. Das, "High-performance bidding agents for the continuous double auction," in *Proc. ACM Conf. Electron. Commerce*, 2001, pp. 206–209.

[15] M. De Luca, C. Szostek, J. Cartlidge, and D. Cliff, "Studies of interactions between human traders and algorithmic trading systems," UK Gov. Office for Sci., London, Foresight Driver Review—The Future of Computer Trading in Financial Markets—DR13, Sept. 2011.

[16] J. Cartlidge, M. De Luca, C. Szostek, and D. Cliff, "Too fast too furious: Faster financial-market trading agents can give less efficient markets," in *Proc. 4th Int. Conf. Agents & Artificial Intell. (ICAART)*, vol. 2, Vilamoura, Portugal, Feb. 2012, pp. 126–135.

[17] J. Cartlidge and D. Cliff, "Exploring the 'robot phase transition' in experimental human-algorithmic markets," UK Gov. Office for Sci., London, Foresight Driver Review—The Future of Computer Trading in Financial Markets—DR25, Apr. 2012.

[18] M. Pike, R. Ramchurn, S. Benford, and M. L. Wilson, "#scanners: Exploring the control of adaptive films using brain-computer interaction," in *Proc. 2016 CHI Conf. Human Factors in Computing Syst.*, Santa Clara, CA, May 2016, pp. 5385–5396.

[19] A. Girouard *et al.*, "From brain signals to adaptive interfaces: Using fNIRS in HCI," in *Brain-Computer Interfaces*, S. D. Tan and A. Nijholt, Eds. London: Springer, 2010, pp. 221–237.

[20] J. Cartlidge and S. Bullock, "Combatting coevolutionary disengagement by reducing parasite virulence," *Evol. Computation*, vol. 12, no. 2, pp. 193–222, 2004.

[21] J. Cartlidge and D. Ait-Boudaoud, "Autonomous virulence adaptation improves coevolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 15, no. 2, pp. 215–229, 2011.

[22] D. Cliff, "Explorations in evolutionary design of online auction market mechanisms," *Electron. Commerce Research and Applicat.*, vol. 2, no. 2, pp. 162–175, 2003.

[23] ——, "Zip60: Further explorations in the evolutionary design of trader agents and online auction-market mechanisms," *IEEE Trans. Evol. Comput.*, vol. 13, no. 1, pp. 3–18, Feb. 2009.

[24] J. Cartidge *et al.*, "ExPo: The Exchange Portal." [Online]. Available: http://sourceforge.net/projects/exchangeportal

[25] S. Stotter, J. Cartlidge, and D. Cliff, "Exploring assignment-adaptive (ASAD) trading agents in financial market experiments," in *Proc. 5th Int. Conf. on Agents & Artificial Intell. (ICAART)*, vol. 1, Barcelona, Spain, Feb. 2013, pp. 77–88.

[26] ——, "Behavioural investigations of financial trading agents using Exchange Portal (ExPo)," in *Trans. Computational Collective Intell.*, 2014, vol. 17, pp. 22–45.

[27] J. Cartlidge and D. Cliff, "Comparison of cloud middleware protocols and subscription network topologies using CReST, the cloud research simulation toolkit," in *Proc. 3rd Int. Conf. Cloud Computing & Services Sci. (CLOSER)*, Aachen, Germany, May 2013, pp. 58–68.

[28] V. Smith, "An experimental study of comparative market behavior," *J. Political Economy*, vol. 70, pp. 111–137, 1962.

[29] J. Cartlidge and S. Bullock, "Caring versus sharing: How to maintain engagement and diversity in coevolving populations," in *Proc. 7th Eur. Conf. Artificial Life (ECAL)*, Dortmund, Germany, Sept. 2003, pp. 299–308.

[30] S. J. Leal and M. Napoletano, "Market stability vs. market resilience: Regulatory policies experiments in an agent-based model with low- and high-frequency trading," Lab. Econ. Manage. (LEM), Sant'Anna School of Advanced Studies, Pisa, Italy," Work. Paper, Apr. 2016.

An Agent-Based Design for Distributed Artificial Intelligence

James A. Crowder, John N. Carbone
Raytheon Intelligence, Information, and Services, Aurora, CO, USA

Abstract – *Recent EEG brain mapping studies, along with studies utilizing Functional Magnetic Resonance Imaging (FMRI) and Positronic Emission Tomography (PET) scans, suggest that the brain operates as a Distributed Intelligence Processing System (DIPS) [1]. This process, within Artificial Intelligence research, is known as Distributed Artificial Intelligence (DAI) and is intended to mimic human processes for complex learning, planning, and decision making [2]. Here, we present an Intelligent Software Agent (ISA) architecture and design to facilitate distributed learning, processing, and communication within an Artificially Intelligent System (AIS), as well as, spatially separated/ distributed extensions to AI systems.*

Keywords: Distributed Processing, Artificial Intelligence, Agent-Based Processing, Distributed Intelligence

1. Introduction

While there are many different definitions of Artificial Intelligence[1], generally, the challenge with AI is creating H/W and S/W systems that mimic human behavior to some level. There is much discussion and many struggle with the differences between Virtual Intelligence (VI) and Artificial Intelligence (AI), but with both, the aim is to mimic some aspects of human intelligence. One of the issues with AI systems is the differences between the physiology of the human brain and the architecture of computer system utilized to mimic human brain functionality. Table 1 below illustrates these differences. In the past, AI systems have been monolithic, or at least tightly coupled, in nature, possessing physically close processing nodes.

In order to develop an AIS capable of truly mimicking complex human processes, we believe a system of distributed processing elements (intelligent software agents) is required to replicate the brain's DIPS.

Table 1 – Brain vs. AI System Comparison

	Brain	**AI System**
Processing	Millions of slow, simple processors w/ rapid high-fidelity multi-sensory processing	Fast, complex, few processors, generally less sensors, lesser sensory perception fidelity
Memories	Integrated into processing, and is content addressable	Separate from processor, RAM, FLASH, EEPROM, etc.
Computation	Parallel, distributed, self-adapting systems	Sequential, centralized execution, seldom self-adapting
Reliability	Very robust, exhibits gradual degradation, self-correcting, Survival Aware	Fragile, brittle, requires outside intervention for correction. Minimally value agnostic
Storage	Knowledge Economy focused & tied to Survival Awareness	Indiscriminate storage with minimal regard to why and how important

2. ISA Architecture

The notion of an intelligent software agent is not new; having been researched for many decades in the fields of psychology, sociology, and computer science. However, the definition of an ISA has only been seriously researched in the couple of decades [6]. The

[1] The term "Artificial Intelligence" was first used by John McCarthy at Massachusetts Institute of Technology in 1956.

ISA architecture and design discussed here carry the cognitive artifacts for developing understanding required to develop a DIPS AIS capable of complex learning, processing and decision making, within an artificial cognitive framework [4]. Each agent resource running within the system has its interfaces specifically designed to be polymorphic and allow each generalized resource to morph through the use of properties and "–ilities" as the stochastic needs of the system changes over time. The next section describes these ISA properties.

2.1 ISA Properties

For the purposes of this paper, the notion of a software agent is described by the following properties:

1. **Autonomous**: agents can operate without the direct intervention of humans and have control of some (possibly all) of their actions and internal states.
2. **Social**: agents can communicate with other agents and humans via discrete interfaces and/or communications channels.
3. **Reactive**: agents can perceive their environment and respond in a timely fashion to changes that occur.
4. **Proactive**: agents can exhibit goal-directed behavior and hence can take the initiative.
5. **Continuous**: agents are continuously executing processes.
6. **Goal-Oriented**: agents are capable of handling ambiguous, complex, high-level tasks. Operational decisions (splitting tasks into subtasks, sequential ordering of subtasks, etc.) can be handled by the agents.
7. **Mobile**: the agents can move around in an electronic network.
8. **Benevolent**: agents do not have conflicting goals; each agent attempts to accomplish only what is asked of it.
9. **Rational**: agents will act in order to achieve their goals and will not act in ways to prevent the goal from being achieved.
10. **Adaptable**: agents should be able to adjust themselves to changing environments, and to working methods and preferences of the human operators.
11. **Collaborative**: agents should validate instructions to ensure they conform to all conditions, goals, and constraints of the AIS they are contained within.
12. **Asynchronous**: agents must be able to run asynchronously.

While no one ISA will contain all of these properties, the collection of ISAs within a system, should contain the collection of these properties

2.2 An Example ISA Architecture

In our example architecture, there are five agent archetypes, each representing a major functionality type within an overall AIS architecture:

1. **Data Steward Archetype** (DSA): acquires data from a variety of sources and prepares incoming data for use by the AIS.
2. **Advisor Archetype** (ADA): disseminates the right information to the right place at the right time and provides capabilities to allow collaboration agent-agent, and agent-human.
3. **Reasoner Archetype** (REA): interacts with the DSA and ADA agents and utilizes micro services, analytics, ontologies and lexicons to automate the development of domain-specific enrichment. Provides question answering and relevant source information to provide answer thresholds to problems posed within the AIS.
4. **Analyst Archetype** (ANA): these agents are fed by the REAs and utilize the developed ontologies and lexicons to expand upon questions and answered learned from collected information (inferencing).
5. **Interface Archetype** (INA): assesses the correctness of major decisions and adjusts the decision processes within the ADAs. INAs also accommodate internal and external system and human-in-the-loop structures.

Each agent Archetype utilizes a core set of micro-services, from which an agents capabilities can be defined. Figure 1 illustrates an example set of initial agent services. Our AIS DIPS architecture allows agents to operate independently, but in coordination and collaboration with other agents within the AIS to achieve system-wide goals and comply with the overall given directives [9].

The agents within the AIS architecture are not mobile in the sense that agent code moves from server-to-server, however the components within them are

mobile. Figure 2 below illustrates the minimum services required for each agent archetype.

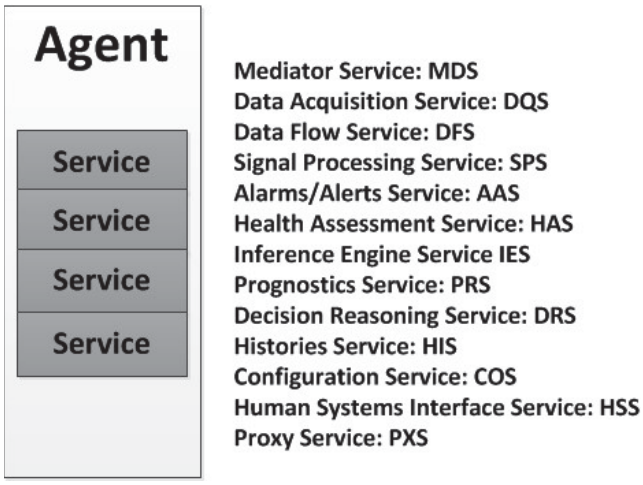


Figure 1 – Basic Set of Available Agent Services

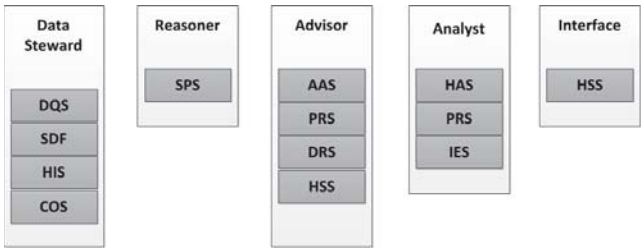


Figure 2 – Minimum Required Archetype Services

There may be other services within each agent archetype, however Figure 2 illustrates the minimum services required to define each agent archetype. Each ISA service shown in Figure 1 Figure 2 is composed of small, discrete, pluggable modules, are nodes. These nodes define the service capabilities within the ISAs. The nodes provide the modularized service architecture required for an AIS DIPS processing framework. Figure 3 illustrates an example of basic Nodes available to each service; with each Node providing functions/capabilities specific to that service.

As explained, nodes are micro-service capabilities. A family of nodes constitutes an ISA Service Plugin. Figure 4 illustrates the minimum nodes required for a service within an ISA to be called a particular service type. Again, there can be many more Nodes that specified here, this simply illustrates the minimum set for each specified service in Figure 1.

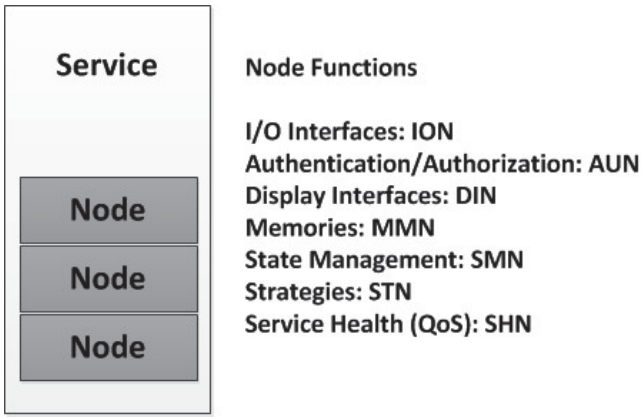


Figure 3 – Basic ISA Service Nodes

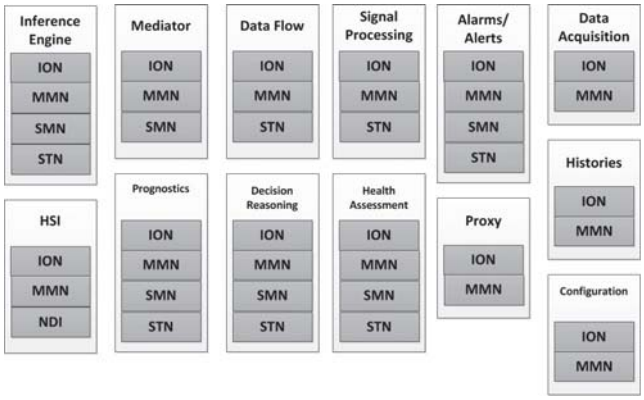


Figure 4 – Minimum Nodes Required for each Service Instantiation

2.3 ISA Personalities

The high-level features of the ISA architecture is enhanced through the use of evolutionary processes embedded within the AIS cognitive architecture and implemented using the distributed capabilities of the agent infrastructure [10, 11]. This allows the ISAs to learn and evolve over time. However, the continuity of operations methods employed for most mission critical systems are not sufficient for a self-evolving AIS. In humans, an injury or disease that causes brain damage can produce serious undesired outcomes [12]. A system failure for an AIS can produce similar outcomes and can result in the loss of learned and/or evolved behavior across a current set of ISAs. To avoid these issues, ISAs store personality tokens known as Cognitrons (COG) at regular intervals that capture the current machine objectives tied continuously to data context arousal and evolutionary states of each ISA in order to envelop the essence of short and long term stored memories [16]. A COG is a collection of state information that describes the high

fidelity and many times multidimensional knowledge and context using Knowledge Relativity Threads [13, 18, 20]. ISA COGs are stored within the AIS memories, allowing personalities to be cloned and/or distributed throughout the system and reconstituted. This allows system regeneration in case of a system failure without loss of evolutionary learning within the AIS. This allows mobility throughout the AIS cognitive framework. This "state mobility" of the ISAs enables them to evolve self-deterministically [13]. However, self-evolution requires policy management for the distribution of ISA updates and COG mobility is handled through Interface ISAs [7]. The COG of a particular ISA is based, in part, on their stochastic need to cooperate, learn, and function autonomously, but within the goals and overall AIS directives. Figure 5 illustrates the COG concept.

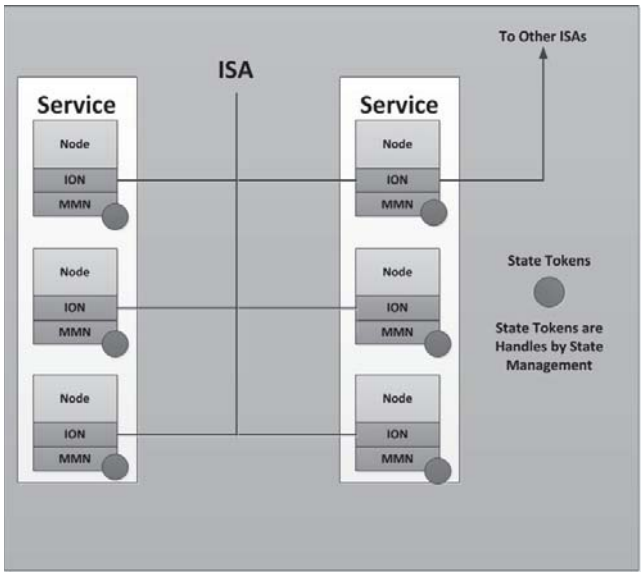


Figure 5 – ISA Personality Tokens

2.3.1 ISA Stochastic Diffusion Modeling

The ability to engage upon any system activity or action, or to engage problems within the AIS we enable a stochastic diffusion process; move resources based upon priority and high fidelity COG KRT combined perceptions of system objectives, perceptions and prioritization of system needs. Figure 6, illustrates as an analogy to Molecular Biology which uses a 2D or 3D computational mesh comprise either of a lattice or spatially aware non-lattice to represent a cellular compartment to represent a physical membrane or cell to simulate activity and interactions. In such simulations, one can characterize and simulate real-time and non-real-time changes within the system and the amount of prioritized ISA resources to use in order to address specific Kinetic system activity [19].

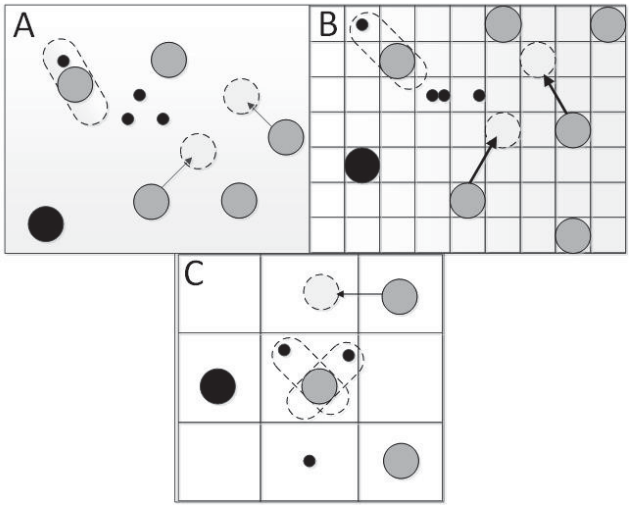


Figure 6 – ISA Stochastic Diffusion (A)Off-Lattice, (B)Micro, (C)Macro Granular Lattice Simulation Method

Off-lattice kinetic ISA simulations have spatial context where Micro Lattice and Macro Lattice allow for different grain size analysis and simulations. Additionally, 3D simulations are possible and can be performed with the use of Voxels which can be tuned to a desired low or high-level of spatial granularity. This allows the ability to spatial calculations for latency evaluations to optimize system performance over time across distributed components or spatially separated nodes. The ISA Stochastic Diffusion method can support distributed simulations from micro-, meso-, macro- or even sizable systems which have a need to model their physically located or moving internal or externally related components and their relationship to required ISA or others system resources. Therefore, this Molecular Biology inspired high definition modeling capability provides usefulness in testing distributed artificially intelligent components and their interactions [18].

2.4 ISA Service Node Strategy

As described above, ISA Nodes provide functional capabilities to ISA services. These strategies provide the algorithms for "how" the ISAs perform the functionalities and capabilities required of a given ISA. There are two main nodal strategies carried within an ISA:

1. **Domain Independent Strategies**: these provide the basic cognitive abilities required for the AIS.
2. **Domain Dependent Strategies**: these are very specific to the domain the AIS is operating within.

Domain independent strategies provide the AIS with cognitive and processing capabilities that are required, regardless of the environment or mission required of the AIS. These strategies include, but are not limited to:

1. Resource Management
2. Rules
3. Inference Implications
4. Learning algorithms
5. Decision Management
6. Memory Creation
7. Memory Integration
8. Memory Construction/Recall
9. Cognitive Needs (basic AIS executives)

Within each of the categories listed above, there may be hundreds of individual strategies that might be used within each ISA service node. An example would be the learning strategy. There are many types of machine learning that may be used, each with advantages and disadvantages, depending on the application.

3. ISA Infrastructure as a Synthetic Nervous System

In order to describe the ISA-based AIS cognitive architecture in familiar terms, we describe the various aspects of an AIS in terms of human brain functionality. Since many people are familiar with the basic human nervous system, it seemed appropriate to describe the ISA DIPS architecture in these terms. Figure 7 illustrates an adaptation of the work done by Kandel [14] and Levine [15], and provides a high-level view of the human Peripheral Nervous System (PNS) in terms of how information is transmitted between bodily components.

The PNS consists of components outside the Brain and Spinal Cord. The effectors consist of the nerves associated with motor nerve fibers, divided between the Somatic Nervous System, the Voluntary Effector Function, and the Autonomic Nervous System (ANS). Information is sent to the brain from the spinal cord and then out to the PNS system. The ANS has two major divisions, the Parasympathetic Division (resource manager and housekeeping functions) and the Sympathetic Division (regulates effector motor nerve fibers and cognitive functions).

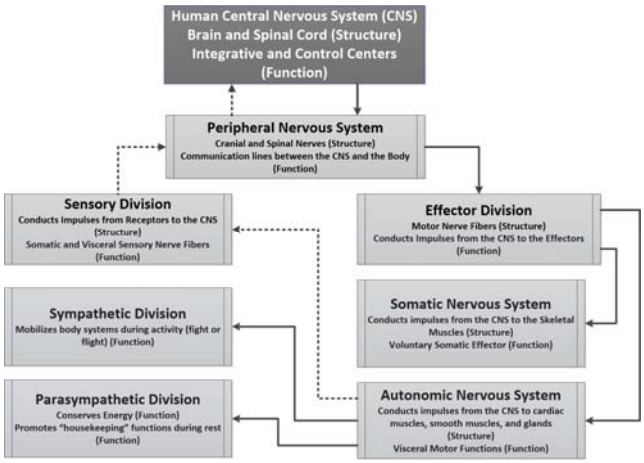


Figure 7 – Human PNS High-Level View

We now adapt Figure 7 to illustrate the cognitive and information flow structures within an AIS, as driven by the ISA infrastructure, to create a Synthetic Nervous System, as illustrated in Figure 8. Here, the ISA-based, distributed processing framework acts as the brain and central nervous system functions within the AIS Synthetic Nervous System (SNS) [5]. As illustrated in Figure 8, effector instructions are transmitted by ISAs through the behavioral processing algorithms to the AIS effectors for both voluntary (overt) execution and involuntary (mandatory) execution. In the case of an AIS, instead of blood vessels, internal organs, and glands, the AIS has an IT infrastructure (processors, memories, networks, etc.). The SNS processes (provided by ISA capabilities) drive the autonomic functionality (mandatory) that regulates the AIS internal systems.

The SNS, like the human Autonomic Nervous System, is divided between the Parasympathetic Division, which in the AIS provides system resource management, and the Sympathetic Division, which drives the cognitive regulation (directives) and effector motor functions (network and activation capabilities) [8].

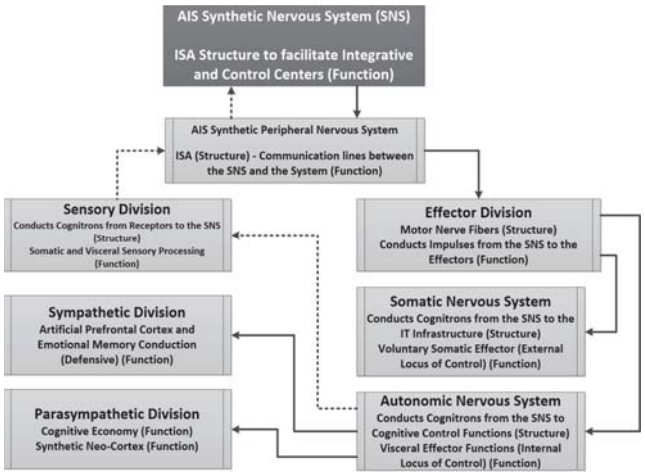


Figure 8 - AIS, ISA-Driven Synthetic Nervous System

4. ISA High-Level Requirements

Based on this view of the AIS cognitive, ISA-driven framework, we discuss this in terms of human brain functionality, providing high-level requirements for overall cognitive functionality required for an AIS DIPS architecture.

4.1 Synthetic Lower Brain Stem Functions

Here, sensory inputs are taken in and early pre-processing of sensory information is performed and sent to the Synthetic Thalamus. The ISA's lower brain functions control the startup of the ISA framework (cognitive awakening) and creates and controls Interface ISAs. Figure 9 illustrates these functions.

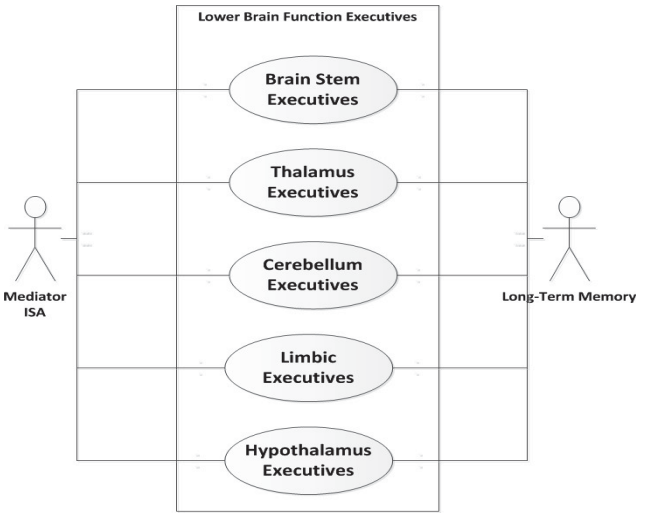


Figure 9 – AIS Lower Brain Function Executives

4.2 Synthetic Thalamus Functions

Here, processes create and control Data Steward ISAs to accept information from the Brain-Stem Data Steward ISAs, where sensory processing algorithms cleanse, categorize, and create metadata and knowledge relativity threads [20]. This defines the initial state of the AIS, creating and disseminating appropriate ISAs, based on the AIS internal state assessment. Advisor ISAs are created to advice on AIS system health status. These executives provide information relay (interface) between the sensory memory and other ISAs within the AIS.

4.3 Synthetic Cerebellum Functions

The Synthetic Cerebellum function within the AIS to create and control interface ISAs for Human Systems Interface (HSI) capabilities. This includes creation of external Advisor ISAs to deliver external motor/actuator control decisions.

4.4 Synthetic Limbic System

These processes within the AIS create and control the Reasoner, Analyst, and Advisor ISAs with specialized strategies for dealing with possibilistic (fuzzy-based) triggers that allow inferences within the AIS cognitive infrastructure. This information is transmitted throughout the AIS via Inference Markup Language (IML) constructs that provide triggers to initiate memory recall/reconstruction.

4.5 Synthetic Hypothalamus System

The Synthetic Hypothalamus serves as the internal Integrated System Health Management (ISHM) for the AIS. This system creates and controls Reasoner, Analyst, and Advisor ISAs with specialized nodes and strategies specific to Integrated and Prognostic Health Management within the AIS. These ISAs monitor and control relationships between the conscious (overt) and subconscious (mandatory) systems within the AIS [17]. These ISAs are sued for self-assessment, self-regulation, and self-healing in the AIS cognitive ecosystem. This includes evaluation of execution of the tasks, goals, initiatives, directives, etc. that are given/provided to the AIS.

5. Conclusion and Discussion

Here we have provided and discussed software structures and stochastic diffusion modeling techniques [19] that facilitate DIPS and provides for

potential artificial cognitive processes within a distributed AIS. This ISA architecture provides structures and processes for creating, storing, and constructing memories, learning, reasoning, and inferencing capabilities required for a functional AIS and testing techniques for distributed AIS [3]. Humans are made up of thousands of biological processes that create our overall structure. However, an AIS core is made up of, hardware and software. Future work will lay out the foundational hardware designs and software frameworks required to implement and actualize this potential ISA infrastructure and the overall functional AIS.

References

1. Bond, A. and Gasser, L. (1988). Readings in Distributed Artificial Intelligence. Morgan Kaufman, San Mateo, CA.
2. Chaig-Draa, B., Moulin, B., Mandiau, R., and Millot, P. 1992. Trends in distributed artificial intelligence. Artificial Intelligence Review, 6(1):35-66.
3. Crowder, J. 2013. The Advanced Learning, Abductive Network (ALAN). Proceedings of the AIAA Space 2013 Conference, San Diego, CA.
4. Crowder, J., Carbone, J., and Friess, S. 2013. Artificial Cognition Architectures. Springer Publishing, New York, NY. ISBN 978-1-4614-8071-6.
5. Jennings, N. 1996. Coordination Techniques for Distributed Artificial Intelligence. Foundations of Distributed Artificial Intelligence, pp 187-210.
6. Yoay, S., Leyton-Brown, K. 2009. Multiagent System: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, New York, NY., ISBN 978-0-521-89943-7.
7. Vlassis, N. 2008. A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence. Morgan & Claypool Publishers, San Rafael, CA., ISBN 978-1-59829-526-9
8. Hopfield, J. 1982. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proceedings of the National Academy of Science, Vol. 79.
9. Jacobs, R., Jordan, M., Nowlan, S, and Hinton, G. 1991. Adaptive Mixtures of Local Experts. Neural Computation, Vol. 3.
10. Crowder, J. and Friess, S. 2011. The Artificial Prefrontal Cortex: Artificial Consciousness. Proceedings of the 13th annual International Conference on Artificial Intelligence, Las Vegas, NV.
11. Crowder, J., Scally, L., and Bonato, M. 2012. Applications for Intelligent Information Agents (I2As): Learning Agents for Autonomous Space Asset Management (LAASAM). Proceedings of the International Conference on Artificial Intelligence, ICAI'12, Las Vegas, NV.
12. Dudai, Y. 2004. The Neurobiology of Consolidations, or, How Stable is the Engram? Annu Rev Psychol 55:51-86.
13. Crowder, J. and Carbone, J. 2011. The Great Migration: Information to Knowledge using Cognition-Based Frameworks. Springer Science, New York, NY.
14. Kandel, E., Schwartz, J, and Jessel, T. The anatomical Organization of the Central Nervous System. Principles of Neural Science (Ch. 17), McGraw Hill Professional, ISBN 978-0-8383-7701-1.
15. Levine, P. 1997. Walking the Tiger: Healing Trauma. North Atlantic Books, Berkeley, CA.
16. Coutaz, J., Crowley, J., Dobson, S., and Garlan, D. 2005. Context is key. Communications of the ACM, vol. 48, pp. 53.
17. Crowder, J. 2006. Intelligent Information Agents for Autonomous Condition Based System Health Management. Proceedings of the AFRL Integrated Systems Health Management (ISHM) Conference.
18. Carbone, J. "A Framework for Enhancing Transdisciplinary Research Knowledge." Texas Tech University (2010).
19. Burrage, Kevin, et al. "Stochastic simulation for spatial modelling of dynamic processes in a living cell." Design and Analysis of Biomolecular Circuits. Springer New York, 2011. 43-62.
20. Crowder, J., and John Carbone. "Recombinant knowledge relativity threads for contextual knowledge storage." Proceedings of the 13th annual International Conference on Artificial Intelligence, Las Vegas, NV. 2011.

Comparing the Group Intelligence of Static Agents versus Mobile Agents in a Mobile Autonomous Environment

J. McKinney Young[1], L. Etzkorn[2]
[1,2]Department of Computer Science, University of Alabama in Huntsville, Huntsville, Alabama, USA
300 Technology Hall
University of Alabama at Huntsville
Huntsville, AL 35899

Abstract

In 2010, Keen and Etzkorn defined and measured the group intelligence of mobile agents operating on a mobile system versus static agents operating on a mobile system at run-time in a simple search scenario. We recreated the experiment in a simulation using a cellular automaton. Then we validated that the simulation modeled the static and mobile agents behavior on the mobile environments for the same search-and-find task. This papers describes the validation scenarios and results. We also discuss the metric used and the definition of intelligence as it relates to action and to comparing different approaches to discrete tasks.

Keywords: Artificial Intelligence, Mobile Agents, Cellular Automata

1 Introduction

In 2010, Keen and Etzkorn defined intelligence as the inverse of complexity given a specific task [6]. He created a real world implementation that compared the group intelligence of actual static agents running on autonomous mobile systems and the group intelligence of mobile agents running on autonomous mobile systems. The case study found that three mobile agents directing three autonomous mobile systems demonstrated higher intelligence than the static agent group. However, the original agents/mobile system teams performed their seek&find task under tightly scoped and controlled conditions. Further analysis in a broader environment is warranted. Given more difficult tasks, such as tasks with competing constraints, do the findings remain constant? Does the original definition of intelligence, and the metric used to capture the expression of intelligence in action, remain true as the problem areas grow more difficult? To begin our investigation of these questions, we moved the experiment to a simulation where we can expand the study to problem domains of greater complexity and difficulty.

2 Related Work in Defining Intelligence

What constitutes intelligence is an on-going discussion. Intelligence in any form, whether biological or artificial, can be difficult to quantify. The comment, "I know it when I see it" [12], while intuitively appealing, is insufficient for scientific discussion.

Allen Newell, in his book, Unified Theories of Cognition, writes that intelligence is the ability to perfectly use all of one's knowledge in support of achieving one's goals [10]. He defines system intelligence as "the degree to which it approximates a knowledge system..." [at the point in time when one actually uses one's knowledge]. Pure knowledge-level creatures (like a thermostat) cannot be considered "intelligent" - they do what they do and are not able to do any more or any better. But real creatures, Newell maintains, face real world difficulties in bringing all of their knowledge to bear "and intelligence describes how well they do that" [10]. This view of knowledge, intelligence and the creature who embodies them is supported by the mathematician, physicist and philosopher, Charles Sanders Pierce, who lived nearly one hundred years before Newell. Pierce believed that the real creature is not outside of the world observing it but is an active participant in the world whose knowledge and understanding of it "have above all to meet urgent needs" [11]. Pierce's central thought was that intelligence is an activity. Creatures are moved to enquire by some need (some objective). This leads one to evaluate one's problem-situation in search of what is not right and how it may be put right [11]. This goal-directed evaluation using what one has in the way of knowledge is intelligence. Both Newell and Pierce saw intelligence as action. Only in the actual striving towards some objective is intelligence exercised or demonstrated. Lewis Johnson ups the ante in the discussion of artificial intelligence by asking not only can we define it, but can we measure it? "Further, it would be helpful to have a relative measure of intelligence that would enable us to judge one program more or less intelligent than another." [5]

Pierce's and Newell's definition requires expansion when one applies the definition to a machine. Alexander Meystel and Elena Messina give an example of six degrees of intelligence that differentiates an air conditioning unit from an "artificial climate control system." Intelligence level 1 is a simple air conditioning unit that can turn on the air when it's too warm and turn it off when the temperature has reached a certain setting. The system becomes ever more intelligent until it reaches level six where it can balance multiple user's comfort, the cost impact of temperature settings and its own life span with regards to reliability and machine aging. [9]

While Newell discusses intelligence as a function of bringing perhaps imperfect knowledge to support achieving an objective, Stevan Harnad suggests that the way a program expresses intelligence is by interacting with an "uncertain" environment [5]. The uncertain environment Harnad references is, in fact, the physical world. Harnad suggests that the program should run on a robot who receives input through multiple sensors. The sensor data would provide "unexpected" data such as light and shadow as well as walls and carpeting. This would serve to literally, as well as figuratively, "trip up" the program as it attempts to achieve its goal. Again, the idea of how well the program responds to imperfect situations is an expression of the "intelligence" of the program. This idea is in keeping with Ahmad, Alvarez and Wah's conclusion that an intelligent object is capable of autonomous operations executed in an unpredictable manner [1]. The authors equate intelligent instructions to be instantiated as conditional branches and that unpredictable external events cause the program to exercise the conditional branch evaluation. The evaluation itself is the exercise of the program's intelligence. These philosophical discussions of intelligence form our understanding of the concept. However, they don't explicitly provide a method to quantify and measure intelligence.

Kevin Keen presents an attractively measurable definition of intelligence. In a study of autonomous static and mobile agents running on autonomous mobile systems, Keen writes,

> intelligence is the inverse of complexity. Any measure of intelligence must be made with respect to the goal of the program. [6]

Keen's concept of intelligence is a "task-based" or "directed" intelligence. It is emphatically not the same as general purpose human intelligence. [6]

It is important to note that Keen's research focused exclusively on code complexity at run-time. Keen's study attempted to measure the complexity of the code that actually executed during the scenario test. Keen, like Pierce, Newell, Harnod, Ahmad, Alvarez, and Wah, believed firmly that it is in the doing that real intelligence is exercised.

3 Background

Recent work by Kevin Keen compared the behavior of two programming approaches by implementing the approaches as agents - static-vs-mobile - running on simple mobile systems fitted with infrared sensors. [6] The mobile systems had a task to perform - to move about a grid in search of three colored balls in a particular order. The "agents" directed the mobile systems and processed the "visual" (infrared) data that the mobile systems perceived.

Keen created the metric $Keenint_{RM}$ to measure the activity of the running code on the mobile system in action. Keen defined $Keenint_{RM}$ = 1/complexity for some particular task. [6] The metric is very similar to a runtime version of McCabe's cyclomatic complexity [6]. It is the sum of decision points encountered by the running code plus one is added at every method invocation even if no decision structures are encountered. This represents the idea that some level of intelligence is represented by the activity of processing the method even without the greater work of processing a control structure. In the Three-Mobile System Scenario, which compared the activity of three mobile systems with static agents and then three mobile systems with mobile agents, starting at the same start points searching for the same balls in the same location, the data suggested that the mobile agent approach showed greater intelligence than the static agent approach. [6] Keen's work also demonstrated that the mobile agent approach showed greater resiliency in the face of obstacles than the static agent approach. When one mobile system got stuck in a corner and couldn't turn enough to remove the wall/obstacle from its view, the mobile agent resident on the "stuck" mobile system was able to jump to another mobile system (that had completed its agent's task) and continue searching for it's colored ball. In the static agent approach, both the "stuck" mobile system and its resident static agent where unable to complete their task.

This metric identifies when the program is using its knowledge - that point in time when its intelligence is applied. By examining the running code using the metric, $Keenint_{RM}$, and by observing the behavior of the mobile systems and the agents executing on them, Keen was able to measure the "intelligence in action" of the two approaches to the same task.

4 Case Study

We recreated the actual static agents running on autonomous mobile systems and mobile agents running on autonomous mobile systems case study in a simulation. The simulation allows us to study the run-time intelligence of the agents code as they seek to solve more complex problems. The first step of this endeavor, creating the simulation and validating its behavior, is described in this paper.

For this inquiry, we use Keen's definition of intelligence. For a program (or a machine) to be considered intelligent, it must be independent, goal-oriented and resilient. And, not only must a system meet the above requirements, but, as in Keen's work, HOW it accomplishes its' objective is important. The program that is judged to be more intelligent must choose the most efficient, or the simplest, approach.

The actual measurements of the $Keenint_{RM}$ metric in the simulation is not identical to the metrics recorded in Keen's case study due to the differences in design and implementation. What is critical is that the metrics gathered by the static agents versus the mobile agents - relative to each other - are comparable to what Keen found in his study.

The simulation is implemented in the form of a cellular automaton (CA) using Microsoft Access. The researcher may observe the progress of the running code (written in Visual Basic) through the graphical user interface (GUI). The persistent datafiles in which the rules, the geography, and the intermediate and final results are stored are used to derive the results of each run.

The formal definition of a CA is expressed by four things: the array dimensionality d, the set of states (the number of cells) S, the neighborhood vector (the definition of what is a neighborhood) N, and the local rule (how local neighbor states are evaluated) f. [2]
CAi = (d, S, N, f).

At time quanta zero, an initial start state is assigned to each cell. The collection of all cells' states is called the configuration of the cellular automaton at that time. The passing of each time quanta causes the states of all the cells to change (according to the cellular automaton's rule) and a new configuration describes the cellular automaton at that instant in time.

The CA expresses a 2-dimensional array(d) representing geographic direction (*x*,*y*) for the case study. The simulation models mobile systems that are capable of moving only "wheels on the ground." There is no vertical component (*z*) to their movement. The cells (S) of the automaton comprise a 7x7 grid which has a total of 49 possible location states. The neighborhood vector (N) is made up of 4 possible cells at the North, East, South, West sides of the cell under analysis. The array of local rules are as follows:

Rules of CA behavior

1. There is a central clock.
2. Time only flows in one direction - forward. (There is no "redo" capability.)

Rules of Mobile System Behavior

1. Mobile Systems can only do one activity at each time tick:
    a. Start
    b. GoToSleep
    c. Die
    d. FindObject
    e. Walk
    f. Turn
    g. Lose Agent
    h. Receive Agent
2. Mobile Systems can only turn 90 degrees in one time tick.
3. A mobile system cannot revisit a cell that it has already visited.
4. Only one mobile system at a time may occupy a cell.
5. A mobile system will "go to sleep" on the 5th time tick after it has turned a complete circle (4 turns in 4 consecutive time ticks) and has not been able to move out of its current cell.
6. Each Mobile System starts the simulation with one assigned Agent.
7. Mobile Systems possess the attribute of "handedness" (left or right) - when a mobile system turns, it will turn in the direction as specified by its "handedness" setting.
8. Mobile Systems "find" a ball by stepping into a cell that contains a ball.
9. Mobile Systems walk a self-selected path using Dijkstra's algorithm.

Rules of Agent Behavior

1. Agents can only do one activity at each time tick:
    a. DoNothing
    b. GoToSleep
    c. Die
    d. FindObject
    e. Jump
2. When a mobile system dies so also does its agent.
3. An agent is assigned a particular colored ball (red, blue, green) to find.
4. It can only "find" that one ball.
5. The agent "checks" if the ball is in the cell during the same time tick that the mobile system has moved into a new cell.
6. After the agent finds its ball, it goes to sleep.
7. A different agent who is still active (has not found it's ball) can jump to a mobile system with a "sleeping" agent.
8. A different agent who is still active (has not found it's ball) can jump to a mobile system without an agent.
9. A "jumping" agent may only leave one mobile system and arrive at another mobile system in one time tick. (The mobile system will "wake up" in the next time tick.)

4.1 Scenario Descriptions

In the three mobile system/static agent teams, three mobile system/mobile agent teams scenarios, the start states are described below.

Scenario 1

Scenario length: 60 time quanta & 120 time quanta
B1: red, at 2,2
B2: green, at 4,5
B3: blue, 1,6
The mobile systems start state is:
R1, agent1 - read ball, 7,1, left-handed
R2, agent2 - green ball, 7,2, right-handed
R3, agent3 - blue ball, 1,7, right-handed

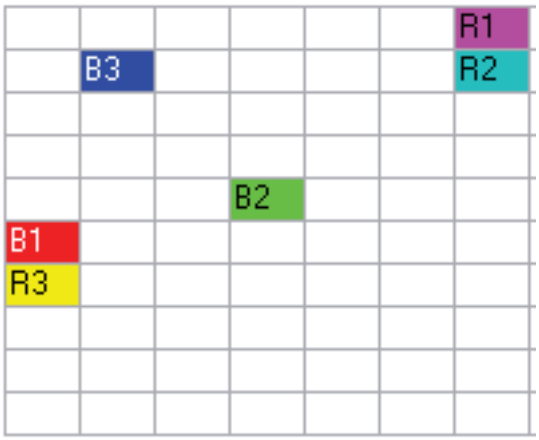


Figure 1. Scenario 1 Start State

In both static and mobile agent setup, R1 will move to the left and "find" it's ball in 33 time ticks. R2 will move down the grid and get "stuck" in the lower-right (SE) corner. At time tick 10, R2 "goes to sleep" after having been blocked at moving out of the cell in the four previous time ticks by the boundaries of the graph, the fact that the cell immediately "above" him is already visited, and that the cell immediately to the left is occupied by R3. In the static agent scenario, agent 2 also "goes to sleep." In the mobile agent scenario, agent 2 enters the "ReadyToJump" state. R3 gets "stuck" at time tick 23 just above R2. Like agent 2 earlier, in the static agent scenario, agent 3 "goes to sleep". In the mobile agent scenario, agent 3 enters "ReadyToJump" state. At time tick 34, R1 is available for a new agent. Agent 2 "jumps" to R1 and begins searching for its green ball. Agent 3 "jumps" to R2 as soon as the platform is available at 35 seconds and begins searching for ball 3. Agent 3 on R2 finds its ball at 45 time ticks. Agent 2 on R1 finds B2 at time tick 56.

Scenario 2

Scenario length: 60 time quanta & 120 time quanta
B1: red, at 2,4
B2: green, at 5,3
B3: blue, 6,6
The mobile systems start state is:
R1, agent1 - read ball, 7,1, left-handed
R2, agent2 - green ball, 7,2, right-handed
R3, agent3 - blue ball, 1,7, right-handed

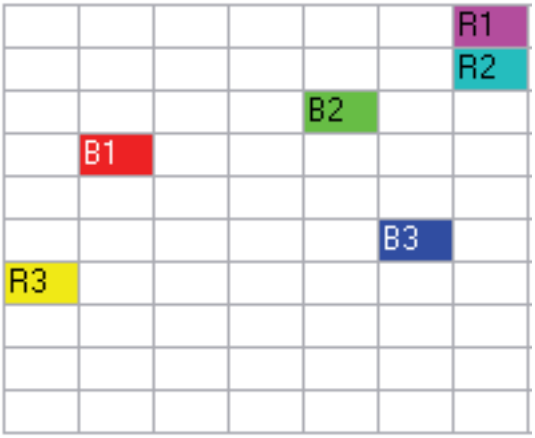


Figure 2. Scenario 2 Start State

In both static and mobile agent setup, agent 3 on R3 will move to the right and "find" it's ball in 9 time ticks. Also in both scenarios, Agent 1 on R1 walks down the graph. It navigates around R2 in the lower-right (SE) corner & continues to circle the graph. The Agent 1/R1 pair successfully locate ball 1 at time tick 36. R2 will move down the grid and get "stuck" in the lower-right (SE) corner. At time tick 10, R2 "goes to sleep" after having been blocked at moving out of the cell in the four previous time ticks by the boundaries of the graph, the fact that the cell immediately "above" him is already visited, and that the cell immediately to the left is occupied by R3. In the static agent scenario, agent 2 also "goes to sleep." In the mobile agent scenario, agent 2 enters the "ReadyToJump" state and "jumps" to R3 at time tick 11. Agent 2 moves R3 down the right side of the graph to the lower right (NE) corner where R3 gets blocked by R2 below it.. At time tick 31, Agent 2 "jumps" back to R2 (who is now no longer blocked by R3 to the left). Agent 2 & R2 walk the graph searching for ball 2 until R2 gets stuck at time tick 58 in the upper right (NE) corner of the graph (the platform cannot rewalk nodes that it has already travelled.). Agent 2 "jumps" to R1 at time tick 59 and finds the green ball at time tick 70.

(This space intentionally left blank to keep the following scenario description and related graphic together.)

Scenario 3
Scenario length: 60 time quanta & 120 time quanta
B1: red, at 1,4
B2: green, at 1,6
B3: blue, 7,3
The mobile systems start state is:
R1, agent1 - read ball, 7,1, left-handed
R2, agent2 - green ball, 7,2, right-handed
R3, agent3 - blue ball, 1,7, right-handed

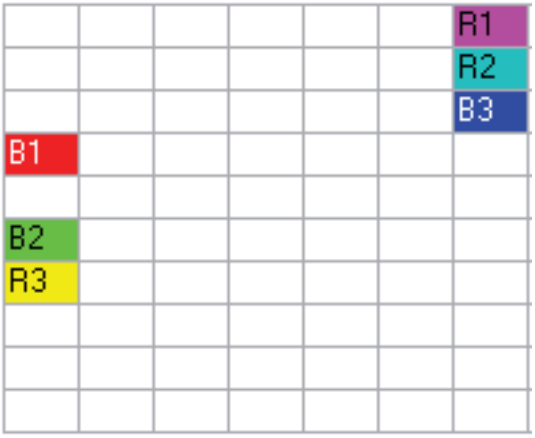


Figure 3. Scenario 3 Start State

In both static and mobile agent setup, R1 will move to the left and "find" it's ball in 13 time ticks. In both scenarios, R2 will move down the grid and get "stuck" in the lower-right corner. Agent 3 on R3, in both static and mobile scenarios, will find its ball in 19 time ticks. At time tick 10, R2 "goes to sleep" after having been blocked at moving out of the cell in the four previous time ticks by the boundaries of the graph, the fact that the cell immediately "above" him is already visited, and that the cell immediately to the left is occupied by R3. In the static agent scenario, agent 2 also "goes to sleep." In the mobile agent scenario, agent 2 enters the "ReadyToJump" state. At time 14, agent 2 jumps to R3. Agent 2 on R3 finds it's ball at 59 time ticks.

Scenario	Balls Found (Static) 60 sec	Balls Found (Mobile) 60 sec	Static Agents Decision Count 60 sec	Mobile Agents Decision Count 60 sec
1	1	3	187,393	179,193
2	1	2	169,470	199,354
3	2	3	164774	23,271

Table 1: Figure Keenint$_{RM}$ Aggregate Data (60 sec run)

Scenario	Balls Found (Static) 120 sec	Balls Found (Mobile) 120 sec	Static Agents Decision Count 120 sec	Mobile Agents Decision Count 120 sec
1	1	3	622,363	179,193
2	1	3	576,360	266,486
3	2	3	563,204	23,271

Table 2: Figure Keenint$_{RM}$ Aggregate Data (120 sec run)

5 Conclusion and Future Work

The scenarios shows that the simulation mimics Keen's findings in his study with the actual mobile systems as far as agent behavior and mobile system behavior is concerned. For example, the simulated mobile systems experienced the same difficulty that the actual mobile systems did when wandering too close to the boundaries of the grid. Both actual and simulated mobile systems were much more likely to get "stuck" in corners. The actual mobile systems experienced "deadlock" [6] when the mobile system approached a wall at such an angle that no matter which way it rotated, it would perceive a barrier and become stuck. In the simulated scenario, there is a high probability that the mobile system will reach "deadlock" at a corner because of the rules that the mobile system cannot backtrack over a previously visited cell AND that the mobile system must enter the "sleep" state if it rotates four consecutive time without being able to exit the cell (say, if another mobile system occupied the "next" cell checked after the last rotation.) In the simulation, both the "GoToSleep" timer and the "BackTracking" prohibition should be evaluated to see if it would be more realistic to relax the constraints under certain "high risk" conditions. An example of a "high risk" condition is given by the start states of R1 and R2 (location on the graph) and the starting orientation (what direction are they facing) in the scenarios shown above. Given their start states and each platforms rules of "handedness" (in what direction the platform will turn when blocked), it is guaranteed that the two platforms will reach the lower right (SE) corner one behind other. These start states created a condition of "blocking" early in the scenario. This was a conscious design of each scenario - to allow the platform agent pairs "room" to demonstrate resilience.

The mobile system/agent teams demonstrated another similarity to Keen's case study with regards to "view" impediment. In the actual case study, when one mobile

system was directly in front of another mobile system's IR sensor, the mobile system in front blocked the other mobile system's view. If there was a ball that the second mobile system was supposed to have detected, the detection action did not succeed. In the simulated experiment, the rule that stated that one mobile system could not enter a cell that was already occupied plus the rule that the agent could only check for a colored ball once the mobile system had entered a new cell effectively recreated the IR blocking behavior.

Another similarity between the actual case study and the simulation is that the mobile agents as a group exhibited greater effectiveness in achieving their objectives in the face of obstacles. In all three scenarios for both scenario lengths, the mobile agent/mobile platform teams outperformed the static agent/mobile platform teams in terms of finding the colored balls.

An important similarity between the actual case study and the simulation are the $Keenint_{RM}$ values recorded for the two types of agents versus each other. Keen found, in his study, that the mobile agent $Keenint_{RM}$ values were lower than those of the static agents in two out of three scenarios. (The third scenario yielded inconclusive data.) [6] The simulation derived metrics that show the same - slightly lower $Keenint_{RM}$ values for all scenarios for the mobile agents relative to the static agents in the 60 second tests. The differences in the $Keenint_{RM}$ values are more obvious in the longer 120 second tests where one sees much larger values for the static agent versus the mobile agents.

The more dramatic differences between the static agents and the mobile agents is due to the differing lengths of the tests and to the prescription to recreate Keen's experiments in the simulation. In Keen's experiment, the mobile agents that achieved their objective "died." The static agents that achieved their objective stayed "alive" and kept reporting status which drove the decision counter metric higher. It is important to note, however, that only in Scenario 3, did all three static agents successfully accomplish their objective. (And, therefore, logically should have quit reporting.) In the other two scenarios, the static agent/mobile platform teams did NOT achieve their objective. So, the the higher decision count due to continuing striving and continual reporting activity until the end of the scenario was a logically correct activity.

Now that we have confidence that the simulation does represent the causal properties [1] of the actions of the real mobile system/agent teams, we will begin our exploratory work studying intelligence and run-time complexity with more complex mobile system task such as path planning in unknown grids. Additionally, we will study how the number of mobile system/agent teams affects the efficiency of the group behavior. Would the teams achieve economies of scale or would the larger number cause more interference for each individual team?

References

[1] Ahmad, O., Alvarez, A.M. and Wah, C., 1992. "Quantitative Measure of Intelligence." In *IEEE ICCS/ISITA* (Singapore, 1992), IEEE, p. 1347-1351.

[2] Brooks, Richard R., Orr, Nathan, 2002. "A Model for Mobile Code Using Interacting Automata," *IEEE Transactions on Mobile Computing*, Vol. 1, No. 4, October-December 2002. pp. 313-325.

[3] Harnad, Stevan, 1989. "Minds, Machines and Searle," *Journal of Theoretical and Experimental Artificial Intelligence*, vol. 1, 5-25.

[4] Harnod, Stevan, 1992. "The Turing Test is Not A Trick: Turing Indistinquishability is a Scientific Criterion," *ACM SIGART* Bulletin 3 (4), 9-10.

[5] Johnson, W.L., 1992. "Needed: A New Test of Intelligence," *ACM SIGART* Bulletin 3 (4), p. 7-9.

[6] Keen, Kevin, 2010. Measuring and Comparing Group Intelligence of Mobile and Intelligent Agents on a Mobile Robotics Platform.

[7] Keen, K., Mathur, R., Etzkorn, L., 2010, Towards A Measure of Software Intelligence Employing A Runtime Complexity Metric, In *Proceedings of the Software Engineering and Applications (SEA) Conference, Cambridge, MA.*

[8] Mathur, R., Keen, K., Etzkorn, L.H., 2010. Towards an Object-Oriented Complexity Metric at the Runtime Boundary Based on Decision Points in Code, In *Proceedings of the ACM Southeast Conference, Oxford, MS.*

[9] Meystel, A. and Messina, E., 2000. "The Challenge of Intelligent Systems." In *Proceedings of the 15th IEEE International Symposium on Intelligent Control*, Rio, Patras, Greece, IEEE, p. 211-216.

[10] Newell, Allen, 1990. Unified Theories of Cognition, Harvard University Press.

[11] Pierce, Charles Sanders, 1931-1938. "Collected Papers."

[12] Stewart, P., 1964, *Jacobellis v. Ohio*.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

COGNITIVE COMPUTING, LEARNING METHODOLOGIES AND APPLICATIONS: REINFORCEMENT, CAUSAL, SUPERVISED AND UNSUPERVISED LEARNING + DEEP LEARNING

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Theory-based Learning Analytics: Using Formal Concept Analysis for Intelligent Student Modelling

Michael. D. Kickmeier-Rust[1], **Michael Bedek**[1], **Dietrich Albert**[1]
[1]Knowledge Technologies Institute, Graz University of Technology, Graz, Austria

Abstract - Learning Analytics is one of the most important fields for driving (educational) artificial intelligence. In this paper we briefly introduce one of Europe's key research initiatives to make a step from primarily statistics and data mining driven ways to do Learning Analytics, towards more theory-grounded, psycho-pedagogically inspired analyses. One approach we suggest is Formal Concept Analysis, which is a strong theoretical backbone in technology-enhanced learning settings For educators, FCA may provide AI-based analyses of 'big data' sets and intuitive representations for a number of pedagogical questions concerning the performance of students on the individual- as well as the class-level.

Keywords: Formal Concept Analysis, Learning Analytics, Visualizations, Learner Modelling

1 Introduction

Learning Analytics (LA) can be considered a best practice in education and it is a key factor for making education more personalized, adaptive, and effective. This, undoubtedly, needs a strong educational AI. Analyzing a variety of available data to uncover learning processes, strengths and weaknesses, competence gaps clearly is a prerequisite for any formative guidance, meaning to assess students in order to provide them with the best possible and most individual support and not just to "rate" them, for changing and adjusting educational measures and teaching, and not least for disclosing and negotiating learner models [1, 2]. On the basis of available data, ideally large scale data sets, smart tools and systems are being developed to provide teachers with effective, intuitive, and easy to understand aggregations of data and the related visualizations. There is a substantial amount of work going on this particular field; visualization techniques and dashboards are broadly available (cf. [3, 4]), ranging from simple meter/gauge-based techniques (e.g., in form of traffic lights, smiley, or bar charts) to more sophisticated activity and network illustrations (e.g., radar charts or hyperbolic network trees). However, LA operates in a delicate and complex area. On the one hand, facing today's classroom realities, we often find environments without a comprehensive availability of technical infrastructure. This, in turn, does not allow or support an easy recording the necessary data, LA and educational data mining (EDM) is always talking about. Also, from a socio-pedagogical perspective, learning must be seen as a process of social interaction that not always occurs in front of some electronic device. Thus, LA must be based on fewer data. Perhaps incomplete data, and accordingly careful must be our approaches to data analyses. On the other hand, it is so very easy to visualize learning on a superficial level using perhaps the aforementioned traffic lights or bar charts. The added value to the teachers is likely of limited utility to them. To provide a deeper and more formatively-inspired insight into the learning history and the current state of a learner and to entire groups of learners (beyond the degree to which a teacher might know it intuitively) requires finding and presenting complex data aggregations. This, most often, bears the significant downside that it is hard to understand.

Challenges for LA and its visualizations, for example, are to illustrate learning progress (including learning paths) and - beyond the retrospective view - to display the next meaningful learning steps and topics.

In this paper we introduce the method of directed graphs, the so-called lattice diagrams, for structuring learning domains and for visualizing the progress of a learner through this domain. This type of visual analytics possibly provides a solid AI foundation for the field of education and LA: In a recent project named Lea's Box (www.leasbox.eu) we propose so-called Formal Concept Analysis (FCA) as a framework for addressing these ideas. We developed a web-based tool which provides a working environment with social and personal open tools to support students in developing their inquiry based learning skills. By this, the tool supports learners by enabling domain and open learner modelling. The fields of application of the FCA in general, and the FCA-tool in particular, have been extended in the course of the LEA´s BOX project (http://leas-box.eu/) which stands for Learning Analytics Toolbox. In the context of LEA´s BOX, the FCA-tool is mainly used by teachers for student modelling and visualization of educational data. By applying formal concept analysis on students´ performance data, a set of pedagogically relevant questions for teachers can be addressed and visualized.

2 Formal Concept Analysis (FCA)

FCA describes concepts and concept hierarchies in mathematical terms, based on the application of order and lattice theory [5]. Insofar, FCA is at the borders between AI research and educational technologies. The starting point is the definition of the formal context K which can be described as a triple (G, M, I) consisting of a set of objects G, a set of attributes M and a binary relation I between the objects and the attributes (i.e. "g I m" means "the object g has attribute m"). A formal context can be represented as a cross table, with objects in the rows, attributes in the columns and assigned relations as selected cells. An example of a formal context is shown in Fig. 1. This formal context has been created by the FCA-tools *Editor View*. Teachers use the *Editor View* to define the formal context and to add learning resources (URLs or files) which can be assigned to both, to objects and to attributes, respectively. In order to create a concept lattice, for each subset $A \in G$ and $B \in M$, the following derivation operators need to be defined:

$A \mapsto A' := \{m \in M \mid g\ I\ m \text{ for all } g \in A\}$, which is the set of common attributes of the objects in A, and

$B \mapsto B' := \{g \in G \mid g\ I\ m \text{ for all } m \in B\}$, which is the set of objects which have all attributes of B in common.

A formal concept is a pair (A, B) which fulfils $A' = B$ and $B' = A$. The set of objects A is called the extension of the formal concept; it is the set of objects that encompass the formal concept. The set B is called the concept's intension, i.e. the set of attributes, which apply to all objects of the extension. The ordered set of all formal concepts is called the concept lattice $\mathfrak{B}(K)$ (see [6] for details), which can be represented as a labelled line diagram (see Fig. 2). The concept lattice shown in Fig. 2 has been created by the FCA-tool *Lattice View*. Every node represents a for-mal concept. The extension A of a particular formal concept is constituted by the objects that can be reached by descending paths from that node. As an example, the node with the label "Goldfish" has the extension {Goldfish, Tree frog}. The intension B is represented by all attributes that can be reached by an ascending path from that node. In the example above, the intension consists of {is able to swim, lives in / on the water}.

Fig. 1. *Editor View* for creating objects, attributes, and relations.

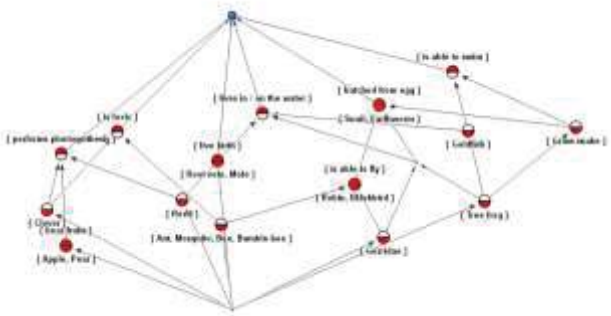


Fig. 2. Concept lattice.

3 Domains Open Learner Models

Once the teacher has created the formal context, students can explore the resulting concept lattice by engaging in interactive graph visualizations (see Fig. 2). By selecting a node, the corresponding concept´s extension and intension are illustrated in a highlighted manner. The concept lattice makes the structure of the knowledge do-main and the interrelations of its concepts explicit. Similar as for concept maps, this kind of graphic organizer aims to facilitate meaningful learning by activating prior knowledge and illustrating its relationship with new concepts [7].

In case the teacher also assigned learning resources to the objects and attributes in the FCA-tool's *Editor View* open learner modelling can be supported. Visualizations of open learner models (for an overview see [8]) are aiming to facilitate reflection on the side of the students and to support teachers to better understand strengths and weaknesses of their students. The FCA-tool's *Lattice View* (cf. Fig. 3) applies the often-used traffic-light analogy (see e.g. [9]) to show the student the extent to which he or she already consumed learning resources.

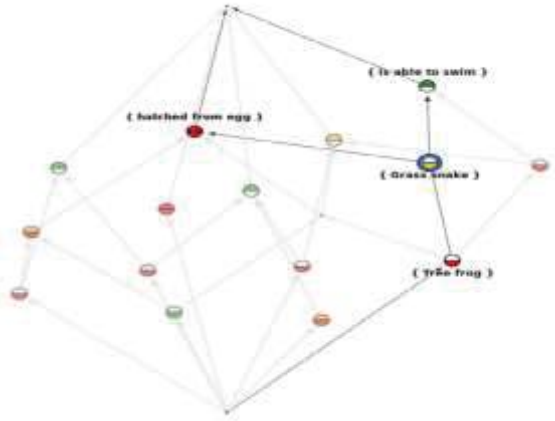


Fig. 3. FCA-tools *Lattice View* for visualizing domain- and learner models.

4 Application in School Settings

Similar as [0] who were the first who applied the FCA with students and their performance data we suggest for-mal contexts with student as "attributes" and problems or test-items as "objects". The relation between these two sets means "student m has solved test item g". An example of a concept lattice which results from such a formal context is shown in Fig. 4 (the data has been reported by [10]). As briefly outlined in the following sections, such a concept lattice visualizes answers to a set of pedagogical questions which are of high interest for teachers.

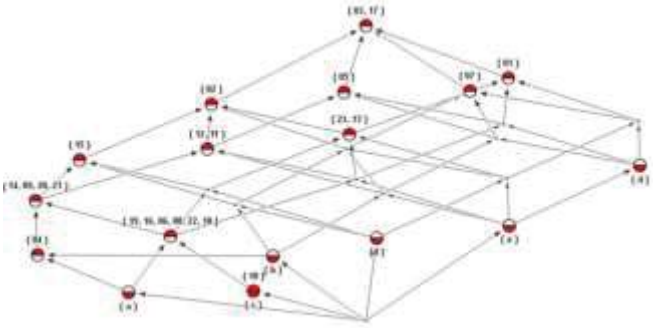


Fig. 4. Concept lattice with students as attributes (numbers from *01* to *23*) and test items (letters *a*, *b*, *c*, *d*, *e*, and *f*) as objects.

4.1 Extensions and Intensions

As mentioned above, the set of test items which have been solved by a particular student can be directly depicted from the extension of the formal concept with the students′ label assigned to it. As an example in Fig. 4, student *10* is the only one who solved only a single test item, *c*, and students *03* and *17* (assigned to the top element of the concept lattice) mastered all problems. When clicking on a particular node the formal concept′s extension and intension is highlighted. As an example shown in Fig. 5 (left side), the student *04* has successfully mastered the test items *a* and *b*.

The intension of a formal concept which has an object-label assigned to it indicates the set of students which have successfully mastered the according test item. As an example, the problem d in Fig. 5 (right side) has been solved by the students 01, 03, 05, 07 and 17. As it can be also seen, this formal concept located above the formal concept with the object-label e assigned to it. This means, that all students who solved item d were also able to solve item e.

4.2 Overlapping and Differences

The performance of two or more students can be compared when examining the intensions of the formal concepts with the according attribute-labels. As an example, the students *07* and *15* mastered different subsets of problems (see Fig. 4): Student

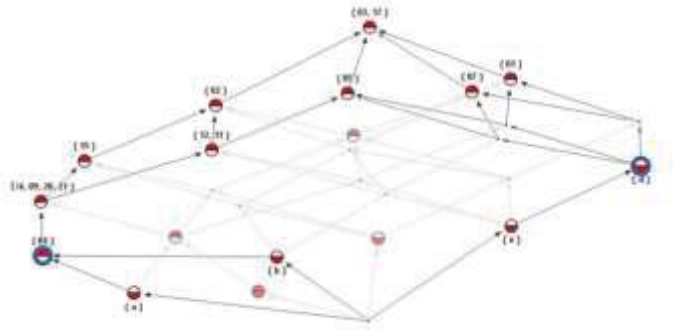


Fig. 5. The extension represents the set of test items solved by a student (see student *04*) and the intension represents the set of students who solved the particular test item (see test item *d*).

07 mastered the items *b*, *d*, *e* and *f* while student *15* mastered the items *a*, *b*, *c*, and *f*. Both students mastered items *b* and *f* (which is the set closure of their intensions) and together they mastered all problems (which is the set union of their intensions).

As a teacher, such kind of information might be of great interest since it helps to effectively arrange groups of students when aiming for collaborative, peer-learning (where students learn together in groups). In the example above, the students *07* and *15* together could be tutors for other students.

4.3 Learning Progress Over Time

The concept lattices shown in Fig. 4 and Fig. 5 are the result of a formal context which is an evaluation of the students′ performances at a certain point in time. However, in some cases it might be of great interest for a teacher to observe the learning progress over a longer period of time. Ideally, all students might be able to master all items at the end of course or the semester. In such a case, all cells in the formal context would be filled with crosses. This would result in a concept lattice with only a single formal concept. Fig. 6 exemplifies such a learning progress over time. The concept lattice in the middle results from adding one solved item to the students′ performance states (except for the students *03* and *17*). The concept lattice on the right results from adding an-other item to the student′s performance states. In general, the visual appearance of the concept lattice gives a first impression of the student's coherence with respects to their performance: A concept lattice which looks "complex" due to a large amount of formal concepts is a clear indication for a high diversity among the students′ performances.

5 Conclusions

In this paper we proposed to apply FCA to support students and teachers. Students apply the FCA-tool in order to learn a certain knowledge domain by interacting with the concept lattice which allows uncovering hidden relationships between domain′s concepts. Insofar, FCA is a powerful technique

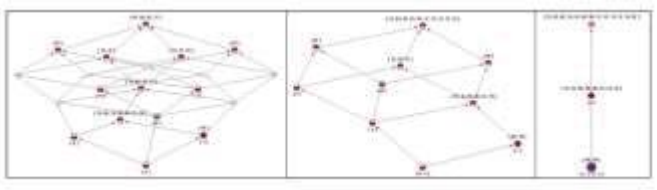


Fig. 6. Concept lattices changing over time reflect the learning progress of the class of students.

coming from AI and data mining research to support education in an intelligent way. In addition to that, a student´s reflection upon his or her learned and still-to-learn concepts is supported by an open learning modelling approach. Students can access the visualizations and compare themselves with peers. Teachers can apply the FCA tool to visualize the answers to a set of pedagogical questions which are of high interest for teachers.

We tried to formulate some of these pedagogical questions; they are the result of focus group studies and inter-views with teachers in the early phase of the LEA´s BOX project. The resulting visualizations as shown above are currently in the spotlight of formative, qualitative evaluation studies with small focused groups of teachers. Current work on the technical side of the project focuses on the development of interactive visualizations which can be easily used by teachers in the classroom. Early feedback of teachers concerns the complexity of the concept lattices, in particular when dealing with a great amount of problems (respectively competences and skills). Conceptual research and the elaboration of ideas on how to reduce this complexity without reducing the amount of information which can be extracted and deduced from the visualizations will be the main focus of our work in the near future.

Acknowledgements. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7) under grant agreement no. 619762 (LEA´s BOX). This document does not represent the opinion of the EC and the EC is not responsible for any use that might be made of its content.

6 References

[1] Dimitrova, V., McCalla, G. and Bull, S. (2007). Open Learner Models: Future Research Directions (Special Issue of IJAIED Part 2). International Journal of Artificial Intelligence in Education 17(3), 217-226.

[2] Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Buckingham Shum, S:, Ferguson, R., Duval, E., Verbert, K., and Baker, R.S..J.D. (2011). Open Learning Analytics: an integrated & modularized platform: Proposal to design, implement and evaluate an open platform to integrate heterogeneous learning analytics techniques. Available online at http://solaresearch.org/OpenLearningAnalytics.pdf

[3] Ferguson, R., and Buckingham Shum, S. (2012). Social Learning Analytics: Five Approaches. In Proceedings of the 2nd International Conference on Learning Analytics & Knowledge, 29 Apr - 02 May 2012, Vancouver, British Columbia, Canada.

[4] Duval, E., (2011). Attention Please! Learning Analytics for Visualization and Re-commendation. In Proceedings of the 1st International Conference on Learning Analytics & Knowledge, 27 Feb – 1 March 2011, Banff, Alberta, Canada.

[5] Wille, R. (1982). Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies. In: Ganter, B., Stumme, G., Wille, R. (eds) Formal Concept Analysis, pp 1-34, Berlin: Springer.

[6] Nesbit, J.C., Adesope, O.O. (2006). Learning With Concept and Knowledge Maps: A Meta-Analysis. Review of Educational Research. 76, 413–448.

[7] Bull, S., Kay, J.: Open Learner Models. Nkambou, R., Bordeau, J., Miziguchi, R. (eds.) (2011). Advances in Intelligent Tutoring Systems, pp. 318-388. Berlin: Springer.

[8] Arnold, K.E., Pistilly, M.D. (2012). Course Signals at Purdue: Using learning analytics to increase student success, In: Proceedings of the 2nd International Conference on Learning Analytics and Knowledge, pp. 267-270. ACM, New York.

[9] Rusch, A., Wille, R. (1996). Knowledge spaces and formal concept analysis. In: Bock, H.H., Polasek, W. (eds.) Data analysis and information systems: Statistical and conceptual ap-proaches, pp. 427-436. Springer, Berlin (1996)

[10] Korossy, K. (1999). Modeling Knowledge as Competence and Performance. In: Albert, D., Lu-kas, J. (eds.) Knowledge Spaces: Theories, Empirical Research, and Applications, pp. 103-132. Springer, Mahwah, NJ.

Relational Reinforcement Rule Induction and the Effect of Pruning

Hebah ElGibreen[1], Mehmet Sabih Aksoy[2]
[1]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh 11415, Saudi Arabia, hjibreen@ksu.edu.sa
[2]Information System Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia, msaksoy@ksu.edu.sa

Abstract: *Covering Algorithm (CA) is a Machine Learning field that produces a powerful repository represented as simple if-then rules. Although this field is well established with discrete data but it has its deficiency when dealing with numeric data. This paper introduces a new algorithm called RULES-CONT, which deal with continuous attributes using Relational Reinforcement Learning (RRL). This algorithm is a non-discretization algorithm that directly deal with all kind of data and transfer past experience to improve and generalize its learning. Different pruning levels will also be tested with RULES-CONT, in order to study the possibility of using pruning with RRL to reduce the time and solve the speed problem. The main contribution is to propose a novel solution for rule induction and analyse the most effective pruning level that should be integrated with RRL. Several experiments will be presented to compare RULES-CONT with other algorithms. These experiments are validated using 10-cross-fold validation and Friedman test to measure the significance between the algorithms and decide on the most suitable pruning level with RRL.*

Keywords – Covering Algorithm, RULES-CONT Algorithm, Pruning, Relational Reinforcement Learning

1. Introduction

Due to the excessive growth in technology, researchers and developers are currently striving for sustainability and adaptability of IS over different needs and requirement. One field of ML called Covering Algorithms (CA) has shown several characteristics that made it more attractive than statistical ML techniques that are very difficult to understand and act as a black box [1, 2]. It represents the discovered knowledge as rules, which are unambiguous, can be used by beginners, and easily updated, understood and validated. Rules can model complex interaction between attributes and, thus, represent any type of rules [3]. Its model simplicity makes it very easy to integrate and its transparency allows experts to reuse the discovered knowledge. Nevertheless, even though CA is domain independent, it was basically designed to deal with nominal values while attributes with infinite space were neglected. CA perform poorly with continuous attributes [4].

In the current literature of CA several attempts have been made to solve this problem, but these attempts are still lacking. From the state-of-the-art presented by Elgibreen and Aksoy [5], the continuous attributes were originally handled using discretization (online and offline). The deficiency of these approaches enclosed between the inflexibility of offline approach and computation complexity of online approach. Another approach was also developed by small group of researcher, called non-discretization. It does not discretize the attributes values and deal with both continuous and discrete attributes in the same manner. Although such approach solved the problems of discretization approaches but it caused a major issue in noise tolerance and speed.

In order to solve the noise tolerance problem and produce a knowledge repository that does not overfit its training set, the concept of relational reinforcement learning (RRL) was previously introduced in [6]. Although the use of RRL significantly improved the accuracy and noise tolerance but the speed was still an issue. It was found that RRL reduces the error rate and complexity but the model size and speed was increased. Thus, further investigation is needed to exploit the properties of RRL with numeric environment while dealing with the speed problem. One way to do so is by introducing a pruning technique, which is a technique used to reduce the search space and reduce the time spent in learning. Further investigation needs to be conducted to test what is the most suitable pruning level that can be used with RRL to deal with infinite space in timely manner.

The purpose of this paper is to propose an improved version of RULES-CONT algorithm that use pruning while dealing with the infinite space of continuous attributes using RRL. In particular, several level of pruning will be tested in this paper to choose what can be used with RRL for rule induction in numeric environment. This version exploits the properties of RRL and introduces human mind's aspects when dealing with continuous attributes. Consequently, instead of targeting a certain problem, the proposed algorithm can make the agent deal with different kind of problems using simple rules. It continuously updates its space so that past mistakes can be corrected automatically. RULES-CONT results in very simple and accurate rules which reduced the effect of large repository. It reduces the number of conditions in every rule to generalize the results and reduce the overspecialization, regardless of the total number of rules in the repository.

In order to test the performance of RULES-CONT, the KEEL tool [7, 8] is used with 10-fold cross-validation to compare eight different algorithms over 41 benchmarks and show its effect over five different measures. At the beginning, RULES-CONT pruning is tuned using different levels and the algorithms' performance is analyzed based on Demsar recommendation [9]. Friedman test is then applied in order to identify the most suitable level of pruning and to rank the algorithms in every measure. After that, the execution and process time is measured to show how RULES-CONT increase the speed, the rule complexity is measured to confirm that its result can be generalized over

noisy data, and the error rate is assessed to ensure that RULES-CONT is not badly affected by the speed increase.

This paper is organized as follows. First, the related work to the continuous attribute problem is discussed. Then, the proposed method and its details are explained. Following that, the algorithm is benchmarked and analyzed based on its empirical results. Finally, the result is discussed and the paper is concluded with some future work.

2. Related Work

In rule induction, there are two types of attributes: discrete and continuous. However, handling of continues values, in general, is a problem due to their infinite space and this domain is an active area of research. Consequently, different methods were developed and characterized based on the concept of discretization. The first type is to use any offline discretization method to converts all continuous attributes into discrete before inducing the rules. Different discretization techniques have been applied over CAs; such as SIA [10], ESIA [11], C&E [4], RULES-3+ [12], and Prism [13, 14]. It was found that offline discretization reduces the time of rule induction while it seriously affect the rules' quality [15]. There is a great tradeoff between the number of intervals and the consistency of the rule. Choosing a small number of split points increases the interval size and the resultant rules might not be consistent. Choosing a large number of split points reduce the interval size and affect the number of rules to overspecialize it.

Offline discretization, in general, does not consider future cases, and it fixes the intervals before rule induction process. However, in reality, some important data and measures can be extracted during the learning process. Thus, it can cause a major problem when the values of unseen data do not remain in the same distribution. As a result, online discretization was introduced to assign a fixed number of intervals during the learning process. In REP-based family, such as Slipper [16] and Ripper [17], an online discretization technique was introduced. It chooses the best split point online by sorting all the numerical values of the features repeatedly. They needed an intensive computation with every attribute in each example and at least three tables for every attribute.

Online discretization was also applied in RULES family by integrating it into RULES-SRI [18]. Instead of examining all the values of an individual, it only examines the boundaries of each continuous attribute during the learning process. The split points are added when there are adjacent values of the same attribute, and each one belongs to a different class, to create the boundary points. The execution time and re-computation of boundaries with each rule in this algorithm have tremendously increased. This method does not consider the attributes interdependency and only considered the classes relationship. Therefore, RULES-8 [19] was developed to discretize the continuous attributes during the learning process. It re-sorts the dataset to decide on the split point based on the seed attribute-value pair. Although it managed to handle the noise, it still has the same problem as REP-based family. It needs to re-sort the dataset with every attribute selection, which increase the computation complexity and execution time.

Online discretization, in general, can be more accurate than offline discretization. The result of the interval depends on the processed rule during the learning process. Hence, it is context-dependant and can consider the bias in the data. However, its computation cost is very high because it requires a high number of evaluations during the learning process. Thus, another type of algorithms has been developed to handle continuous attributes without any discretization. It deals with continuous and discrete values in similar manner and tries to discover the best threshold for an attribute depending on its relationship with other examples and classes. Small attempts have been made to handle continuous attribute in CAs without discretization. A modified version of AQ, called Continuous AQ (CAQ), was developed in [20] to handle both continuous and discrete attributes. It was proven that dealing with continuous attributes as a real number instead of forcing it into discrete representation would lead to more efficient results. However, CAQ could not obtain appropriate ranges because it did not consider all the data and only focus on the seed.

An algorithm called RULE-5 [15] was developed to deal with continuous attributes based on the examples' distribution. It was also improved into RULES-5+ [21] to reduce the role of statistical measures by applying a new knowledge representation. However, it caused the problem of overspecialization, and, thus, it became sensitive to noise. In addition, RULES-IS [22] also included a procedure to handle continuous values without discretization. During the generation of rules, numerical ranges were created with every continuous attribute to cover the positive examples. However, this algorithm would need to match every rule with all different conditions which need a lot of time and computation. Another rule based algorithm, called uRule [23], was developed to handle uncertain continuous attributes. It was built based on REP-based family, specifically Ripper, and used new heuristics to optimize and prune resulting rules, identify the optimal thresholds of continuous data, and handle uncertain values. During the learning process, the threshold that best divide the training data is decided based on extended information gain measures. In [24], the empirical result was presented to find that uRule can handle uncertainty in continuous and discrete attributes but it consumes a lot of time, either in uncertain or certain datasets, because of the complexity of rule pruning.

Ultimately, non-discretization algorithms in CA can deal with the infinite space of continuous attributes and induce good rules without the need of discretization. However, the speed of handling continuous attribute is a major problem that could affect its performance. Moreover, handling of continuous values should not cause overspecialization or make the results highly affected by noise. The answer to these problems is using pruning techniques. Unfortunately, however, even though using pruning is one of the mostly wide solutions to deal with speed and noise tolerance problem, choosing the most appropriate technique to use with a certain method is a challenge itself. Several empirical studies need to be conducted and statistical investigation is also necessary. For that reason, the following section presents a non-discretization solution for rule induction in numeric environment which adopt RRL properties with the most suitable pruning. This way it will be possible to deal with the infinite space of continuous attributes without the use of discretization while avoiding the current speed problem with the help of pruning.

3. RULES-CONT Algorithm

Due to the significance of numerical values and the deficiency of current approaches in CA, a new algorithm called RULES with Continuous Attributes (RULES-CONT) is proposed. It is a novel algorithm that is inspired from RULES family and the behavior of humans. It tries to learn from scratch and build its experience to deal with numerical values similarly to nominal by integrating the concept of RRL. RULES-CONT transfers other agents' knowledge in different domains to solve new problems by using the concept of Transfer Learning (TL), as in RULES-TL [25].

The main idea of RULES-CONT is to deal with the continuous attributes without discretization or fuzziness during the rule induction process. Hence, intervals are not fixed and approximation function is not needed. It discovers the best threshold for a label depending on its relationship with other examples and classes. RULES-CONT basically work the same way as any RULES algorithm. It starts by reading the training set and induces new rules until all examples are covered. However, the concept of TL is applied in RULES-CONT to reduce the induction time and take advantage of past knowledge. RRL is integrated to automatically deal with continuous attributes without discretization during the learning process. It is designed to be automatically and easily updated so that it can be directly integrated with any incremental rule induction.

In addition to RRL, RULES-CONT also adopt pruning with the RRL agent. In particular, it was tested with different pruning techniques available for the RULES family, as presented in [26-28], to find that PMC pruning is the most appropriate one for the proposed algorithm, as will be explained later in this section. PMC is a general incremental pruning technique that generalizes the rules during the induction process, instead of afterward. A threshold is used to specify the allowed misclassification level in order to generalize the results further. It controls the consistency of the resultant rules, where using a threshold greater than zero gives inconsistent results. To know the most suitable level of PMC pruning, empirical tests needs to be conducted as will be presented in the next section.

From Figure 1, RULES-CONT starts by reading the training set and initializing the parameters. Then, the initial state space is defined to include null rules for every class. The environment of RRL is still unknown and is built through the life time of the agent. This is because the state space will be different depending on the training set, and its actions are changing from seed example to another. Thus, only the initial state space is built at this phase, where a state is called rS and an action is rA. After initializing the state space, TL is applied similarly to RULES-TL. Hence, if there is a rule in the past that can cover an existing example then the example is marked as covered and the rule is stored in the final rules set. Following that, RULES-CONT initializes the PMC condition based on the PMC threshold level. Then, it starts to discover new rules to cover unmarked examples using RRL agent. It learns the best rule for a seed example without prior assumptions and handles all continuous and nominal attributes in one state.

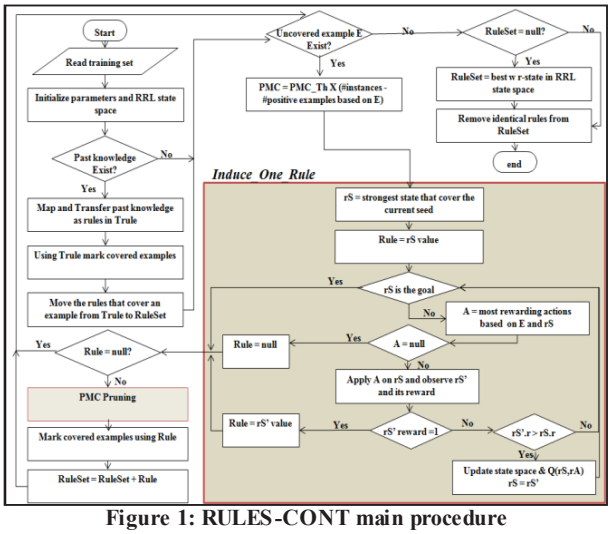


Figure 1: RULES-CONT main procedure

In this procedure, highlighted in Figure 1, RRL agent is activated to discover the best rule for every uncovered seed example. At the beginning, the strongest rS that cover the seed example is chosen as the initial state to make a shortcut and start from previously discovered states instead of always starting from null. After that, the current state value is chosen as the best rule. Then, the algorithm start to search for the goal state until it either reach the goal (consistent state) or reach a stop condition, where it cannot improve the learning afterword. Using RRL, the algorithm tries to reduce the searching time by stopping the search even if the goal is not found. This step will not cause a problem in the future because it is possible to find the goal in the next seed using the knowledge extracted in a previous step. Hence, it sometimes stops in a local minimum to reach a global one in the future, instead of mistakenly choosing weak results.

The search process continues to look for the best action based on the current state. The sub actions in every rA represent the possible values for every sub-state. Hence, for a condition (i), rA[i] represent the possible actions that can be assigned to one sub-state to go to another state. These actions are created based on the visited state and seed examples to avoid dealing with infinite space. Basically, when a seed example visits an rS, the most similar negative examples from every class are identified, using (1); where V^i_{E1} and V^i_{E2} are the values of the continuous attribute (i) in examples E1,E2 respectively, and V^i_{min} and V^i_{max} are the minimum and maximum values of attribute (i). The distance between discrete attributes is computed using (2).

$$D(E_1, E_2) = \sqrt{\sum_{\text{All cont.Attr}} \left(\frac{V^i_{E1} - V^i_{E2}}{V^i_{max} - V^i_{min}}\right)^2 + \sum_{\text{All disc.Attr}} d(A1, A2)} \quad (1)$$

$$d(A1, A2) = \begin{cases} 0 & \text{if } A1 = A2 \\ 1 & \text{if } A1 \neq A2 \end{cases} \quad (2)$$

In every sub-state (i.e. attribute) in each negative example an action A that belongs to rA is produced from one of the cases presented in Figure 2. Ei is the attribute (i) for the seed current example, Neg(i) is attribute (i) for the current similar negative example, and Min/Max functions return the minimum or maximum value of attribute (i) from all examples that are covered by current state (rS).

Moreover, from step one, missing values are handled automatically by considering no actions and overlapping is reduced by neglecting identical negative examples.

1. If (Ei.value = Neg(i).value) OR (Ei.value is null) OR (Neg(i).value is null) THEN return null.
2. Else if (Ei is nominal), THEN rA[i].add ('Equal (Ei, Ei.value)').
3. Else if (Ei is numeric), THEN
   a. If (Neg(i).value greater than Ei.value) THEN rA[i].add('GreaterEqual (Ei, Min(rS, i)) ∧ Less (Ei, Neg(i).value)')
   b. Else if (Neg(i).value smaller than Ei.value) THEN rA[i].add('Greater(Ei, Neg(i).value) ∧ LessEqual (Ei, Max(rS, i))').

Figure 2: rAction creation cases

Accordingly, the discovered actions are either ranges that represent the continuous attributes, or one value that represents the discrete attribute. To choose the strongest action, the action value (Q-value) of every action is initialized by none zero value using (3), where p is the number of examples that have its value in the same range as rA while (P+N) is the total number of examples covered by rS. Alternatively, if no actions are found or all the actions have already been tested in the current state, then no further improvement can be done using the current seed example. In that case, the stop condition is activated without finding the best rule. By stopping the search at this point, sticking into an infinite loop is prevented and discovered information will be stored into the state space for later use.

$$\text{Initial } Q(rS, rA) = \frac{p_{rA}}{(P+N)_{rS}} \quad (3)$$

If a new action is found, the algorithm will first mark the action as tested. Then, the new state will be observed after applying the action, and its reward will be computed using (4), where S is the sub-state values representing the rule conditions and n is the number of negative examples covered by rS. From this reward, RULES-CONT gives the max penalty to null states in order to avoid rules that have no conditions. This is because null rules cannot be used in the classification model. Moreover, the reward will be one if the rule does not cover any negative examples. In this case, the rule discovered is the goal, and the search should stop. Nevertheless, if the current state is neither the goal nor the null state, then the agent should be guided toward the goal. The minus percentage of covered examples is used because covering fewer negative examples is better. Hence, more reward value is closer to the goal until n (number of negative examples) is 0, which is the goal.

$$r(rS) = \begin{cases} -1 & if\ all\ S \in rS\ are\ null \\ 1 & if\ n = 0 \\ -\frac{n}{n+p} & else \end{cases} \quad (4)$$

If the new state is better than the older but it is still not the goal, the action value function of rS is updated and the search continues from the new state. However, if the new state is consistent, i.e. its reward is one, then the goal is reached and the best rule will be updated with the value of the goal state. Finally, after reaching the stop condition or finding of the goal, the best rule will be returned as the discovered rule. With every discovered rule, PMC pruning is applied. The pruning procedure, presented in Figure 3, generalize the rule by removing one condition at a time until the average number of negative examples is of the current rule become less that the previous one. After that, the last rule before the stop condition will be returned to mark all covered examples. The resulting rule is represented as a relational string that represents the common characteristics of the attributes with their appropriate values; such as {(A1, "K") ^ (A2, (1,5]) ^ (A3,[5,8)) ^ (A4,8) => (class, T)}. This string can be stored in the agent repository to be used in future prediction and discovery.

New rule = nR; Original Rule = R; number of negative examples = n; number of positive example = p.

1. If ((R.n + R.P) < (0.1 * total number of examples)
   a. While (R.n < PMC)
      i. nR = R & R = generalize R until ((R.n/(R.n+R.p)) < ((nR.n/(nR.n+nR.p))

Figure 3: PMC pruning

After pruning a rule, it is then used to cover any matching example, and its value is added to final rule set. At the end, if the goal is not yet found, weaker rules that are near the goal will be taken as the final rule set using beam width search. Finally, identical rules will be removed, and the resulting rule set can be used to classify future unseen examples.

In order to predict future actions three cases occurred with RULES-CONT, as follows.

- **One Rule:** When only one rule covers the example then use this rule for prediction.
- **No Rule:** When no rules are found then the most similar rule is chosen based on its distance. The rule distance is computed using (5, 6, and 7); where Di is the discrete value for attribute (i), *Cj* is the continuous value for attribute *(j), R.C.max* and *R.C.min* are the maximum and minimum value of the condition range, and *C.max* and *C.min* are the maximum and minimum values of the continuous attributes in the training set. If more than one rule has the same distance, the strongest one is used.

$$\boldsymbol{Dis(R,E) = \sqrt{\sum_{\forall Discrete\ attributes} d(R.Di, E.Di) + \sum_{\forall\ Continuous\ attributes} d(R.Cj, E.Cj)}} \quad (5)$$

$$\boldsymbol{d(R.D, E.D) = \begin{cases} 0 & if\ R.D = E.D \\ 1 & if\ R.D \neq E.D \end{cases}} \quad (6)$$

$$\boldsymbol{d(R.C, E.C) = \begin{cases} 0 & if\ E.C\ value \in R.C\ range \\ \left(\frac{\min(|E.C - R.C.max|, |E.C - R.C.min|)}{C.max - C.min}\right)^2 & else \end{cases}} \quad (7)$$

- **Conflicted Rules:** When more than one rule covers the example with different class choosing the most similar rule is not an appropriate prediction procedure. This is because the rules have been modified and not in their original form. Thus, the strongest rule will be used to classify the new example using (4).

4. Experiment

In order to study the behavior of RRL agent with different level of pruning, the KEEL tool [7, 8] with 10-fold cross-validation are used to conduct the experiments on a PC with IntelR _CoreTM i7 CPU, 2.40 GHz processes, and 16 GB RAM. The experiments study the behavior of RULES-CONT in three level of pruning (0.05, 0.10, and 0.20) in addition to no pruning. This is because the accuracy reaches an undesired value when the threshold is more than 0.2. Five different measurements are recorded in every level over 41 datasets. These datasets were taken from the KEEL dataset repository [8] and gathered as a real-life sample to serve a certain domain.

The experiments compare RULES-CONT with five CA. From RULES family, RULES-5+ is tested as the predecessor of RULES-CONT, while the RULES-6 and SRI are scalable algorithms that deal with continuous attribute using offline discretization. Ripper and Slipper are REP-based algorithm that handles continuous attribute using online discretization. Moreover, two DT algorithms that deal with continuous attribute using online discretization were also included to ensure that RULES-CONT performance not only improved CA but also DT algorithms. These algorithms are C45RulesSA [29] and DT_GA.

At the beginning of the experiments, it was important to study the behavior of pruning over RULES-CONT and how increasing the pruning level effects the measurements. PMC threshold controls the number of negative examples allowed to be covered. Increasing the PMC threshold would result in pruning more rules to cover more examples. However, such control needs to be studied to ensure that the accuracy, speed, and complexity of RULES-CONT are not badly affected. Therefore, RULES-CONT is applied with different threshold values to analyze its performance. In this section, for better visualization, the relationship between the PMC threshold and the measures are illustrated through the average value over all datasets.

Staring from the pruning behavior, Figure 4 presents all measurements and its relationship with the PMC threshold increase. When increasing the PMC threshold, the error rate, number of rules, and execution time decreases. Therefore, increasing the PMC threshold improves the learning rate and speed of RULES-CONT and badly affect its accuracy. However, the effect on the number of rules and speed is much more than the effect on the error rate. In fact, such reduction in the accuracy is sometimes desired in order to resist noise in future cases and reduce the overfit effect. Hence, the control of PMC threshold allows the user to decide if the current accuracy should be sacrificed for the sake of future precise prediction.

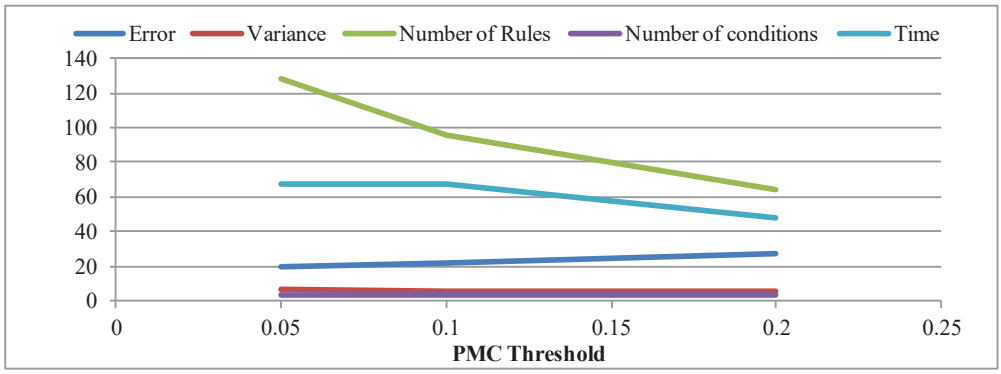


Figure 4: PMC threshold effect on RULES-CONT average measures over all datasets, including error rate, variance, number of rules and conditions, and time

When it comes to the overfit and rule complexity, it can be noticed that MPC threshold has lower effect on these measures. Whatever the PMC value is, the variance and number of conditions are barely changed. RULES-CONT preserves its generality and rules' simplicity at any PMC threshold value. In fact, the increase in the threshold slightly decreases the variance and number of conditions, and thus further reduces the complexity.

In addition to the effect of PMC threshold, it is important to provide accurate evidence about the pruning over the detailed performance in order to choose the best version based on the total performance. However, instead of confusing the reader with excessive analyses regarding the detailed performance, it was decided to study all versions of RULES-CONT in this section and show the significance effect of each one and summarize the results using the statistical measures. Based on Demsar [9] recommendation, all algorithms are statistically compared using Friedman test and Nemenyi post hoc test. This test is applied to rank the algorithms and measure their significance, as illustrated in **Figure 5**. Note that the CD line is not presented in the graphs because the target of this section is to analyze all versions of RULES-CONT. Hence, there is no single algorithm to compare with but instead four versions of RULES-CONT. The significance can be easily seen through the group lines, where algorithms that belong to the same group have insignificant difference. From the error rate (**Figure 5**.a), RULES-CONT is one of the leading algorithms. RULES-CONT with no pruning, 5%, and 10% pruning comes in the first group with RULES-5+ and Slipper but is significantly better than the other algorithms. RULES-CONT that has no or small pruning are in first and second place and produce the lowest error rate.

In the variance (Figure 5.b), RULES-CONT also belongs to the first group, with RULES-6 and SRI, when pruning is applied (5%, 10% and 20% PMC). When no pruning is used, it's variance also belongs to the same group as RULES-6 but is significantly higher than the variance of RULES-SRI. Whether pruning is applied or not, RULES-CONT's variance is significantly better than its preceding non-discretization (RULES-5+) and online discretization CA (Slipper and Ripper). Hence, RULES-CONT improves the variance of CA and solves the overspecialization problem of its preceding non-discretization version.

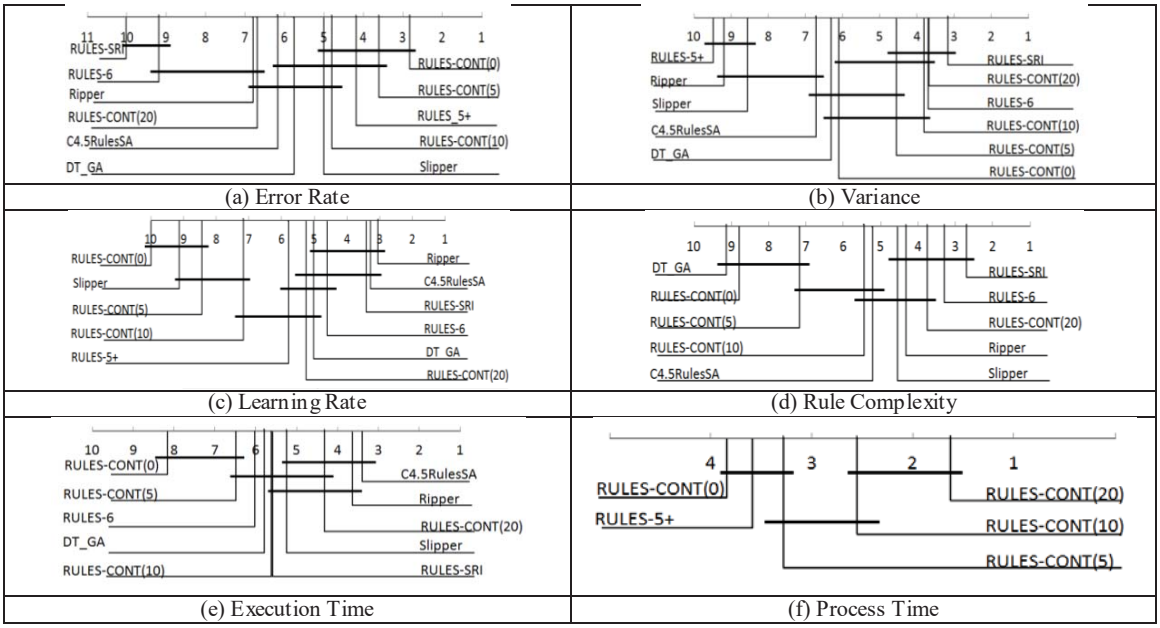


Figure 5: CD diagrams of Friedman two-tailed Nemenyi post hoc test for clear data

In the learning rate (Figure 5.c) and rule complexity (Figure 5.d), RULES-CONT with a small pruning percentage gives an intermediate number of rules and conditions in order to avoid overgeneralization and overspecialization problems. However, if simplicity is more important, then a higher pruning percentage can be used because it will significantly reduce the number of rules as seen in RULES-CONT with 20% pruning.

When comparing the speed of RULES-CONT with the other discretization-based algorithms (**Figure 5**.e), it can be noted that the speed of RULES-CONT significantly changes depending on the PMC threshold. When the pruning threshold is high (20%), its speed belongs to the first group and has insignificant difference to all discretization-based algorithms. When PMC threshold is 10%, the speed of RULES-CONT belongs to the second group with insignificant difference to all discretization-based algorithms except for C4.5RulesSA. When the PMC threshold is 5%, RULES-CONT has insignificant difference to its version with no pruning in addition to all algorithms except for C4.5RulesSA and Ripper. However, the speed of RULES-CONT when no pruning is applied is highly affected due to its preservation of all the rules induced, which significantly increases its execution time. Such a difference is natural since all other algorithms use pruning so it is natural to be slower when pruning is not applied. Finally, when considering the process time of RULES-5+ (**Figure 5**.f), the speed of the non-discretization approach is improved through the use of RRL in RULES-CONT. RULES-CONT improved the speed problem of the non-discretization approach and, when pruning is applied, regardless of its degree, RULES-CONT maintains a similar speed to the other CA and DT algorithms.

5. Discussion

RULES-CONT considered the problems of existing numerical attributes and can directly deal with noise without pruning. By using RRL, RULES-CONT can deal with multiple continuous attributes at the same time while taking advantages of the information gathered from their common characteristics. Furthermore, since RRL has the property of continuous learning so updating the extracted knowledge and integrating RULES-CONT into incremental learning can be directly done without additional procedures. In addition, because of the concept of TL, the induction time can be reduced, and past knowledge is exploited.

From the practical test it was concluded that RULES-CONT solved the speed problem recognized in the literature and introduce noise tolerance while improving its accuracy. From the Friedman tests, RULES-CONT significantly improved the accuracy (error rate and variance) and speed of RULES family. It resisted noise the best when no pruning was introduced, but when the model's complexity is important the version that works the best is when 5% pruning is introduced. Hence, depending on the need, the PMC percentage can be increased. If accuracy and noise tolerance are what matters, then 0% pruning can be used. If speed and simplicity are more important, then 10% pruning and more can be used. However, if all the measures are important, then 5% pruning is the most appropriate option.

From all above, RULES-CONT not only introduce new property into CA but also filled part of the gaps in non-discretization approach. When it comes to the speed, RULES-CONT reduced the process time tremendously due to knowledge transfer and re-use; the overspecialization is

reduced by significantly reducing the rules' complexity, where the resulting rules were noticeably generalized. Therefore, it can be useful for decision-makers and its results can be reused by non-experts, not only to let the machine decide but also to understand how the system works, acquire a new knowledge, and gain more experience. RULES-CONT does not need to trade its performance to accurately deal with infinite space.

6. Conclusion

Due to the excessive change in human desires and future interest, the knowledge collected from experts is frequently changing in ML and knowledge comprehensibility is highly desired by decision makers. However, current methods in the literature of CA are still lacking when applied over continuous attributes. This paper introduced a new non-discretization approach based on RRL. The approach was adopted with pruning in order to manage the infinite space of continuous attributes. Through this paper, it was possible to study what is the most suitable pruning level for RRL in rule induction. From the experiments, it was verified that RULES-CONT filled the current gaps of CA in infinite space. It improved the accuracy of CA, does not overfit its training set, and solved the speed and overspecialization problems while preserving similar complexity to past non-discretization approaches. The problems of discretization were avoided because it applies a non-discretization approach, where intervals are not fixed. Its result generality can be easily controlled by modifying a simple pruning threshold which made it possible to control its performance.

Acknowledgement: *This research is supported by a grant from King Abdulaziz City for Science & Technology. We also thank Dr. Samuel Bigot for his cooperation in providing us with RULES-5+.*

7. References

[1] M. S. Aksoy, H. Mathkour, and B. A. Alasoos, "Performance Evaluation of RULES-3 Induction System for Data Mining," *International Journal of Innovative Computing, Information and Control,* vol. 6, pp. 3339–3346, 2010.

[2] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques," *Informatica,* vol. 31, pp. 249-268, 2007.

[3] K. Dembczyński, W. Kotłowski, and R. Słowiński, "ENDER: a statistical framework for boosting decision rules," *Data Mining and Knowledge Discovery,* vol. 21, pp. 52-90, 2010/07/01 2010.

[4] C. Chiu and N. S. Chiu, "An adapted covering algorithm approach for modeling airplanes landing gravities," *Expert Systems with Applications,* vol. 26, pp. 443-450, 2004.

[5] H. ElGibreen and M. S. Aksoy, "Continuous Features in Inductive Learning and the Effect of RULES Family," *International Journal of Computers,* vol. 8, pp. 66-75, 2014.

[6] H. Elgibreen and M. Aksoy, "Adopting Relational Reinforcement Learning in Covering Algorithms for Numeric and Noisy Environments," *International Journal of Computational Intelligence Systems,* vol. 9, pp. 572-594, 2016.

[7] J. Alcalá-Fdez, L. Sánchez, S. García, M. J. d. Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, J. C. Fernández, and F. Herrera, "KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems," *Soft Computing,* vol. 13, pp. 307-318, 2009.

[8] J. Alcalá-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework," *Journal of Multiple-Valued Logic and Soft Computing* vol. 17, pp. 255-287, 2011.

[9] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets," *Journal of Machine Learning Research,* vol. 7, pp. 1-30, 2006.

[10] S. W. Wilson, "Classifier systems and the animat problem," *Machine Learning,* vol. 2, pp. 199-228, 1987.

[11] J. J. Liu and J. T.-Y. Kwok, "An Extended Genetic Rule Induction Algorithm," in *Proceedings of the 2000 Congress on Evolutionary Computation* La Jolla, CA, 2000, pp. 458-463.

[12] T. Pham and S. S. Dimov, "An efficient algorithm for automatic knowledge acquisition," *Pattern Recognition,* vol. 30, pp. 1137–1143, 1996.

[13] J. Cendrowska, "PRISM: An algorithm for inducing modular rules," *International Journal of Man-Machine Studies,* vol. 27, pp. 349-370, 1987.

[14] F. Stahl, M. Bramer, and M. Adda, "PMCRI: A Parallel Modular Classification Rule Induction Framework," in *Machine Learning and Data Mining in Pattern Recognition*. vol. 5632, P. Perner, Ed., ed: Springer Berlin / Heidelberg, 2009, pp. 148-162.

[15] D. Pham, S. Bigot, and S. Dimov, "RULES-5: a rule induction algorithm for classification problems involving continuous attributes," in *Institution of Mechanical Engineers*, 2003, pp. 1273-1286.

[16] W. W. Cohen and Y. Singer, "A simple, fast, and effective rule learner," in *Sixteenth National Conference on Artificial Intelligence*, 1999, pp. 335-342.

[17] O. Dain, R. K. Cunningham, and S. Boyer, "IREP++, a Faster Rule Learning Algorithm," in *SIAM International Conference on Data Mining - SDM*, 2004.

[18] D. T. Pham and A. A. Afify, "Online Discretization of Continuous-Valued Attributes in Rule Induction," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science,* vol. 219, pp. 829-842, 2005.

[19] D. Pham, "A novel rule induction algorithm with improved handling of continuous valued attributes," Doctor of Philosophy, School of Engineering, Cardiff University, Cardiff, 2012.

[20] B. L. Whitehall, S. C. Y. Lu, and R. E. Stepp, "CAQ: A machine learning tool for engineering," *Artificial Intelligent Engineering,* vol. 5, pp. 189–198, 1990.

[21] S. Bigot, "A new rule space representation scheme for rule induction in classification and control applications," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering,* 2011.

[22] D. T. Pham and A. J. Soroka, "An Immune-network inspired rule generation algorithm (RULES-IS)," in *Third Virtual International Conference on Innovative Production Machines and Systems*, WhittlesDunbeath, 2007.

[23] B. Qin, Y. Xia, R. Sathyesh, S. Prabhakar, and Y. Tu, "uRule: A Rule-based Classification System for Uncertain Data," in *IEEE International Conference on Data Mining Workshops (ICDMW)*, Sydney, NSW, 2010, pp. 1415- 1418.

[24] B. Qin, Y. Xia, and S. Prabhakar, "Rule induction for uncertain data," *Knowledge and Information Systems,* vol. 29, pp. 103-130, 2010.

[25] H. ElGibreen and M. S. Aksoy, "Multi Model Transfer Learning with RULES Family," in *International Conference on Machine Learning and Data Mining MLDM'2013*, New York, 2013.

[26] D. T. Pham, S. Bigot, and S. S. Dimov, "A rule merging technique for handling noise in inductive learning," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science,* vol. 218, pp. 1255-1268, 2004.

[27] D. T. Pham and A. A. Afify, "Three New MDL-Based Pruning Techniques for Robust Rule Induction," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science,* vol. 220, pp. 553-564, 2006.

[28] K. Shehzad, "Simple Hybrid and Incremental Postpruning Techniques for Rule Induction," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 25, pp. 476-480, 2013.

[29] J. R. Quinlan, *C4.5: Programs for Machine Learning*, First ed. San Francisco: Morgan Kaufmann 1993.

Deep Learning Architectures for Hard Character Classification

Vy Bui[1], **Lin-Ching Chang**[1]
[1]Department of Electrical Engineering and Computer Science, The Catholic University of America, Washington, District of Columbia, USA
01bui@cua.edu, changl@cua.edu

Abstract – *Recent research indicates that deep learning has achieved noticeably promising results in a wide range of areas such as computer vision, speech recognition and natural language processing. This paper offers an empirical study on the use of deep learning techniques for hard characters recognition on the notMNIST dataset. The MNIST dataset has been widely used for training and testing in the field of machine learning such as for the performance comparison of different deep learning algorithms. However, similar performance evaluation using the notMNIST dataset has not been reported. This dataset is harder and much less clean than the MNIST dataset. In this paper, we constructed several experiments to evaluate various deep learning architectures and proposed a multi-layer convolutional neural network for large-scale hard character classification on the notMNIST dataset. The result shows that our method can achieve 98% accuracy of classification. Comparisons were also performed against conventional fine tuning models such as logistic classifier and shallow neural network to demonstrate that well-constructed deep neural networks can significantly improve the accuracy of hard character classification on the notMNIST dataset.*

Keywords: Deep learning, character recognition, neural network, convolutional neural network, image classification

1 Introduction

Recently, learning through deep models draws significant attention. Various deep learning architectures such as deep neural networks, deep convolutional neural networks, and recurrent neural networks have been applied to a wide range of fields like computer vision, automatic speech recognition, or natural language processing. The concept of deep learning is to discover the features in the input data in multiple levels of representation, in other words, higher level features represent more abstract semantics of the data. Deep learning techniques have been shown to produce state-of-the-art results on several difficult tasks [1-5]. For examples, computer vision tasks such as image classification based on visual content is a very challenging task, mostly because of the intra-class variation, scale variation, viewpoint variation, deformation or occlusion. Numerous efforts have also been made to recognize handwritten digits on the famous MNIST dataset [6] using conventional machine learning techniques like linear/nonlinear classifier, K-nearest neighbors, and SVMs, or deep learning architectures like deep neural networks and deep convolutional neural networks [1-3]. These studies have reported that the deep learning architectures outperform the conventional methods and are able to achieve near-optimal human performance.

In this paper, we constructed several experiments of large-scale character datasets drawing from the notMNIST dataset [7] to evaluate various deep learning architectures. We then proposed a five-layer deep convolutional neural network for character recognition on the notMNIST dataset. This classification task is more challenging comparing to the handwritten digits recognition on the MNIST dataset. We also compared the classification accuracy of the proposed architecture against several conventional fine tuning models such as logistic classifier and shallow neural networks.

The rest of the paper is organized as follows. In section 2, we provide the necessary theoretical background of neural networks. Section 3 describes the notMNIST dataset and the preprocessing steps used in this study. The detailed description of the conducted experiments, the proposed model, and the testing results are presented in section 4, followed by the conclusions in section 5.

2 Theoretical Background

Character recognition using different features has been extensively studied, and a nice survey on feature extraction methods for character recognition can be found in [8]. In this section, we provide the necessary theoretical background of a neural network that will help our readers to understand the empirical experiments we have performed. Two types of neural networks will be described: regular neural networks and convolutional neural networks. The specific architectures used in our experiments will be discussed in section 4.

2.1 Score Function

A score function maps the pixel values of an input image to a confident score for each class. A learning system computes a score function that can be defined in equation (1):

$$Y_i = F(X_i, W, b) \qquad (1)$$

, where X_i $(i = 1, \dots, N) \in R^D$ is the i-th input image with D dimension and N is the total number of training images. W and b represent the collection of adjustable parameters in the system. The output $Y_i \in 1 \dots K$ is the recognized class label of a given image X_i, or the score associated with each class, and K is the total number of distinct classes. For example, in notMNIST we have a training set of $N = 500k$ images, the dimension of each image is $D = 28 \times 28 \times 1 = 784$, and the

number if classes is $K = 10$ (A-J). A simplest score function can be represented as a linear classifier:

$$Y_i = F(X_i, W, b) = WX_i + b \qquad (2)$$

The pixels of an input image X_i is arranged as a single column vector of size $D \times 1$. W is a weight matrix of size $D \times K$ and the bias vector b of size $[K \times 1]$ which contains the adjustable parameters of the score function. W and b determine the behavior of the score function, and directly influence the output scores Y_i. For the notMNIST dataset, the size of X_i is $[784 \times 1]$; W is $[10 \times 784]$; b and Y_i is $[10 \times 1]$.

In our experiments, we implemented a popular linear classifier called *Softmax* [9]. The advantage of *Softmax* is that instead of treating the output Y_i as un-calibrated scores for classes, it converts these scores to a vector of values between zero and one with the sum of the values equals one. Therefore, these scores can be interpreted as the probabilities for classes:

$$P(Y_i|\, X_i; W) = S(Y_i) = \frac{e^{Y_i}}{\sum_j e^{Y_j}} \qquad (3)$$

2.2 Cross-Entropy Loss Function

In equation (3), X_i and Y_i are fixed, and we can adjust W and b to optimize an objective function. Therefore, the goal is to fine tune these parameters so that the output probabilities of Y_i given X_i best match the ground truth labels L_i. An error function measures the discrepancy (cross-entropy) between the correct output for an input pattern and the output produced by the system can be defined in equation (4).

$$\mathcal{E}_i = \mathcal{D}\big(L_i, S(Y_i)\big) = -\sum_i L_i \, log(S_i) \qquad (4)$$

, where S_i is defined in eq. (3) and L_i is the correct output (or true label), and $\mathcal{D}$ is the discrepancy between the true label and the output score.

A loss function measures the average of errors $\mathcal{E}_i$ over a set of labeled samples (i.e., training dataset $\{(X_1, L_1), \ldots (X_N, L_N)\}$ can then be defined as equation (5)

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{E}_i^{train} \qquad (5)$$

Therefore, the objective of the training task is to find the optimal values of W and b that minimize the loss function $\mathcal{L}$.

It is known that the gap between the error rate on the test set $\mathcal{E}^{test}$ and the error rate on the training set $\mathcal{E}^{train}$ will drop with the increased number of training samples [1].

2.3 Optimizing Loss Function

2.3.1 Gradient Descent

Optimization is the process of finding the set of parameters W and b that minimize the loss function. In other words, we are searching a moving direction in the weight space to improve the weight vector so that it will return a smaller loss. The best moving direction is related to the direction of the steepest descend or the gradient of the loss function with respect to the parameters. There are many ways of performing the optimization, but Gradient Descent (GD) is probably the most popular way to optimize the loss functions in neural networks. Here two gradient descent algorithms are brief described: Simple GD and Stochastic GD.

The simplest minimization procedure is the simple gradient descent algorithm where W is iteratively updated in the negative direction of the gradient as shown in equation (6):

$$W_k = W_{k-1} - \epsilon \frac{\partial \varepsilon(W)}{\partial W} \qquad (6)$$

, where ϵ is the step size (or so called the learning rate) that decides how far along this direction in the next iteration. The complexity in evaluating the gradient is linearly related to the number of parameters in the system. In our case, we have 7840 parameters and therefore 7840 evaluations of the loss function are needed in each iteration to obtain the best set of W and b. The computational complexity will get worse as the modern neural networks often have millions of parameters in training data. Apparently, this approach is not scalable.

To address this large-scale problem, one can estimate the loss by computing the gradient over a small subset (i.e. a mini-batch) of training data. This approach is called Stochastic Gradient Descent (SGD), which is also known as the on-line update. The size of the mini-batch is an adjustable parameter to be determined which is usually constrained by the capacity of hardware memory. Common mini-batch sizes range between 50 and 256 samples, but can vary for different applications.. Equation (6) can then be modified, so an approximated version of average gradient will be used in the next iteration.

$$W_k = W_{k-1} - \epsilon \frac{\partial \varepsilon_{i_k}(W)}{\partial W} \qquad (7)$$

This procedure makes the parameter vector fluctuates around an average trajectory, but usually converges considerably faster than simple gradient descent on a large training set [1].

2.3.2 Gradient Back- Propagation

The gradient-based techniques are limited to linear systems; therefore the back-propagation algorithms are introduced to calculate the gradient in non-linear systems.

Let's assume that a system can be built as a cascade of modules, each of which is a function $X_k = F_k(W_k, X_{k-1})$,

where X_k is a vector representing the output of the module, W_k is the vector of tunable parameters in the module, and X_{k-1} is the previous module's input vector. If the partial derivative of $\mathcal{E}_i$ with respect to X_k is known, then the partial derivatives of $\mathcal{E}_i$ with respect to W_k and X_{k-1} can be computed using the backward recurrence:

$$\frac{\partial \mathcal{E}_i}{\partial W_k} = \frac{\partial F}{\partial W}(W_k, X_{k-1}) \frac{\partial \mathcal{E}_i}{\partial X_k} \qquad (8)$$

$$\frac{\partial \mathcal{E}_i}{\partial X_{k-1}} = \frac{\partial F}{\partial X}(W_k, X_{k-1}) \frac{\partial \mathcal{E}_i}{\partial X_k} \qquad (9)$$

, where $\frac{\partial F}{\partial W}(W_k, X_{k-1})$ is the Jacobian of F with respect to W evaluated at (W_k, X_{k-1}), and $\frac{\partial F}{\partial X}(W_k, X_{k-1})$ is the Jacobian of F with respect to X evaluated at (W_k, X_{k-1}). Equation (8) computes some terms of the gradient of $\mathcal{E}_i$, while equation (9) produces a backward recurrence. We can average the local gradients over the training set to obtain the full gradient.

2.4 Multi-Layer Neural Networks

To describe multi-layer neural networks, we begin with the simplest neural network − one which includes only a single neuron/node. This neuron is a computational unit that can take several inputs $X_1, X_2, \ldots, X_N$, and generates an output which is given by equation (10)

$$F(W^T X) = F\left(\sum_{i=1}^{N} W_i X_i + b\right) \qquad (10)$$

, where $F: \Re \to \Re$ is called the activation function. In this study, the popular Rectified Linear Unit (ReLU) [10] and the recent introduced Exponential Linear Unit (ELU) [11] were selected to be the activation function. The ReLU activation function can be defined by equation (11),

$$F(x) = \max(0, x) \qquad (11)$$

The ELU activation function can be defined by equation (12),

$$F(x) = \begin{cases} x & if\ x > 0 \\ \alpha(e^x - 1) & if\ x \le 0 \end{cases} \qquad (12)$$

A single node neural network which uses ReLU activation function is shown in Fig. 1. The ReLU can be replaced by the ELU activation function in some applications.

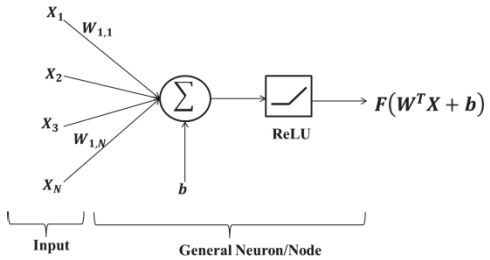


Fig. 1. A diagram of 1-hidden layer neural network with rectified linear units (ReLU)

A multi-layer neural network often have more than one hidden layers followed by an output layer of neurons. Multiple layers of neurons with nonlinear activation function allow the network to learn nonlinear relationships between input and output vectors. An example of two-layer neural network is shown in Fig. 2 where ELU is used in the first layer and ReLU in the second layer.

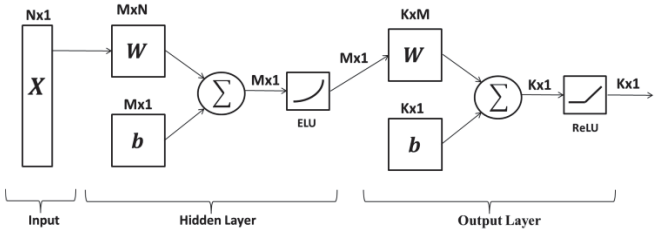


Fig. 2. A diagram of a two-layer neural network with M ELUs in its hidden layer and K ReLUs in its output layer

2.5 Convolutional Neural Networks (CNNs)

A general architecture of a convolutional neural network (CNN) usually consists three core layers: Convolutional Layer, Pooling Layer, and Fully-Connected Layer. A neural network without the convolutional and pooling layers is known as the regular neural network as we have mentioned in section 2.4. These three layers are stacked to form a complete CNN architecture.

The convolutional layer is the core building block of a CNN. This layer comprises of a set of learnable filters. During the forward pass, each filter is convolved across the whole input volume to produce a 2D activation (or feature) map. In other words, features were extracted from the input volume and give to the next layer to extract more complex features. In the context of image recognition, features can be patterns in lower layers and object parts in higher layers. For example, the first convolutional layer can take a raw image as an input, then different neurons along the depth dimension may activate in presence of various oriented edged, or blobs of color.

The spatial extent of the connectivity along the depth axis is always equal to the depth of the input volume. For example, suppose that the size of the input volume is [28×28×1], and the filter is 4×4 as shown in Fig. 3, then each neuron in the convolutional layer can have weights cover a [4×4×1] region in the input volume, and for a total of 16 weights. Since notMNIST consists of grayscale images, thus the depth of our input volume is 1.

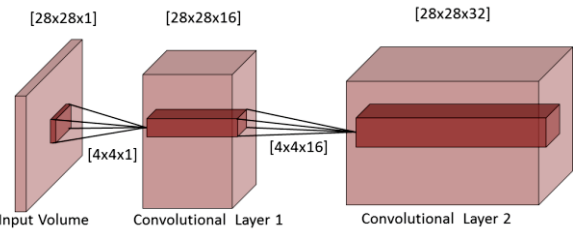


Fig. 3. A diagram of 3 convolutional layers

It is a common practice to periodically insert a pooling layer in between successive convolutional layers in a CNN. A pooling layer is used to progressively reduce the spatial size of the representation. It reduces the number of parameters and the computation time needed in training the network. In some applications, it is also used to control the possible overfitting. In this paper, the pooling layer functions independently on every slice along the depth of the input image. It reduces the dimension (width×height) of the input using the MAX operation. The MAX operation gives maximum activations over non-overlapping sub-regions of the input volume. For example, suppose the size of input volume is [4×4×16], in max-pooling, a pooling unit outputs the maximum activation in the 2×2 input region, as illustrated in Fig. 4. After pooling we have the output volume of size [2×2×16]. Notice that the volume depth is unaffected.

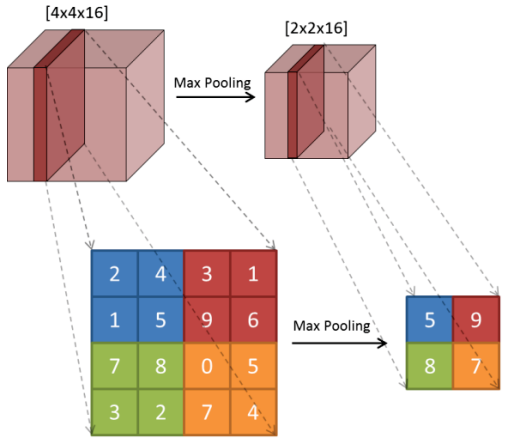


Fig. 4. A max pooling with a filter of size 2×2

The third layer in a CNN is a fully-connected layer that is considered as a regular neural network that we have discussed in section 2.4.

2.6 Overfitting in Neural Networks

Overfitting is a major problem for deep neural networks especially for deep learning that has multi-layer and contains a large number of parameters. Regularization and Dropout have been efficiently used to prevent neural networks from overfitting in training. One of the most popular methods for regularization is the L2 regularization [12]. The idea is to penalize the squared magnitude of all parameters directly in the loss, i.e. for every weight W in the network, we add the term $\frac{1}{2}\lambda W^2$ to the loss, where λ is the regularization strength.

Another recently introduced approach is the Dropout approach [13]. It randomly drops units from the neural network during training that prevents units from co-adapting. In other words, dropout is employed by keeping only a neuron active with a certain probability during the training; otherwise setting its value to zero.

3 Data and Preprocessing

3.1 notMNIST

The notMNIST dataset is designed to be similar to the famous digits MNIST dataset, while appearing a bit more like real data since the data is a lot less clean compared to the MNIST dataset. The dataset is made of characters rendered in a diversity of fonts of 28×28 gray-scale images. The labels are 10 classes from 'A' through 'J'. The original released training set has about 500k labeled images and the test set about 19k images. In this paper, we further split the original training set into two parts including a training set and a validation set for parameter tuning. Furthermore, it is necessary to balance the data across classes. In particular, the training set contains 52900 images and the testing set comprises about 1870 images for each class.

3.2 Data Preprocessing

Normalization is a common practice in data pre-processing stage in which the data is transformed to a specific range such as between 0 and 1, or between −1 and +1. In this paper, image normalization is required since the signal intensity values of raw images In particular, the data is normalized to have approximately zero mean and standard deviation about 0.5 using equation (13). After normalization, the intensity value of images will be ranged between −0.5 and 0.5.

$$\frac{x - \frac{pixel_{depth}}{2}}{pixel_{depth}} \qquad (13)$$

, where x is the signal intensity value of a pixel in the input image, and the pixel depth, $ixel_{depth}$ is 255 since the images are in 8-bit gray scaled format.

4 Experiments and Discussions

In this section, we provide the detailed description of six experiments we have performed. We showed how we utilize multi-layer and convolutional neural networks in hard character recognition. We implemented these architectures using Python 2.7 with Tensorflow 0.7 API [14]. The implemented models were trained and tested on Amazon Elastic Compute Cloud (Amazon EC2) [15]. The Amazon EC2 provides scalable computing capacity in the Amazon Web Services which is suitable and cost-effective for our tasks. We also utilized the cluster GPU instances available on Amazon EC2 to accelerate the computation time needed for training the networks.

As mentioned earlier, character recognition on the notMNIST dataset is a more challenging task than the written digits recognition on the MNIST dataset. Therefore, the conventional methods that work well on MNIST may not work well on notMNIST. We have tested the Logistic Regression classifier [9] using the tools supported by scikit-learn version 0.17 [16]. We trained on 200k training samples

and received the accuracy of 85.11% on 10k test data. As we will show later, neural network architectures can achieve better performances on the same training and testing dataset. We also compare the predictive accuracy of the Multinomial Logistic Regression using Simple Gradient Descent [9]. As discussed in section 2.3.1, simple gradient descent is not scalable for a large dataset. The training time is irrational if we would load all the data into memory, thus we trained the network using a subset of 10k samples only. Validation set and test set are also 10k each. We run the algorithm for 3k iterations with a fixed step size/learning rate of 0.5. The test accuracy is 83.5%. This accuracy is worse than the conventional logistic regression classifier due to the fact that we trained the network on a smaller training set. Since Stochastic GD is designed to be faster than Simple GD, thus we can deploy it using the same Multinomial Logistic Regression but with a larger dataset. With a mini-batch size of 128, we increase the size of training samples to 200k. Validation and test set size are 10k each. We run the algorithm with 3k iterations, and with a constant step size/learning rate of 0.5. The test accuracy is slightly increased to 86.8% due to the fact that a bigger training data set results in better prediction.

We conducted five experiments to evaluate the performance of character classification using various shallow neural networks and well-constructed deep neural networks. In each experiment, 200k samples were used to train the network, and 10k samples were used for validation and testing. We run each algorithm with 30k iterations and with a constant step size/learning rate of 0.5

In the first experiment, a simple neural network of 1-hidden layer with ReLU and 1024 hidden nodes was trained. This is so called a shallow neural network since it contains only one hidden layer. Fig. 5 shows the detailed architecture, the input image X_i is arranged as a single column vector of [784×1], and passed to the hidden layer. The hidden layer generates an output which is given by equation (10) with the activation function defined in equation (11). The new set of weights W and biases b are used to compute the output scores Y_i as described in equation (2). The output scores are then converted to probabilities using *Softmax* defined in equation (3). With the current set of weights and biases, we measure the discrepancy (cross-entropy) between the true label for an input pattern and the output produced as defined in equation (4). Thus, the loss function $\mathcal{L}$ is computed by taking the average of cross-entropy over a set of labeled samples as stated equation (5). We then finally optimize $\mathcal{L}$ using Stochastic GD in equation (7). Using this architecture, the test accuracy is improved to 92.7%.

Fig. 6 shows both the mini-batch and validation accuracies of experiment 1 that demonstrates the potential over-fitting problem during the neural network training. The gap between the mini-batch and validation accuracy indicates the amount of overfitting. In practice, when the gap is getting bigger or the validation accuracy starts to decrease, it implies the overfitting, so one can early stop training the network [17]. In Fig. 6, the gap is getting bigger after 20~25k iterations that suggest the validation accuracy may decrease at some point after it. Therefore, we set the number of iteration to be 30k in our experiments.

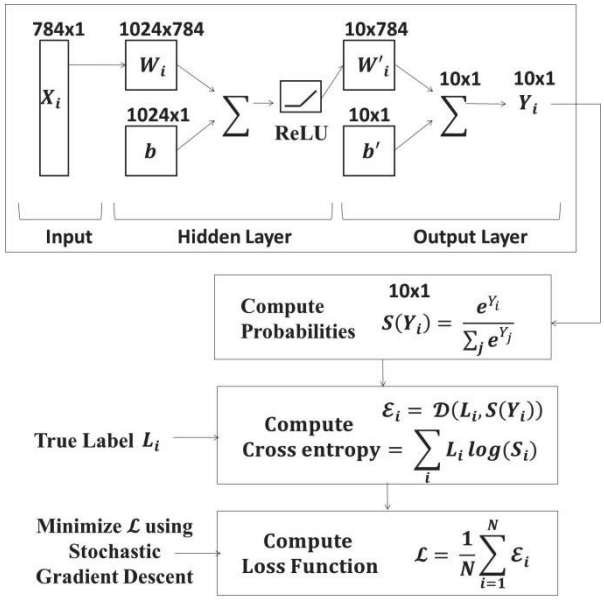


Fig. 5. The architecture used in the experiment #1: 1-hidden layer neural network with ReLU and 1024 hidden nodes.

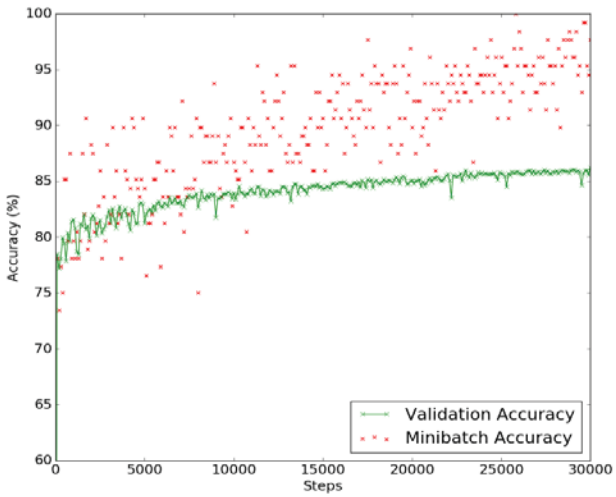


Fig. 6. The training accuracy of Experiment #1

In the second experiment, we added L2 regularization to the same model in Exp #1. With added L2 regulation, the accuracy is improved to 94.8%. In the third experiment, we added both L2 regularization and a 50% Dropout to the same model in Exp #1 to prevent overfitting. The test accuracy is 94.3% which is similar to Exp #2.

We then proposed to tune the performance of the shallow neural network model to a deep learning model by adding more hidden layers. In Exp #4, we use a multi-layer model

along with the learning rate decay as shown in Fig. 7. Using the learning rate decay is a good practice to lower the learning rate as the training progresses [18, 19]. In particular, this neural network contains 4 hidden layers with ReLU and different numbers of hidden nodes in each layer. Specifically, the number of hidden nodes is 2048, 1024, 1024, and 1024 in each layer respectively. L2 regularization and 50% Dropout were also used in this experiment. The test accuracy is improved to 96.5%.

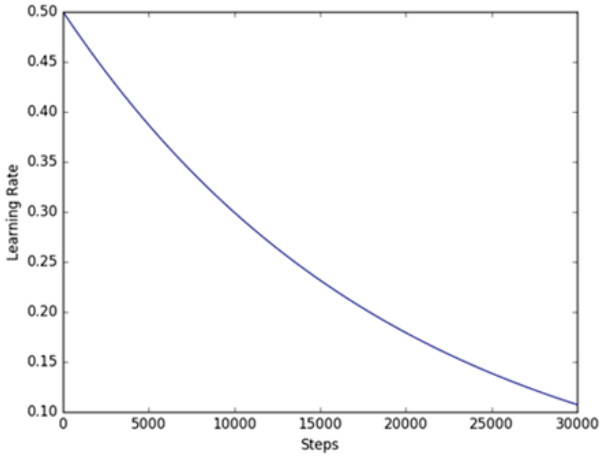


Fig. 7. The learning rate decay used in Experiment #4

In the fifth experiment, we proposed a five-layer convolutional neural network model to further improve the classification accuracy. The proposed CNN is shown in Fig. 8 and can be described as following: the first convolutional layer computes the output neurons that are connected to local regions in the input. The ReLU layer applies element wise activation function. The pool layer performs a down sampling operation along the spatial dimensions. The depths of the first, second, third, fourth and fifth convolutional layer are 16, 32, 64, 64 and 256, respectively.

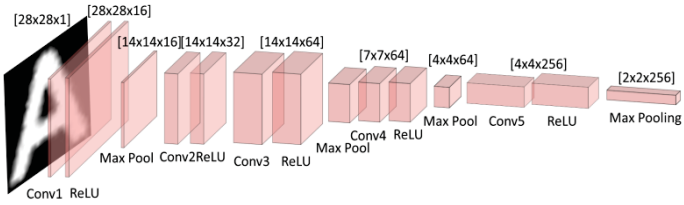


Fig. 8. The ConvNet architecture used in Experiment 5 and 6

The proposed architecture has 5 convolutional layers with a filter size 4×4, and ReLU layer in between; max pooling layers are inserted after ReLU layer for all layers except for the second convolutional layer. It is then connected to a Fully-Connected Layer of 2-Hidden Layers which have the size of 1024 and 205 nodes. Note that the first layer of the fully-connect part uses ReLU while the second hidden layer uses the ELU activation function. We also use a 50% Dropout, L2 Regularization, and a staircase learning rate decay shown in Fig. 9. The test accuracy for this experiment further improved to 97.5% after 30k iterations.

In Fig. 10, we show the validation accuracies from various neural networks (Exp #1 to #5) during the training process. The simple neural network of 1- hidden layer in Exp #1 offers the lowest validation accuracy. The validation accuracy improves as we employ L2 regularization and Dropout (Exp #2 and #3). In Exp #4, the deep learning neural network achieves better accuracy than the shallow neural network. Lastly, CNN in Exp #5 offers the best result.

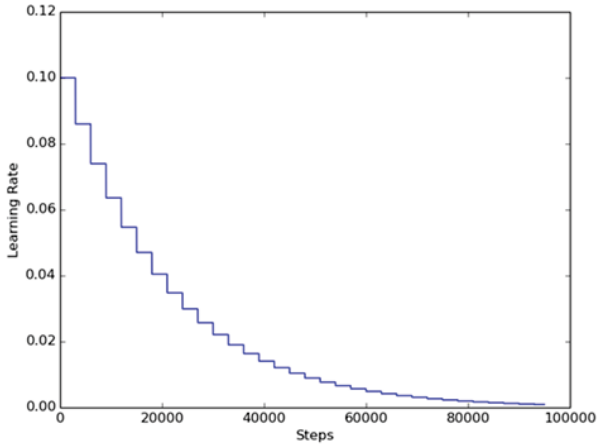


Fig. 9. The learning rate decay used in Experiment #5 and #6

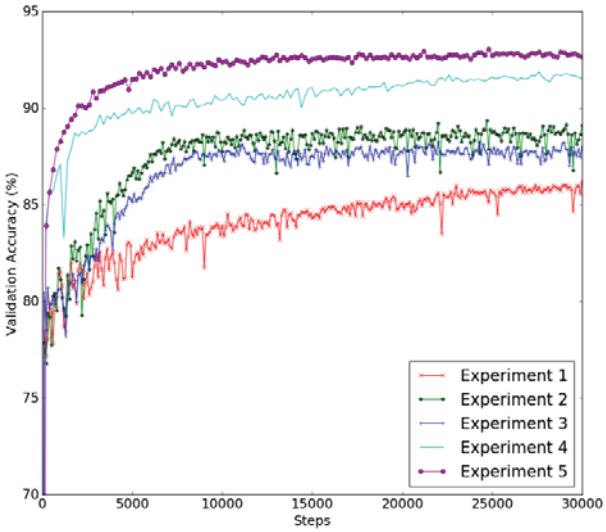


Fig. 10. The training accuracy of Experment #1 to #5

In addition, we performed another experiment similar to Exp #5 but increase the training size to 400k images. With this change, the test accuracy is further improved to 98%. The validation and test size was also increased to 15ks each and with 95k iterations. Table 1 summary the validation accuracies for the various neural networks from our study.

TABLE I. THE CLASSIFICATION ACCURACY ON THE NOTMINST DATASET

Exp #	Method	Accuracy [%]
1	1-hidden layer with ReLU (1024)	92.7
2	1-hidden layer with ReLU (1024) + L2	94.8
3	1-hidden layer with ReLU (1024) + L2 + 50% Dropout	94.3
4	4-hidden layers with ReLU (2048-1024-1024-1024) + L2 + 50% Dropout	96.5
5	CNN with 200k training set	97.5
6	CNN with 400k training set	98

5 Conclusions

In this paper, we conducted an empirical study for large-scale character recognition using the notMNIST dataset to evaluate the classification accuracy for various neural networks. We showed that deep learning architectures performed better than the conventional neural networks and proposed a five-layer convolutional neural network for this large-scale hard character classification task. The proposed network incorporates rectified linear units (ReLU) and exponential linear unit (ELU) the activation function, and employed L2 regularization and dropout techniques to prevent possible overfitting in training. The result shows that our method can achieve 98% classification accuracy.

All experiments with various architectures were implemented using Python with Tensorflow API. These networks were trained and validated on Amazon Elastic Compute Cloud and with GPU-accelerated utility to reduce the total training time. The proposed architecture and the implementation framework are scalable and cost-effective, and can be easily adapted to other image classification applications using deep neural networks.

6 References

[1] LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86, no. 11, pp. 2278-2324, 1998.

[2] Ciresan, Dan, Ueli Meier, and Jürgen Schmidhuber. "Multi-column deep neural networks for image classification." In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 3642-3649. IEEE, 2012.

[3] Ciresan, Dan Claudiu, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. "Deep, big, simple neural nets for handwritten digit recognition." Neural computation 22, no. 12, pp. 3207-3220, 2010.

[4] Deng, Li, and Dong Yu. "Deep learning: Methods and applications." Foundations and Trends in Signal Processing 7, no. 3–4, pp. 197-387, 2014.

[5] Deng, Li, Geoffrey Hinton, and Brian Kingsbury. "New types of deep neural network learning for speech recognition and related applications: An overview." In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pp. 8599-8603. IEEE, 2013.

[6] MNIST. http://yann.lecun.com/exdb/mnist/

[7] notMNIST. http://yaroslavvb.com/upload/notMNIST/

[8] Trier, Øivind Due, Anil K. Jain, and Torfinn Taxt. "Feature extraction methods for character recognition-a survey." Pattern recognition 29.4, pp. 641-662, 1996.

[9] Bishop, Christopher M. "Pattern Recognition and Machine Learning", Springer, pp. 115, 198, 205, 240, 236, 356, 497, 2006.

[10] Nair, Vinod, and Geoffrey E. Hinton. "Rectified linear units improve restricted boltzmann machines." In Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 807-814. 2010..

[11] Clevert, Djork-Arné, Thomas Unterthiner, and Sepp Hochreiter. "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)." arXiv preprint arXiv:1511.07289 (2015).

[12] Evgeniou, Theodoros, Massimiliano Pontil, and Tomaso Poggio. "Regularization networks and support vector machines." Advances in computational mathematics 13, no. 1, pp. 1-50, 2000.

[13] Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: A simple way to prevent neural networks from overfitting." The Journal of Machine Learning Research 15, no. 1, pp. 1929-1958, (2014).

[14] Abadi, Martın, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado et al. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems." arXiv preprint arXiv:1603.04467, 2016.

[15] Amazon Elastic Compute Cloud (Amazon EC2), http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using_cluster_computing.html

[16] Machine Learning in Python. http://scikit-learn.org/stable/

[17] Prechelt, Lutz. "Early stopping-but when?." Neural Networks: Tricks of the trade. Springer Berlin Heidelberg, , pp. 55-69, 1998.

[18] H. Robbins and S. Monro, "A stochastic approximation method," Annals of Mathematical Statistics, no. 22, pp. 400–407, Sep. 1951.

[19] Senior, Alan, Georg Heigold, Marc'Aurelio Ranzato, and Ke Yang. "An empirical study of learning rates in deep neural networks for speech recognition." In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pp. 6724-6728. IEEE, 2013.

Cognitively Realistic Problem Solving through Causal Learning

Seng-Beng Ho
Department of Social and Cognitive Computing
Institute of High Performance Computing, A*STAR
Singapore

Abstract - *Traditionally, search algorithms such as the A* algorithm have been used for general problem solving in AI, including looking for paths that circumvent obstacles. However, even with the assistance of (often hand-coded) heuristics, the search space is still often very large. The problem solving process is not cognitively realistic because natural intelligent systems such as human beings do not solve problems in this manner. Building on our previous work on effective causal learning, this paper demonstrates how an intelligent system can learn about the physical properties of objects in the environment through identifying the causes of, say obstruction, and use that knowledge to solve problems rapidly without having to resort to extensive and unintelligent search processes.*

Keywords: Problem solving; causal learning; rapid learning; cognitively realistic problem solving; intelligent agents; spatial movement

1 Introduction

The visual sense allows an intelligent system to learn about the inherent physical properties of objects as well as how they interact with each other or respond to causal entities such as forces. In our previous work [1] we formulated an effective causal learning framework that allows causalities such as those that could be present in the physical environment to be learned through visual observation. Causal rules that are learned using this method could be used for problem solving. Thus, this method could contribute to an intelligent system's problem solving process in two major ways. Firstly, the causal rules are learned from observing the environment, so if the environment changes, such as if the physical laws that govern physical entities and their attendant interactions change, the intelligent system would learn and encode a new set of causal rules for problem solving and would then formulate different solutions accordingly. This allows the intelligent system to be adaptive. Secondly, unlike traditional methods of learning such as reinforcement learning [2] and the various kinds of supervised and unsupervised learning [3], effective causal learning requires very few training examples. Hence the system can learn the causal rules involved rapidly and deploy them rapidly for problem solving.

Traditionally, a general method for problem solving in AI is search, such as the A* search [4]. In A* search, if the process is applied to solve a problem such as one that involves physical entities, typically some knowledge of the physical world is built-in (such as knowing that some objects are impenetrable and could impede the movement of other objects, thus functioning as an obstacle), and the search process then uses this knowledge to generate a search tree to find a solution to a problem. One example would be a physical movement problem in which an agent is supposed to move from a starting location to a goal location. The search process generates many possible paths, and if there are obstacles, the built-in physical knowledge would inhibit the generation of the search tree in certain directions accordingly. There is also the use of heuristics, which is the measure of how close a certain search node is to the goal. This helps to cut down the search – only the most promising parts of the search tree as measured by the heuristics are searched. However, the search process is not cognitively realistic as it generates a large amount of unnecessary search space, and no attempt is made to learn useful causal knowledge about the physical properties of the environment that may help to drastically reduce the problem solving effort.

In this paper, we describe a method that overcomes the shortcomings of the traditional search method and apply it to a spatial movement problem that involves obstacles. As mentioned earlier, firstly the causal rules governing the behavior of the physical entities in the environment are learned and used in the problem solving process. These causal rules are learned as a result of a small number of training instances. Secondly, using the causal rules, the problem solving process can obviate a large amount of unnecessary search efforts and produce a solution rapidly, and this is due to the fact that the causal rules encode powerful generalizations about the physical properties of entities in the environment. This computational problem solving process is not only rapid, but also cognitively realistic as it parallels the process used by intelligent systems such as human beings for solving the same problem. The learning and problem solving method is also general, much like A* is a general problem solving method that can be applied to a wide range of problems.

1.1 Comparisons between A* and Current Method

In the earlier work [1] we have demonstrated the application of the effective causal learning method to solve the spatial movement to a goal without obstacle (SMG) problem. That process demonstrated mainly the learning of domain specific heuristics to obviate a large amount of search effort. In the current spatial movement to a goal *with* obstacle (SMGO) problem, the emphasis is on learning the causal rules associated with the obstacle to allow the intelligent system to reason a way to overcome the obstacle.

Figure 1 shows a comparison between the problem solving process generated by the A* algorithm and one that is expected to be generated by our causal learning method for the SMGO problem. In Figure 1(a), it can be seen that the A* search process has to search a relatively wide area of space, most of which is actually irrelevant to the problem at hand. The process is hence relatively "blind." In Figure 1(b), we show the path that would be followed by the causal learning method that is also cognitively realistic. Basically, the *Agent* involved first heads toward the goal in a straight line (thinking that the earlier straight-line path learned in a SMG problem would work as it does not have any experience with obstacles yet), and then when it encounters an impediment to its movement – the *Obstacle* - it attempts a small amount of search/trial-and-error around the point of impediment to encode the cause of its impediment. Then it reasons out a method to counter the cause of impediment, and this knowledge is used by it to rapidly plan a path around the *Obstacle* as shown. Natural intelligent systems such as human beings would do likewise in a first encounter with an obstacle.

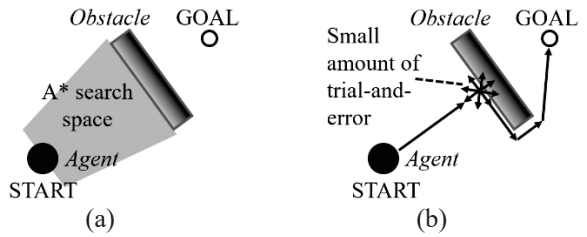


Figure 1: SMGO problem. (a) A* search space. (b) "Intelligent" path generated by causal learning.

In the case of Figure 1(b), the path shown around the *Obstacle* is what is expected in a *first* encounter with the *Obstacle*. Having learned the movement impediment property of the *Obstacle*, subsequently the *Agent* would just head straight toward one of the corners of the *Obstacle* from the START position.

2 General Problem Solving Framework

In this section we use an example to develop a general framework for causal learning and problem solving that will be applied to the SMGO problem in the ensuing discussion.

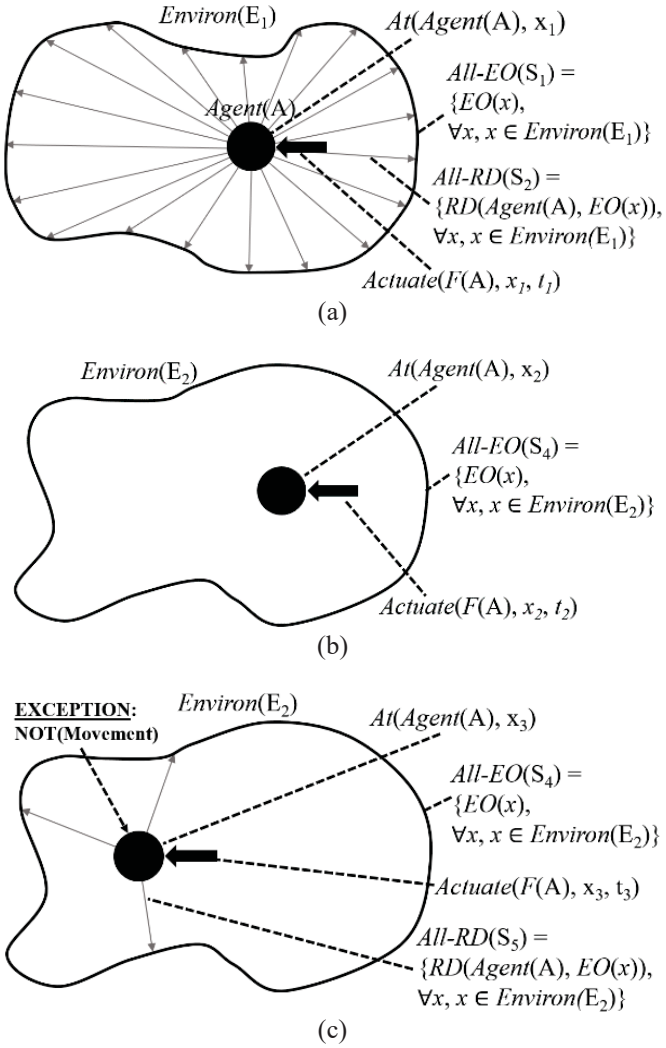


Figure 2: (a) An *Agent* actuates a force to move itself in *Environ*(E_1). (b) The *Agent* actuates a force in *Environ*(E_2). (c) The *Agent* actuates a force in *Environ*(E_2) at a different location but experiences no movement. "x" represents the absolute location and "t" the time of the action/event.

In Figure 2(a), we show an irregular boundary representing the "environment." There is an *Agent* in the environment. This environment is made up of a set of elemental objects (*EO*s). Suppose the *Agent* actuates a force, *F*(A), within itself to propel itself to generate a movement. The following causal rule would be learned through the method of effective causal learning as described in [1]:

$At(Agent(A), x_1)$ **[SYN]** $\wedge$
$\exists All\text{-}EO(S_1) = \{EO(x), \forall x\ x \in Environ(E_1)\}$ **[SYN]** $\wedge$
$\exists All\text{-}RD(S_2) = \{RD(Agent(A), EO(x)),$

$\forall x\ x \in Environ(E_1)\}$ **[SYN]** $\wedge$
Actuate(F(A), x_1, t_1) **[DIA]**
$\rightarrow$ *Move*(*Agent*(A), $t_1+\Delta$) (1)

In Rule 1, *Environ*(E_1) is the environment shown in Figure 2(a) and *RD* is the relative distance between the *Agent* and an *EO* in the environment. We are assuming that a visual system would provide this and other visual information to the system. SYN and DIA are the synchronic and diachronic causal preconditions respectively as discussed in [1]. Basically SYN is the static causal context that must be present before an effect can take place and DIA is the change of state of some variable that must be present before an effect can take place. At the first observation of a change of state of the *Agent*, after an elemental time Δ, in the form of movement at time $t_1+\Delta$ – *Move*(*Agent*(A), $t_1+\Delta$) – the system formulates causal Rule 1 by assuming that the three SYN conditions are necessary. Firstly, the *Agent* must be at the absolute location x_1 - *At*(*Agent*(A), x_1). (Note that x_1 is a 2D vector.) Secondly, the existence of *Environ*(E_1), represented by the set of all *EO*s in it, is necessary. S_1 stands for this set. Thirdly, the existence of all the *RD*s from the *Agent* to all the *EO*s of *Environ*(E1) is necessary. S_2 stands for this set of values. The DIA condition *Actuate*(F(A), x_1, t_1) is identified to be the diachronic causal precondition of the effect *Move*(*Agent*(A), $t_1+\Delta$), based on the procedures discussed in [1].

Note that in our universe, the physical law is such that the force law – the movement of an object as a result of a force application – is not dependent on the presence of and the distance to other objects in the environment, unless the object involved and other objects in the environment are, say, electrically charged. In our case here, the causal learning process will discover if these synchronic conditions initially thought necessary are indeed necessary.

Now, suppose the force is actuated again after the *Agent* has moved to location x_2. The following rule is formulated:

At(*Agent*(A), x_2) **[SYN]** $\wedge$
$\exists$*All-EO*(S_1) = {*EO*(x), $\forall x\ x \in Environ(E_1)\}$ **[SYN]** $\wedge$
$\exists$*All-RD*(S_3) = {*RD*(*Agent*(A), *EO*(x)),
$\forall x\ x \in Environ(E_1)\}$ **[SYN]** $\wedge$
Actuate(F(A), x_2, t_2) **[DIA]**
$\rightarrow$ *Move*(*Agent*(A), $t_2+\Delta$) (2)

There are three parameters that are different between Rules 1 and 2: $x_1 \neq x_2$, $S_2 \neq S_3$, and $t_1 \neq t_2$. This is because the *Agent* is now at a different location and the event takes place at a different time, and because the *Agent* is at a different location, the set of *RD*s to the *EO*s in *Environ*(E_1) is also different. The set of *EO*s, S_1, remains the same. A generalization can now be effected using Rules 1 and 2. Depending on the urgency of the situation, if an *Agent* is not desperate to derive a general rule to encode the phenomenon (in this case the force law) involved, it can wait for more instances of observation. On the other hand, if it is desperate, it can effect generalization with fewer instances. The derivation of the level of desperation in a given situation requires a separate computational module which we will not describe here. Here we assume that at least two instances are necessary to effect a generalization - we term this "dual instance generalization" [1]. Hence, given Rules 1 and 2, the following generalized rule is obtained:

At(*Agent*(A), x_{ANY}) **[SYN]** $\wedge$
$\exists$*All-EO*(S_1) = {*EO*(x), $\forall x\ x \in Environ(E_1)\}$ **[SYN]** $\wedge$
Actuate(F(A), x_{ANY}, t_{ANY}) **[DIA]**
$\rightarrow$ *Move*(*Agent*(A), $t_{ANY}+\Delta$) (3)

In Rule 3, the SYN condition associated with *RD* has been removed because $S_2 \neq S_3$ means that the *RD* condition is not necessary. In principle, the *At*(*Agent*(A), x_{ANY}) condition should also be removed as $x_1 \neq x_2$, which implies that it does not matter where the *Agent* is located. However, in order to have an explicit statement about the *Agent*'s location, we retain that term and instead use *At*(*Agent*(A), x_{ANY}) to indicate the fact that the *Agent* can be anywhere for this rule to hold. The x_{ANY} and t_{ANY} parameters in the DIA condition indicate that this "force law" is applicable anywhere and anytime in *Environ*(E_1).

Suppose now the entire physical "experiment" (actuating the force twice on the *Agent*) takes place in a different environment - *Environ*(E_2) – as shown in Figure 2(b). A general rule similar to that of Rule 3 would be formulated as follows:

At(*Agent*(A), x_{ANY}) **[SYN]** $\wedge$
$\exists$*All-EO*(S_4) = {*EO*(x), $\forall x\ x \in Environ(E_2)\}$ **[SYN]** $\wedge$
Actuate(F(A), x_{ANY}, t_{ANY}) **[DIA]**
$\rightarrow$ *Move*(*Agent*(A), $t_{ANY}+\Delta$) (4)

Now the set of *EO*s - S_4 – is different from the earlier S_1 because the environment is different. Combining Rules 3 and 4 a further generalized rule is obtained:

At(*Agent*(A), x_{ANY}) **[SYN]** $\wedge$
Actuate(F(A), x_{ANY}, t_{ANY}) **[DIA]**
$\rightarrow$ *Move*(*Agent*(A), $t_{ANY}+\Delta$) (5)

The general rule is applicable anywhere, anytime, and in any environment. This is like the force law as we know it in our physical reality (provided that the *Agent* is not, say, electrically charged and the environment does not contain any charged objects).

Now, suppose at a specific location, x_3, the force law of Rule 5 does not hold, which is that the actuation of a force does not cause the *Agent* to move, as shown in Figure 2(c). An exceptional condition will then be added to Rule 5:

At(*Agent*(A), x_{ANY}) **[SYN]** $\wedge$
Actuate(F(X), x_{ANY}, t_{ANY}) **[DIA]**
$\rightarrow$ *Move*(*Agent*(A), $t_{ANY}+\Delta$)
EXCEPT

At(*Agent*(A), x_3) **[SYN]** ∧
∃*All-EO*(S_4) = {*EO*(*x*), ∀*x* *x* ∈ *Environ*(E_2)} **[SYN]** ∧
∃*All-RD*(S_5) = {*RD*(*Agent*(A), *EO*(*x*)),
∀*x* *x* ∈ *Environ*(E_2)} **[SYN]** ∧
Actuate(*F*(A), x_3, t_3) **[DIA]**
→ *NOT*(*Move*(*Agent*(A), t_3+Δ)) (6)

2.1 Inductive Competition for Rule Generalization

A seemingly conflicting situation can arise as a result of applying the generalization process to the NOT(Movement) situation. Suppose now there is yet another location, say x_4, at which the NOT(Movement) of the *Agent* is obtained when a force is actuated on it. Using the dual instance generalization process described above, a general rule could be obtained that states that an actuation of a force anywhere, anytime, and in any environment will lead to NOT(Movement) of the *Agent*, which is in conflict with the movement rule, Rule 5.

The process of generalization is an inductive process. Rules can change when new observations are made. And how we formulate the rules is also dependent on how we plan to use them. If there are not many situations under which Not(Movement) occurs, then when the situation of the *Agent* being at x_4 mentioned above occurs, it can be added to Rule 6 as yet another exception. However, if more and more locations are found at which Not(Movement) is true, then it will be better to state that it is generally true that NOT(Movement) holds anywhere, anytime, and in any environment except at a few locations movement holds. We term this "inductive competition for rule generalization."

Suppose in a planning process, the system wishes to plan a path across the environment, then the way the rule is formulated based on observations so far has an impact on the planning process. Suppose currently Rule 6 is the best characterization of the situation. Then the planning process can assume that the *Agent* would be relatively unimpeded when it moves across the environment, and at a few of the locations when this is expected not to be true, the planning process will formulate the plan accordingly. On the other hand, suppose the Not(Movement) version of Rule 6 is true, then the planning process would have to assume that at most locations the *Agent* will be impeded.

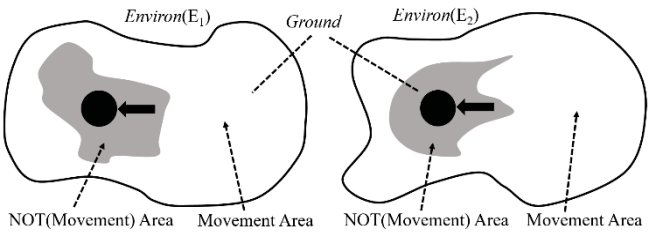


Figure 3: Ground colors of environment provide extra synchronic (SYN) causal conditions.

The tussle between the movement and NOT(Movement) situations for rule formulation could be resolved if additional information in the environment is available. Figure 3 shows that there are colors of the "patches" on the "ground" of the environment that the visual system can provide that can assist with the formulation of causal rules. Suppose a few instances of physical experiments are performed each on the WHITE and GRAY areas of the environment, and it is discovered that when the *Agent* is on the WHITE area, the movement rule is obtained and when the *Agent* is on the GRAY area, NOT(Movement) is obtained, then a set of rules as follows can be formulated using the extra SYN conditions provided by the ground color:

At(*Agent*(A), x_{ANY}) **[SYN]** ∧
Color(*Ground*(x_{ANY}), WHITE) **[SYN]**
Actuate(*F*(A), x_{ANY}, t_{ANY}) **[DIA]**
→ *Move*(*Agent*(A), t_{ANY}+Δ) (7)

At(*Agent*(A), x_{ANY}) **[SYN]** ∧
Color(*Ground*(x_{ANY}), GRAY) **[SYN]**
Actuate(*F*(A), x_{ANY}, t_{ANY}) **[DIA]**
→ *NOT*(*Move*(*Agent*(A), t_{ANY}+Δ)) (8)

3 Application to the SMGO Problem

The spatial movement to a goal (SMGO) problem is illustrated in Figure 4(a). Basically, an *Agent*, starting from a START location, is to move to a GOAL location, and in between these two locations, there is an *Obstacle*. In this section, we apply the general framework developed in the previous section as regards causal learning to the problem.

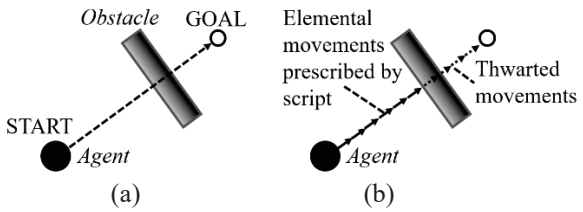


Figure 4: (a) SMGO problem. (b) An attempted solution that results in the thwarting of a plan based on a script.

Firstly, we assume that the *Agent*/system had earlier solved the SMG (spatial movement to a goal without obstacle) problem as discussed in [1] and learned and encoded the solution in a form of a "script." [5] This script basically prescribes a straight-line movement from START to GOAL. Suppose now the *Agent* encounters the SMGO problem the first time as shown in Figure 4(b). Given the START and GOAL locations, but without the knowledge that the *Obstacle* is an impediment, the *Agent*/system would retrieve the solution to the SMG problem and execute it to attempt to get to the GOAL to solve the problem. However, in the process of executing the solution, the *Agent* encounters a NOT(Movement) situation when it contacts the *Obstacle*.

Suppose the *Obstacle* has an identifiable color, GRAY, supplied by the visual system, the following causal rule is formulated:

$$\begin{aligned}
&At(Agent(\mathrm{A}), x_{11})\ \textbf{[SYN]} \wedge \\
&\exists All\text{-}EO(S_{11}) = \{EO(x), \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge \\
&\exists All\text{-}RD(S_{12}) = \{RD(Agent(\mathrm{A}), EO(x)), \\
&\quad \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge \\
&Contact(Agent(\mathrm{A}), Obstacle(x_{11}+\delta))\ \textbf{[SYN]} \wedge \\
&Color(Obstacle(x_{11}+\delta), \mathrm{GRAY})\ \textbf{[SYN]} \wedge \\
&\quad Actuate(F(\mathrm{A}, RA{=}0), x_{11}, t_1)\ \textbf{[DIA]} \\
&\quad\quad \rightarrow NOT(Move(Agent(\mathrm{A}), RA{=}0, t_1 + \Delta)) \qquad (9)
\end{aligned}$$

This is similar to the formulation of Rule 8 using the color information of the ground. (Here, it is the color of the *Obstacle*, which is GRAY, that is the SYN condition.) Rule 9 is basically a logical description of the thwarting of the attempt (impediment of the movement) to reach the GOAL using the straight-line script. x_{11} is the location of the *Agent*, represented by its center. Because of the finite size of the *Agent*, the point that it contacts the *Obstacle* is slightly further away, at $x_{11}+\delta$. Similar to the environment, say *Environ*(E_1) in the earlier example, which provides two SYN conditions to the causal rule involved (say, equation (1)), here it is the *Obstacle* that provides the sets of *EO*s and the *RD*s as synchronic causal conditions to the *Agent*'s movement impediment. We use S_{11} and S_{12} to represent these two sets of parameters respectively. In contrast to the parameters associated with the force, *F*(A), in the earlier discussion, here there is an additional parameter associated with *F*, which is the relative angle (*RA*) subtended by the direction of the force and a line joining the center of the *Agent* and the GOAL. *RA*=0 means that the force is applied directly toward the GOAL. This parameter and its associated value are derived from the script of the solution of the SMG problem [1, 5] that stipulates a series of forces pointed in the direction of the GOAL as the solution. The NOT(Movement) consequence is also in the direction of the GOAL.

The DIA part of Rule 9 specifies that the actuation of a force on the *Agent* in the direction of the GOAL leads to a NOT(Movement) consequence. In order to achieve movement, we need to negate the NOT(Movement). We consider negating the SYN conditions to achieve this:

$$\begin{aligned}
&NOT(At(Agent(\mathrm{A}), x_{11})) \vee \\
&NOT(\exists All\text{-}EO(S_{11}) = \{EO(x), \forall x\ x \in Obstacle(O_1)\}) \vee \\
&NOT(\exists All\text{-}RD(S_{12}) = \{RD(Agent(\mathrm{A}), EO(x)), \\
&\quad \forall x\ x \in Obstacle(O_1)\}) \vee \\
&NOT(Contact(Agent(\mathrm{A}), Obstacle(x_{11}+\delta))) \vee \\
&NOT(Color(Obstacle(x_{11}+\delta), \mathrm{GRAY})) \qquad (10)
\end{aligned}$$

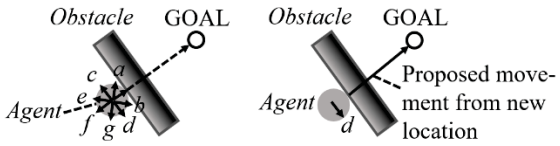
In words, these mean:

(i) Cause the *Agent* not to be at x_{11}.
(ii) Cause the *Obstacle* not to exist.
(iii) Cause the *RD*s from *Agent* to all the *EO*s on the *Obstacle* to be of a different set.
(iv) Cause "not-contact" between *Agent* and the *Obstacle* at $x_{11}+\delta$.
(v) Cause the color of the *Obstacle* to be "not-GRAY" at the point of contact $x_{11}+\delta$.

If (ii) could be achieved, the problem is solved right away. We will assume that this is not an option for the *Agent* – there are no known means that the *Agent* can dematerialize the *Obstacle*. (i), (iii), and (iv) could be achieved by changing the location of the *Agent*. (iii) and (iv) could also be achieved by moving the *Obstacle* - again, we assume this is not an option available for the *Agent*. We also assume that there is no ability on the *Agent*'s part to change the color of the *Obstacle* – option (v) – at the point of contact. So, what remains would be to change the location of the *Agent*. This process is carried out automatically computationally by checking the causal rule base to see whether the corresponding rules that can effect the various conditions (i) – (v) are available.

Note that any of the possible actions above is not guaranteed to work. For example, even if one can dematerialize the *Obstacle*, there could be another invisible force that impedes the movement of the *Agent*. So, even if the *Agent*'s location were to change, as recommended by option (i), there is no guarantee that it will work. However, it is still a good option to try.

The *Agent* now carries out a small number of physical experiments – a small blind search process – to determine the consequences of changing the location of the *Agent*, using the known means: the actuation of a force on the *Agent*. This force is applied in all possible directions to determine the various consequences of location change so that a good one could be selected. This process is not only applicable to the spatial movement scenario – in general, when a process is thwarted in the same manner as in the current case of spatial movement, all available actions will be tried in a small blind search process. Similarly, other non-spatial movement scenarios should also execute the ensuing process described for the current spatial movement scenario.



(a) (b)

Figure 5: (a) Various directions of force actuation to derive the consequences of moving the *Agent* in these directions. (b) Direction *d* is chosen and another straight-line movement toward the GOAL is attempted from the new location.

Figure 5(a) shows the various directions of forces actuated on the *Agent* to move the *Agent* elementally in various directions. It is discovered that in certain directions, the *Agent* continues to be impeded, while in some other

directions, the *Agent* is free to move. This is learned and encoded in the following rules:

$$\begin{aligned}&At(Agent(\mathrm{A}), x_{11})\ \textbf{[SYN]} \wedge\\ &\exists All\text{-}EO(S_{11}) = \{EO(x), \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge\\ &\exists All\text{-}RD(S_{12}) = \{RD(Agent(\mathrm{A}), EO(x)),\\ &\quad \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge\\ &Contact(Agent(\mathrm{A}), Obstacle(x_{11}+\delta))\ \textbf{[SYN]} \wedge\\ &Color(Obstacle(x_{11}+\delta), \mathrm{GRAY})\ \textbf{[SYN]} \wedge\\ &\quad Actuate(F(\mathrm{A}, 270° < RA < 90°), x_{11}, t_1)\ \textbf{[DIA]}\\ &\quad \rightarrow NOT(Move(Agent(\mathrm{A}), RA{=}0, t_1 + \Delta)) \qquad (11)\end{aligned}$$

$$\begin{aligned}&At(Agent(\mathrm{A}), x_{11})\ \textbf{[SYN]} \wedge\\ &\exists All\text{-}EO(S_{11}) = \{EO(x), \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge\\ &\exists All\text{-}RD(S_{12}) = \{RD(Agent(\mathrm{A}), EO(x)),\\ &\quad \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge\\ &Contact(Agent(\mathrm{A}), Obstacle(x_{11}+\delta))\ \textbf{[SYN]} \wedge\\ &Color(Obstacle(x_{11}+\delta), \mathrm{GRAY})\ \textbf{[SYN]} \wedge\\ &\quad Actuate(F(\mathrm{A}, 90° \leq RA \leq 270°), x_{11}, t_1)\ \textbf{[DIA]}\\ &\quad \rightarrow Move(Agent(\mathrm{A}), RA{=}0, t_1 + \Delta) \qquad (12)\end{aligned}$$

From these rules, it can be seen that the force actuation for the directions between *c* and *d*, and toward the GOAL (e.g., directions *a* and *b*, corresponding to $270° < RA < 90°$), will lead to NOT(Movement) of the *Agent*, and the directions between c and d, and away from the GOAL (e.g., directions *e*, *f*, and *g*, corresponding to $90° \leq RA \leq 270°$), will lead to movement. And among the directions in the latter case, directions *c* or *d* (depending on the point of contact between the *Agent* and the *Obstacle*) will give rise to a minimum distance to the GOAL. (For scenarios other than spatial movement, there are equivalent measures of "minimum distance" to the GOAL.) We apply the usual minimum distance to goal heuristic (often also used in A*) and select, say, *d* as the direction of force actuation to change the location of the *Agent*. This results in the situation shown in Figure 5(b) – the *Agent* is moved elementally in the direction *d* – and then another attempt is made to apply the straight-line movement script to the *Agent* to move it toward the GOAL to see if now the problem (of reaching the GOAL) can be solved.

The *Agent* will discover that its action of moving in direction *d* will again result in an impediment of movement in the direction of the GOAL much like before. However, this effort is not completely a waste as the *Agent* now learns a more general causal rule. Because in this second instance of impediment, the *Agent* is at a different location and the *RD*s to the *EO*s on the *Obstacle* are hence also different, Rule 9 is generalized to the rule below (the "$\exists All\text{-}RD(S_{12})$" line is removed and x_{11} is generalized to x_{ANY}):

$$\begin{aligned}&At(Agent(\mathrm{A}), x_{ANY})\ \textbf{[SYN]} \wedge\\ &\exists All\text{-}EO(S_{11}) = \{EO(x), \forall x\ x \in Obstacle(O_1)\}\ \textbf{[SYN]} \wedge\\ &Contact(Agent(\mathrm{A}), Obstacle(x_{ANY}+\delta))\ \textbf{[SYN]} \wedge\\ &Color(Obstacle(x_{ANY}+\delta), \mathrm{GRAY})\ \textbf{[SYN]} \wedge\\ &\quad Actuate(F(\mathrm{A}, RA{=}0), x_{ANY}, t_1)\ \textbf{[DIA]}\\ &\quad \rightarrow NOT(Move(Agent(\mathrm{A}), RA{=}0, t_{ANY} + \Delta)) \qquad (13)\end{aligned}$$

The corresponding negated SYN conditions are:

$$\begin{aligned}&NOT(At(Agent(\mathrm{A}), x_{ANY})) \vee\\ &NOT(\exists All\text{-}EO(S_{11}) = \{EO(x), \forall x\ x \in Obstacle(O_1)\}) \vee\\ &NOT(Contact(Agent(\mathrm{A}), Obstacle(x_{ANY}+\delta))) \vee\\ &NOT(Color(Obstacle(x_{ANY}+\delta), \mathrm{GRAY})) \qquad (14)\end{aligned}$$

The first condition means to bring about a situation in which the *Agent* is not anywhere. This is not a viable option. The second condition means to dematerialize the *Obstacle* and again we assume this is not an available option. Again, we assume there is no means for the *Agent* to change the color of the *Obstacle*. Therefore, what remains is to execute the third condition, which is for the *Agent* not to contact the *Obstacle* anywhere.

There are many ways to achieve a non-contact situation with the *Obstacle*. One way is for the *Agent* to stay next to the *Obstacle* and move along the length of it until the *Agent* moves slightly beyond one of the "corners" and arrives at a location at which there is an unobstructed path toward the GOAL. The other is simply to move away from the *Obstacle*. However, if we impose the minimum distance heuristic earlier as a result of which we picked the *d* direction for optimal movement, then it makes sense that for every step of a proposed long sequence of actions to achieve an unobstructed path to the GOAL, direction *d* is selected (adding a number of minimum distances will lead to an overall minimum distance).

At this point after the *Agent* has moved elementally in the *d* direction, similar exploratory elemental force actuations that led to Rules 11 and 12 can be effected, and more general forms of Rules 11 and 12, which apply to any location along the *Obstacle*, would be formulated.

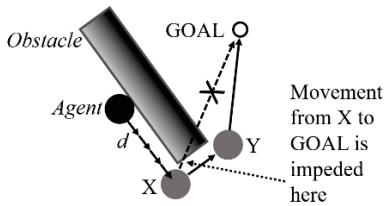


Figure 6: The repeated mental simulation of earlier learned elemental solution that leads to a complete solution.

Figure 6 shows that the *Agent*/system carries out a series of computational mental simulations, selecting the optimal direction of movement, *d*, at every step and moving the *Agent* until it moves "beyond" the "right corner", at which location (labeled X in the figure) the *Agent* no longer contacts the *Obstacle*. At every step of the simulation, there is no need to carry out any further physical experiments to learn about which directions of movement would give rise to obstruction, as the generalized versions of Rules 11 and 12 (that specify that at any location along the *Obstacle*, there are similar obstructions in the direction of the GOAL) provide that

information and imply that the optimal direction is *d* at every location next to and along the length of the *Obstacle*.

There are two situations that can obtain after this first stage mental simulation that generates a straight-line path along the *Obstacle* as the first steps of movement toward the GOAL. Suppose the *Obstacle* has some thickness as shown in Figure 6 and the *Agent* has knowledge about it. After the mental simulation brings the *Agent* to location X, it then attempts to move the *Agent* in the direction of the GOAL and the effort fails because the thickness impedes the movement of the *Agent* in the same manner as before. Then, another stage of mental simulation similar to that executed earlier would bring the *Agent* to location Y, at which point the *Agent* can move straight to the GOAL unobstructed. This entire path – move to X, then to Y, and then to the GOAL is returned as a solution to be executed physically. Another situation is that the *Agent* does not have the knowledge of the thickness of the *Obstacle* from its earlier locations (because its view is obstructed). The straight line movement leading to location X is then executed physically after the first stage mental simulation, at which point the *Agent* can "see" the thickness of the *Obstacle* and it then executes a second stage of mental simulation to bring it to location Y. Now, executing this straight-line movement to Y physically would then allow it to move straight to the GOAL unobstructed after that.

The above mental simulation steps have been encoded in a general problem solving algorithm that can be applied to general situations.

4 Conclusions

We have successfully shown that the causal learning process as described in [1] can be applied to the SMGO problem to obtain a cognitively realistic solution to the problem, in contrast with the traditional AI approach of using the A* algorithm that leads to blind and unintelligent extensive search effort. Causal learning, like A*, is a general method for problem solving.

This research provides not only a computer implementable process for AI but also a framework and model for psychologists to investigate problem solving processes in natural intelligent systems such as human beings.

5 Acknowledgement

This research is supported by the Social Technologies+ Programme funded by A*STAR Joint Council Office.

6 References

[1] S.-B. Ho. "On effective causal learning"; Proceedings of the 7th International Conference on Artificial General Intelligence, Springer-Verlag, 43-52, 2014.

[2] R. S. Sutton and A. G. Barto. "Reinforcement Learning: An Introduction". MIT Press, 1998.

[3] C. M. Bishop. "Pattern Recognition and Machine Learning". Springer, 2007.

[4] P. E. Hart, N. J. Nilsson, and B. Raphael. "A formal basis for the heuristic determination of minimum cost paths"; IEEE Transactions on Systems Science and Cybernetics SSC4, 4(2),100-107, 1968.

[5] S.-B. Ho and F. Liausvia. "Rapid learning and problem solving"; Proceedings of the IEEE Symposium Series on Computational Intelligence for Human-like Intelligence, IEEE press, 110-117, 2014.

Comparative studies in methods of feature recognition with machine learning for affective computing: A survey

Deok Hee Nam
Engineering and Computing Science, Wilberforce University, Wilberforce, OHIO, USA

Abstract – *A survey study about the various methods of feature recognition with machine learning for affective computing is examined. In order to explore the methods of feature recognition with machine learning methods, Sequential Floating Forward Selection (SFFS), Minimum Redundancy – Maximum Relevance (mRMR), Information Gain(IG), and Fisher projection (FP) are discussed. As the machine learning methods, k-Nearest Neighbor (kNN), Support Vector Machine (SVM), and Multilayer Perceptron (MLP) are described. Then, the various feature recognition methods with machine learning methods are compared by the statistical analyses using the classification accuracy performance with applying the discrete emotion data.*

Keywords: affective computing, cognitive computing, feature recognition, feature reduction, machine learning

1 Introduction

The research about the automated emotion recognition methods is a very important issue for affective computing and other recognition studies. Picard [1] introduced the study of the affective computing toward machine emotional intelligence with the analysis of affective physiological state. Affective computing is an important study for the development of systems and devices that can recognize, interpret, process, and simulate human affects. In general, the emotion is always interacting with various thinking and affected by the intelligent functioning. LeDoux [2] showed an experimental study using a rat to examine the emotional-oriented processing. Moreover, the brain before stimulated by the incoming signals can be functioning based upon processing the emotion. Hence, the recognition of the emotional senses can be detected by performing pattern recognition of the data based upon the act or emotional senses. In order to recognize the pattern recognition more efficiently, various techniques can be used for recognizing features with intelligent methodologies. Therefore, feature reduction methods and machine learning methods are very useful to develop the study of affective computing. In fact, affective computing fields can be spanning computer science, psychology, and cognitive science that relates to, arises from, or deliberately influences emotion or other affective phenomena.

2 Purpose of Research

The emotion can be formed by the fundamentals of human experiences such as influencing cognition, perception, learning, communication, rational decision-making and etc. Since the range of means and modalities of emotion expression is very wide and various because of the inaccessible factors such as blood chemistry, brain activity, neuro-chemicals, and etc.. In order to differentiate the unclassified or unknown factors to evaluate the emotional senses more efficiently, it is very helpful if an automated intelligent method can determine or classify the fuzzy information of the emotional sense data into the crispy and discrete groups or clusters. In addition, among various application fields of studying the classification of the emotional senses, Human-Computer Interaction (HCI) is a very important field to apply and recognize a personalized adaptive emotion to evaluate the emotional response characteristics of individual person and enhance the accuracy of its estimation using appropriate methods.

3 Methods of Research

Through this research, the paper shows a survey of preferable methods with feature reduction and machine learning methods for estimating emotion based on physiological signals. The methods for the feature reduction can be divided into feature selection and feature transformation. For the feature selection, Sequential Floating Forward Selection (SFFS), Minimum Redundancy – Maximum Relevance (mRMR), and Information Gain (IG) are discussed. For the feature transformation, Fisher Projection (FP) is described. For the Machine Learning Methods, k-Nearest Neighbor (kNN), Support Vector Machine (SVM), and Multilayer Perceptron (MLP) are explored.

3.1 Sequential Floating Forward Selection (SFFS)

Sequential Floating Forward Selection (SFFS) was introduced by Pudil et al. [3] and is a widely used methodology in the field of the feature selection. The SFFS consists of a forward (insertion) step and a conditional backward (deletion) step that partially avoids the local optima of the correct classification rate (CCR) [4] with applying the basic Sequential Forward Selection (SFS) procedure starting from the current feature set followed by the successive conditional exclusion of the worst feature in the newly updated set. The procedure of SFFS is following [5]:

Suppose k, the set of features, is determined. Then, apply one step of SFS algorithm followed by the increment of the next features to find the least significant feature in the set. If the current feature is the least significant feature in the set, then, repeat the procedure of SFS. But if the current feature is not the least significant feature, exclude the feature between the previous feature and the current feature. Then, in order to continue to find the least significant feature in the updated set, examine the updated subset and repeat the procedure if the subset is not meeting the conditions.

3.2 Minimum Redundancy Maximum Relevance (mRMR)

The minimum Redundancy-Maximum Relevance (mRMR) approach [6][7] is to select a feature subset, each of which has the maximal relevance with target class and the minimal redundancy with other features based on recognizing that the combinations of individually good variables do not necessarily lead to good classification/prediction performance. The feature subset can be obtained by calculating the mutual information between the features themselves and between the features and the class variables. In other words, to maximize the joint dependency of top ranking variables on the target variable, the redundancy of the feature subset must be reduced, which suggests incrementally selecting the maximally relevant variables while avoiding the redundant ones. The following procedure summarizes mRMR steps. First, the mutual information (MI) between two features x and y is evaluated. Then, the average mutual information $MI(x, l)$ between classification variable l and feature x is also calculated. To calculate the potential value of mRMR and achieve feature subset, the two conditions should be optimized, subtract the redundancy by calculating the minimum redundancy condition to minimize the total redundancy of all features selected from the relevance by calculating the maximum relevance condition is to maximize the total relevance between all features in S and classification variable.

3.3 Information Gain (IG)

Information gain is based on information theory [8] and a univariate method that selects features on the basis of the information contribution related to the class variable without considering feature interactions [13] and works well with texts [10]. It measures the expected reduction in entropy of class before and after observing the features. The information gain of term t is defined by the i^{th} category, the probability of the i^{th} category, and the conditional probabilities that the term t appears or not in the documents. The information gain indicates the amount of additional information about the features which measured as the entropy using the prior probability for the i^{th} discretized value of the feature.

3.4 Fisher Projection

The Fischer projection was introduced by Hermann Emil Fischer in 1891 [11], and originally proposed for the representation of carbohydrates. Hence, it is used by chemists, particularly in organic chemistry and biochemistry in order to display a 2-D representation of a 3-D organic molecule by the projection. The use of Fischer projections [12] in non-carbohydrates is discouraged; as such drawings are ambiguous when confused with other types of drawing. The Fischer Projection consists of both horizontal and vertical lines, where the horizontal lines represent the atoms that are pointed toward the viewer while the vertical line represents atoms that are pointed away from the viewer. The point of intersection between the horizontal and vertical lines represents the central element. First convert Dashed- Wedged Line Structure to a "flat" Dashed-Wedged Line Structure to apply the Fischer Projection. The horizontal line represents atoms that are coming out to the front side and the vertical line represents atoms that are going into the backside. The cross image to the right of the arrow is a Fischer projection.

3.5 *k*-Nearest Neighbor (*k*-NN)

The k-Nearest Neighbors (k-NN or kNN) is a simple classifier which makes use of instance based learning to classify samples [14][15]. The training data is stored and when there is a query on a new sample, the unclassified data is compared with the training data to determine the k-nearest neighbors. The type of the majority of these k-nearest neighbors is used to predict the class of the new sample. Even though k-NN is simple to execute, the run-time performance is

heavily dependent on the amount of training data. The value of k is decided based on the performance of the classifier for a range of k values using 10-fold cross validation method.

3.6 Support Vector Machine (SVM)

The SVM classifier uses input features from two classes to determine a maximum margin hyperplane that separates the two classes [16][17]. If the features are not linearly separable, special kernel functions can be used to transform the data to a higher dimensional feature space. Some commonly used kernel functions are linear, polynomial, Radial Basis Function (RBF), and sigmoid kernel functions. Typically, one or more of these kernel functions are applied to the data, and function resulting in the highest accuracy for the least amount of features is selected.

3.7 Multilayer Perceptron (MLP)

A multilayer perceptron (MLP) [18] is a feedforward artificial neural network model arranged in layers with forward connections in subsequence layers as a supervised network. The feedforward connections have weights and map the input data onto the appropriate outputs. A MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. A MLP utilizes a supervised learning technique called backpropagation for training the network. A MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable. A graphical representation of an MLP is shown below:

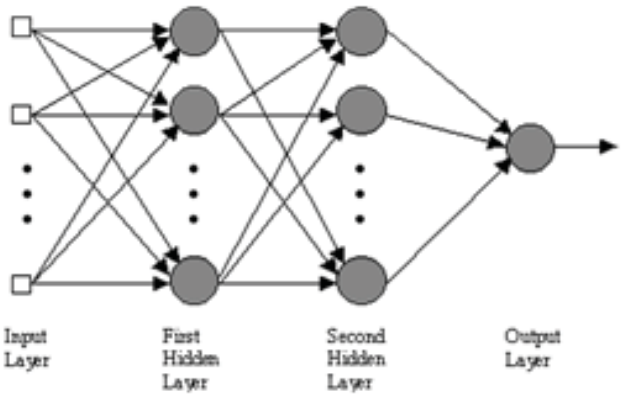


Fig. 1 The structure of the neural network layers for multilayer perceptron

The first layer is the input layer, and the inputs can be distributed into the subsequence layer as the first hidden layer followed by the second hidden layer. Each input sums its inputs, adds a bias or threshold term to the sum and nonlinearly transforms the sum to produce an output. The output layer collects all outputs from the second hidden layer for the outputs. Mellit et al. [18] shows the steps of the learning process in the multilayer perception with given a finite input pattern, $\{x_i(k)\} \in R$ where $1 \leq k \leq K$.

Step 1: Select the total number of layers, the number of the neurons in each hidden layer, and an error tolerance parameter.
Step 2: Select the initial value of the weight vectors.
Step 3: Initialize the weight vectors.
Step 4: Calculate the neural outputs.
Step 5: Calculate the output error.
Step 6: Calculate the output delta values.
Step 7: Calculate the propagation errors of the hidden neurons from the subsequence layers.
Step 8: Calculate the hidden delta values.
Step 9: Update weight vector.
Step 10: Calculate the error function.
Step 11: If k = K, go to Step 12, otherwise increment k by 1 and go to Step 4.
Step 12: If the error function is less than or equal to the error tolerance parameter, go to Step 13. Otherwise, go to Step 3.
Step 13: Learning is completed and output the weights.

4 Analyses and Results

Using the examined methodologies of feature reduction and machine learning methods, Kukolja et al.[9] shows the comparison of the methodologies using the sample data of the discrete emotion such as 89 samples of sadness, 99 samples of disgust, 38 samples of fear, 78 samples of happiness and 226 samples of other categories which are not suitable for the analysis of the classification of discrete emotions. First of all, the comparison of the classification accuracy using different feature reduction and machine learning methods shows the performance [9] as following on TABLE 1.

Based upon TABLE 1, using the combination of SFFS and machine learning methods, the best classification performance was 60.30 % using SFFS and MLP. Using the combination of mRMR and machine learning methods, the best classification performance was 57.61 % using mRMR and MLP. Using the combination of IG and machine learning methods, the best classification performance was 44.47 % using IG and MLP. Hence, MLP shows the best classification performance without using FP in the feature reduction methods.

TABLE 1 The comparison of feature reduction methods and machine learning methods in the classification accuracy performance using the discrete emotion data [9]

Feature reduction methods	Machine learning methods			Average (%)
	kNN (%)	SVM(%)	MLP(%)	
SFFS	49.24	50.00	60.30	53.18
SFFS-FP	56.18	57.61	54.01	55.93
mRMR	50.33	49.13	50.54	50.00
mRMR-FP	49.02	48.70	47.51	48.41
IG	36.01	41.52	44.47	40.67
IG-FP	42.52	41.52	40.78	41.61
Average (%)	47.22	48.08	49.60	48.30

However, applying FP with the feature reduction methods, kNN shows a relatively better performance in the classification performance. In overall, the best average performance using feature reduction methods and machine learning methods is MLP as 49.60 %. In the other hand, the best feature reduction method with the various machine learning methods is SFFS-FP with 55.93 % in the classification accuracy performance. In overall, the total average classification accuracy performance is 48.30 %.

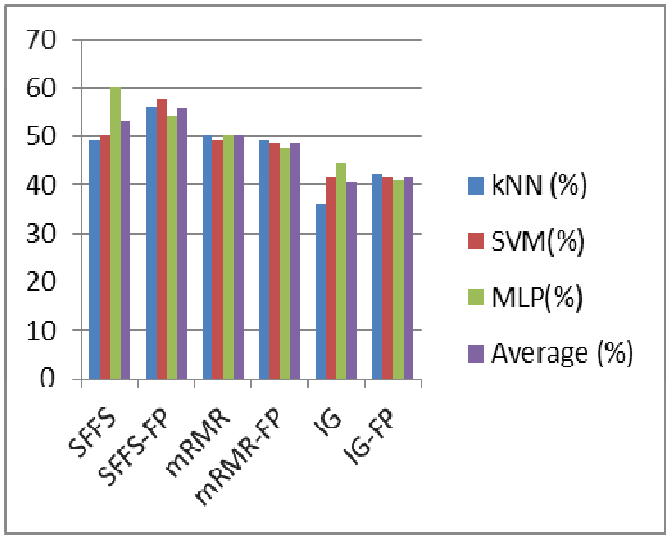


Fig. 2 Comparison of Classification Accuracy Performance between Feature Reduction Methods and Machine Learning Methods

5 Conclusions

As a conclusion, each feature reduction and machine learning methods are presented based upon their classification performance based upon the classification accuracy applying the examined methods. Based on the survey of the explored studies, a preferred method for real- time estimator adaptation under the certain conditions is Multilayer perceptron (MLP) with the various feature reduction methods followed by kNN. However, the data used for the evaluation of the classification performance is only for the discrete emotion data which used by Kukolja et al. [9]. Therefore, for the future study, more various data need to be explored to find out the best feature reduction method and machine learning method with estimating the best performance of the feature reduction methods and machine learning methods.

6 References

[1] R. Picard, "Toward Machine Emotional Intelligence: Analysis of affective physiological state," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 10, Oct. 2001, pp. 1175 – 1190.

[2] J. LeDoux, The Emotional Brain, New York, Simon and Schuster, 1996.

[3] P. Pudil, J. Novovicova, J. Kittler, "Floating search methods in feature selection. " Journal of Pattern Recognition Letters vol. 15, Issue 11, November, 1994, pp. 1119-1125.

[4] D. Ververidis, C. Kotropoulos, "Fast and accurate sequential floating forward feature selection with the Bayes classifier applied to speech emotion recognition," Signal Processing vol. 88 issue 12 December, 2008, pp. 2956-2970.

[5] P. Somol, P.Pudil, J. Novovičová, P. Paclík, "Adaptive floating search methods in feature selection." Pattern Recognition Letters, vol. 20, issue 11, 13 November, 1999, pp. 1157-1163.

[6] Zhanchao Li, Xuan Zhou, Zong Dai and Xiaoyong Zou, "Classification of G-protein coupled receptors based on support vector machine with maximum relevance minimum redundancy and genetic algorithm," BMC Bioinformatics, Vol. 11, 2010, pp. 325-349, http://www.biomedcentral.com/1471-2105/11/325

[7] O. Kursun, C. O. Sakar, O. Favorov, N. Aydin, F. Gurgen, "Using covariates for improving the minimum Redundancy Maximum Relevance feature selection Method," Turk J Elec Eng & Comp Sci, Vol.18, No.6, 2010, 975 – 987.

[8] H. Uğuz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," Knowledge-Based Systems, vol. 24, issue 7, October, 2011, pp. 1024 – 1032.

[9] D. Kukolja, S. Popovic, M. Horvat, B.Kovac, and K. Cosic, "Comparative analysis of emotion estimation methods based on physiological measurements for real time applications," International Journal of Human-Computer Studies, Vol. 72, 2014, pp. 717 – 727.

[10] C. Lee and G. Lee, "Information gain and divergence-based feature selection for machine learning-based text categorization," *Information Processing and Management*, vol. 42 issue 1 January, 2006, p. 155-165.

[11] J. McMurry, *Organic Chemistry,* 7th ed., Brooks/Cole - Thomson Learning, Inc., 2008, p. 975.

[12] N. Shore, *Study Guide and Solutions Manual for Organic Chemistry* (5th Ed.). New York: W.H. Freeman, 2007, pp.182 – 186.

[13] H. Uguz, "A hybrid system based on information gain and principal component analysis for the classification of transcranial Doppler signals," Computer Methods and Programs in Biomedicine vol. 107, issue 3, September, 2012, pp. 598 – 609.

[14] J. Han, M. Kamber, J. Pei, Data mining: Concepts and Techniques, Morgan Kaufmann, Waltham, MA, 2005.

[15] D.T. Larose, KNN, "Discovering Knowledge in Data: An Introduction to Data Mining," 1st ed., Wiley Interscience, New Jersey, 2004, pp. 90 – 106.

[16] R.J. Martis, C. Chakraborty, Arrhythmia disease diagnosis using neural network, SVM, and genetic algorithm-optimized k-means clustering, J. Mech. Med. Biol., Vol. 11, 2011, pp. 897 – 915.

[17] J. Wu, "Efficient HIK SVM Learning for Image Classification," IEEE Trans. Image Process. 21, 2012, pp. 4442 – 4453.

[18] A. Mellit and S. Kalogirou, "Artificial intelligence techniques for photovoltaic applications: A review," Progress in Energy and Combustion Science, vol. 34, issue 5, October, 2008, p. 574 – 632.

TOWARDS INTERACTIVE LEANING FOR OCCUPANCY ESTIMATION

Manar Amayri[1], Stéphane Ploix[1], Patrick Reignier[2], Sanghamitra Bandyopadhyay[3]
[1]G-SCOP lab, Grenoble Institute of Technology, Grenoble, France
[2]LIG lab, Grenoble Institute of Technology, Alpes, Grenoble, France
[3]Indian Statistical Institute, Kolkata, India

Abstract

A new kind of supervised learning approach is proposed to estimate the number of occupants in a room. It introduces the concept of interactive learning where actual occupancy is interactively requested to occupants when it is the most valuable to limit the number of interactions. Occupancy estimation algorithms rely on machine learning: they use information collected from occupants together with common sensors measuring motion detection, power consumption or CO_2 concentration for instance. Two different classifiers are considered for occupancy estimation with interactions: a decision tree C4.5 classifier and parameterized rule based classifier. In this paper, the question of when asking to occupants is investigated. This approach avoids the usage of a camera the determine the actual occupancy.

Keywords: interactive learning, human behavior, building performance, office buildings, machine leaning and data mining

Type of Submission/Paper: Regular Research Papers

1 Introduction

Recently, research about building turns to focus on occupant behavior. Interactive learning opens the gate of the involvement of occupants through the exchange of information. It is a new challenge in occupancy estimation because it solves the issue of the measurement of actual occupancy usually done thanks to cameras and a posteriori labeling, which is both time consuming and invasive. Interactive learning has not been investigated up to now and should be considered as a new approach, that has been applied for occupancy estimation. Most of the works deal with the design stage: the target is to represent the diversity of occupant behavior in order to improve building energy manegment capabilities . Most of the approaches use statistics about human behavior (Roulet et al., 1991; Page et al., 2007; Robinson and Haldi, 2009). (Kashif et al., 2013) emphasized that inhabitants' detailed reactive and deliberative behavior must also be taken into account and proposed a co-simulation methodology to find out the impact of certain actions on energy consumption. Nevertheless, human behavior is not only interesting during the design step, but also during operation. It is indeed useful for diagnostic analyses to discriminate human misbehavior from building system performance, and also for energy management where strategies depend on human activities and, in particular, on the number of occupants in a zone. Such a system as to be trained in each new environment. Unfortunately, using supervised learning algorithm on site is not widely accepted because of the required target to build the set of training data , which usually come from cameras which are not acceptable for many users. In addition, labeling occupancy from videos is time consuming.This paper tackles this issue. It proposes an occupancy estimation approach based on interactive learning with occupants in the studied area. Section 2 presents a state of the art about occupancy estimation. Section 3 discusses the proposed process of interactive learning that interacts with occupants to collect the actual occupancy. Section 4 focuses on C45 decision tree classifier for interactive learning. Section 5 focuses on parameterized classifier for interactive learning and section 6 compares the two classifiers in interactive learn-

ing context in an office.

2 State of the art

Different approaches for estimating occupancy have been investigated but still without using an interactive learning process. Methods vary from basic single feature classifiers that distinguish among two classes (presence and absence) to multi-sensor, multi-feature models. A primary approach, which is prevalent in many commercial buildings, is the usage of passive infrared (PIR) sensors for occupancy. However, motion detectors fail to detect a presence when occupants remain relatively still, which is quite common during activities like working on a computer or regular desk work. This makes the use of only PIR sensors for occupancy counting purpose less attractive. Conjunction of PIR sensors with other sensors can be useful as discussed in (Agarwal et al., 2010). It uses motion sensors and magnetic reed switches for occupancy detection to increase the efficiency of HVAC systems in smart buildings, which is quite simple and non-intrusive. Apart from motion, acoustic sensors (Padmanabh et al., 2009) may be used. However, audio from the environment can easily fool such sensors, and with no support from other sensors, it can report many false positives. In the same way, other sensors like video cameras (Milenkovic and Amft, 2013b), which exploit the huge advances in the field of computer vision and the ever increasing computational capabilities, RFID tags (Philipose et al., 2004) installed on id cards, sonar sensors (Milenkovic and Amft, 2013a) plugged on monitors to identify presence of a person on a computer, have been used and have proved to be much better at solving the problem of occupancy count, yet can not be employed in most office buildings for reasons like privacy and cost concerns. The use of pressure and PIR sensors to determine presence/absence in single desk offices has been discussed in (Nguyen and Aiello, 2012); it further tags activities based on this knowledge. However, for various applications like activity recognition or context analysis within a larger office space, information regarding the presence or absence of people is not sufficient and an estimation of the number of people occupying the space is essential. (Lam et al., 2009) investigates this problem in open offices, estimating occupancy and human activities using a multitude of ambient information, and compare the performance of HMMs, SVMs and Artificial Neural Networks. However, none of these methods generate human-understandable rules which may be very helpful to building managers. In general, an occupancy count algorithm that fully exploits information available from low cost, non-intrusive, environmental sensors and provides meaningful information is an important yet little explored problem in office buildings. This occupancy detection systems still have certain limitations with respect to occupant privacy.

3 Principle of interactive learning

A new methodology for occupancy estimation has been investigated by using interactive learning approach.

Interactive learning is a process involving exchange of information with the users in order to collect some important data according to the problem context. In supervised learning methods, which are used widely in a lot of applications, the problem of the required target arises in the estimation of the number of occupants i.e. the labeling issue usually have been taken from installed video cameras. Using camera is still not acceptable in many places for respecting the privacy of occupants. Interactive learning is an extension of supervised learning machine that in our case will estimate the occupancy by collecting the required labeling from the occupants themselves. The problem statement of occupancy estimation could be formalized like this: let $(A_{1,k}, A_{2,k}, \ldots, A_{n,k}, C_k)$ be a *record* where $A_{i,k}$ is an attribute value, i.e. feature or sensor data, for record k, and C_k a belonging class, i.e. the actual number of occupants provided thanks

to interactions here. $(A_{1,k}, A_{2,k}, \ldots, A_{n,k})$ is named an *ask*: it is an incomplete record. A classifier is defined over $D = dom(A_{1,k}) \times dom(A_{2,k}) \times \cdots \times dom(A_{n,k})$ with $dom(A_{i,k}) = [\check{A_i}, \hat{A_i}]$ where $\check{A_i}$ and $\hat{A_i}$ stand respectively for the maximum and the minimum value of the recorded attributes. It leads to: $\forall i, (A_{1,k}, A_{2,k}, \ldots, A_{n,k}) \in D_i(\Theta) \rightarrow C_k = class_i$ with $\{D_i(\Theta); \forall i\}$, a partition of the attribute space D corresponding of $class_i$. Θ is the list of the classifier parameters. Asking problem is the main issue in interactive learning to define when it is necessary to ask and when asking is useless. Here, the interaction will be with people to give the actual number of occupants, To perform the task of asking the number of occupants, let $\{(A_{1,k}, A_{2,k}, \ldots, A_{n,k}, C_k); \forall k\}$ be a set of records. Let $\{D_i(\Theta); \forall i\}$ be a parameterized classifier with Θ the current parameter values of the classifier. Wherever the model of the classifier is more complicated the parameterized classifier will be more difficult to analyze.

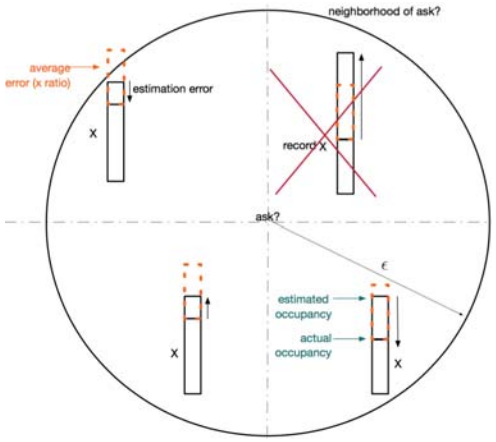


Figure 1: Asking problem

The asking problem consists in determining a utility function for an ask $(A_{1,k'}, A_{2,k'}, \ldots, A_{m,k'})$ taking into account the already available records and the classifier. Three criteria are taken into account to determine whether an ask is potentially useful or not:

1. the density of the neighborhood. It is the number of existing records in the neighborhood of the potential ask. The neighborhood is defined by the distance $\left\| \left(\frac{A_{1,k'} - A_{1,k}}{\hat{A_1} - \check{A_1}}, \frac{A_{2,k'} - A_{2,k}}{\hat{A_2} - \check{A_2}}, \ldots \right) \right\|_\infty < \frac{\epsilon}{2}$ where $\hat{A_i}$ and $\check{A_i}$ stand respectively for the maximum and the minimum value of the recorded attributes A_i. The neighborhood can be modified according to the $\epsilon \in [0, 1]$. The local record density in the neighborhood of an ask $(A_{1,k'}, A_{2,k'}, \ldots)$ is given by equation (1); $(A_{1,k}, A_{2,k}, \ldots, C_k)$ stands for existing record k.
2. the classifier estimation error in the neighborhood of the potential ask that leads to the concept of quality neighborhood. As shown in figure 1, if the classifier estimation error is too high for a record, it is removed from the neighborhood because of the poor quality. A record $(A_{1,k}, A_{2,k}, \ldots, C_k)$ is considered in the quality neighborhood of a potential ask if error is small i.e. $|average(C_k) - occupancy_k| < E_r \zeta$ where $average(C_k)$ stands for the average occupancy of the class C_k, $occupancy_k$ is the actual recorded occupancy collected thanks to an ask, $E_r \in [1, 2)$ typically, is an error ratio than can be adjusted, ζ is the average estimation error of the n existing records: $\frac{1}{n} \sum_{k=0}^{n-1} |average(C_k) - occupancy_k|$.
3. the minimum class weight i.e. the minimum number of records for each class: $weight(class_x) = \left| \{(A_{1,k}, A_{2,k}, \ldots, C_k); C_k = class_x\} \right|$. In case of an potential *ask* $(A_{1,k'}, A_{2,k'}, \ldots)$, the ask will contribute to a class determined by the classifier because ask has not been performed up to now:

$$class = D_{k'}(\Theta, (A_{1,k'}, A_{2,k'}, \ldots))$$

The minimum class weight $weight(class_x) < C_w$ can be adjusted according to the problem.

All the potential asks that satisfy the above three criteria are asked to the occupants in order to possibly become an additional record.

if record density in quality neighborhood of a potential ask (neighborhood without

records with big estimation error) is low or $weight(class_x) \leq C_w$ **then**
Ask
else
Do not Ask
end if

$$d(A_{1,k'}, A_{2,k'}, \dots) = \frac{\left| \left\{ \begin{array}{c} (A_{1,k}, A_{2,k}, \dots, C_k); \dots \\ (A_{1,k}, A_{2,k}, \dots) \in \dots \\ \left\| \left(\frac{A_{1,k'} - A_{1,k}}{\hat{A_1} - \check{A_1}}, \frac{A_{2,k'} - A_{2,k}}{\hat{A_2} - \check{A_2}}, \dots \right) \right\|_\infty < \frac{\epsilon}{2}, \\ \dots \epsilon \in (0,1], \forall k \end{array} \right\} \right|}{n} \quad (1)$$

For validation, occupant reaction has to be taken into account as a response probability p i.e. whether the occupants answer or not. For a given context, the number of asks depends on the classifier used for occupancy estimation:

- C45 decision tree classifier can be used directly together with the ask mechanism to generate occupancy labels for preparing training data .
- a parameterized classifier can be used as well where parameters can be adjusted according to the growing number of records.

These classifiers together with the proposed ask mechanism are presented in the 2 next sections.

4 Decision tree classifier

The test bed is an office in Grenoble Institute of Technology, which accommodates a professor and 3 PhD students. The office has frequent visitors with a lot of meetings and presentations all through the week. The set-up for the sensor network includes:

- 2 video cameras for recording real occupancy numbers and activities. Those two cameras are only used for validation purpose.
- An ambiance sensing network, which measures luminance, temperature, relative humidity (RH), motions, CO2 concentration, power consumption, door and window position, microphone. Data are sent thanks to ENOCEAN protocol on significant value change event.
- A centralized database with a web-application for retrieving data from different sources continuously.

To perform the task of finding the number of occupants, a relation has to be discovered between the office environment and the number of people in it . The office environment can be represented as a set of state variables, $A_t = [A_1, A_2, \dots, A_m]_t$. This set of state variables A at any instance of time t must be indicative of occupancy. A state variable can be termed as a feature, and therefore the set of features as feature vector. Similarly, the m-dimensional space that contains all possible values of such a feature vector is the feature space. The underlying approach for the experiments is to formulate the classification problem as a map from a feature vector into some feature space that comprises several classes of occupancy or activities. Therefore, the success of such an approach heavily depends on how good the selected features are. In this case, features are attributes from multiple sensors accumulated over a time interval. The choice of interval duration is highly context dependent, and has to be done according to the granularity required.

Features is the information extracted from the data i.e acoustic pressure from a microphone, time slot, occupancy from power consumption, door or window position, motion counting,day type, indoor temperature,... One quantitative measurement of the usefulness of a feature is *information gain*, which depends on the concept of *entropy* (Amayri et al., 2015). Information gain is helpful to distinguish among a large set of features, the most worthwhile to consider for occupancy estimation.

A supervised learning approach has been used. Occupancy has been determined before using a classification algorithm: occupancy counting was manually annotated using a video feed from two cameras strategically positioned in an office to simulate the

occupant replies, determine the structure of parameterized classifier and validate interactive learning results.

The *decision tree* classification technique has been selected because it provides both very good results and the results are easy to analyze and adapt. The decision tree algorithm selects a class by descending a tree of decision nodes. Each internal node represents a comparison of a single feature value with a learned threshold. The target of the decision tree algorithm is to select features that are more useful for classification. Finally, because decision trees are human readable, they can be adjusted using expert knowledge and extract the estimation rules (if-then) from the decision tree structure.

if $X_i \leq threshold$ **then**
  left child node
else
  right child node
end if

5 Parameterized classifier

Another approach for occupancy estimation is investigated because it fits well with interactive learning. It uses a predetermined classifier structure with parameters to be adjusted according to the incoming records. Any classifier could be used in this approach, but still it is important to choose a general structure for the sake of adaptability. Additionally, the number of parameters should be low because the tuning mechanism relies on an optimization process that may become inefficient when complexity increases. A depth-limited *decision tree* classifier has been selected here. The depth of a tree varies depending upon the size and nature of the sample set. For example if the depth of the tree is set to '1', a tree with a single node is generated. Otherwise, the most complicated case builds a complete tree, where every path test every feature. Limiting the depth avoids data over-fitting phenomena by rejecting non significant features. Assuming n_s samples and n_f features, at each level (i), the remaining $(n_f - i)$ features for each sample at the level (i) should be examined to calculate the information gain. However, learned trees are rarely complete (number of leaves is lower or equal to n_s). In practice, complexity is linear in both number of features (n_f) and number of training samples (n_s). In addition, a maximum tree depth of d will limit the maximum number of rules for a decision to d. In general, a deep tree with many leaves is usually highly accurate on the training data but less with the validation data. In addition, finding a shortest decision tree is preferred over longer trees: it is indeed easier to understand and more reliable, it is also easier to implement and to use. Tuning problem can be solved by adjusting the classifier parameters (node thresholds of the decision tree) in the final structure according to each updated record set and how much it's different from the previous one. An objective function is determined to minimize the distance between actual (coming from an ask) and estimated (coming from the classifier) number of occupants in the room. Optimization covers a required period of asking, interacting with the occupants in the studied area.

A depth equal to 2 is the limitation chosen for the next analysis of occupancy estimation because of the low average error of the resulting decision tree and of the little number of thresholds to adjust. Additionally, the tree is readable and rules are quite general as it is shown in figure 2.

if microphone is low **then**
  ≈ 0 person
else if microphone is high and CO2 physical model is low **then**
  ≈ 1 person
else if microphone is high and CO2 physical model is high **then**
  ≈ 2 persons
end if

Note that, (if-then) rules from the tree structure could be extracted now easily to be applied in a tuning context.

6 Results

The data covers 11 days from 04-May-2015 to 14-May-2015. During these 10 days, an Human Machine Interface (HMI) is assumed to be used to interact with the users in the office.

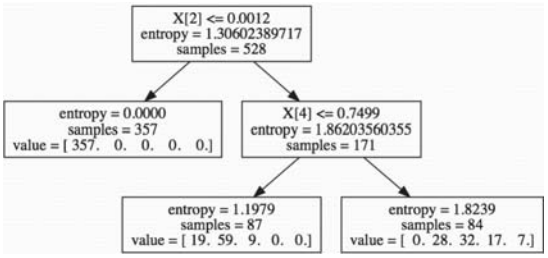


Figure 2: Decision tree used by the parameterized classifier

In this HMI, an alarm is triggerer to ask the user the actual number of occupants. The replies of the occupants are modeled but a random process with a reply probability 50% i.e. only half of the asks get replies.

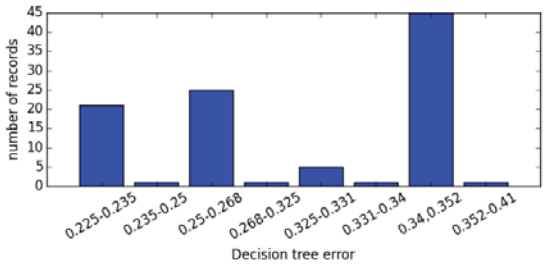


Figure 3: Distribution of decision tree error with 13 asking

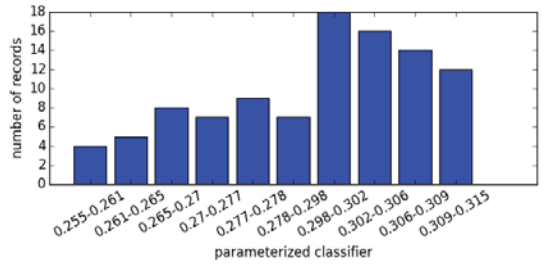


Figure 4: Distribution of parameterized classifier error with 13 asking

Both estimation methods have been applied. The interactive learning process has been performed 100 times to show the distribution of the error for both *decision tree* (see figure 3) and parameterized classifier (see figure 4), with 13 asks, $\epsilon = 0.5$, and $E_r = 1.5$.

For the first 13 asks, the parameterized classifier is giving better results than the whole decision tree. While reproducing the same interactive learning process with 50 asks, figures 5 and 6 with $\epsilon = 0.5$, and $E_r = 1.5$ are obtained. The decision tree starts to give better results. However *decision tree* needs more than 13 asks for training data to build an acceptable estimator. Additionally, it is important to notice that increasing ϵ decreases the number of asks while increasing the error ratio decreases the number of asks. It can be noticed also that the asking process is dependent on the classifier used because the estimation error intervenes. The following table illustrates how the 13 asks are distributed along the days with parameterized estimator. Asking process with decision tree leads to almost the same results depending of the run that contains randomness because of the ask replies.

Day	1	2	3	4	5	6	7	8	9	10
Asking	9	3	0	0	0	0	0	1	0	0

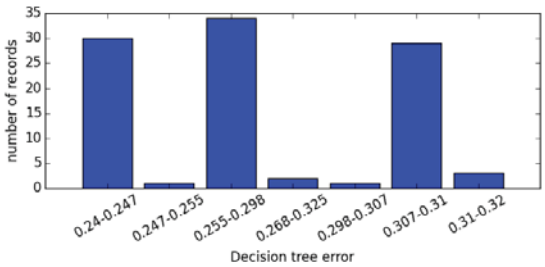


Figure 5: Distribution of decision tree error with 50 asking

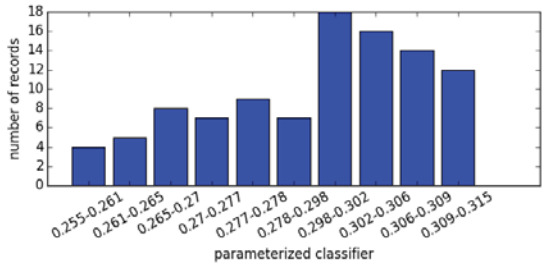


Figure 6: Distribution of parameterized classifier error with 50 asking

7 Conclusion

An interactive learning approach has been proposed in this paper to avoid the manual labeling of actual occupancy in a room for

supervised learning approaches. It is then possible with a little number of interactions with occupants to estimate the number of occupants in a zone. Two different classifiers have been tested together with the interactive ask process: a pure C45 decision tree algorithm and parameterized rule based classifier. The approach can be easily extended to any kind of classifier.

The C45 decision tree algorithm is very general because its structure is not assumed: it is discovered from the data and can be therefore extended to any room with any sensors. It leads to the best results after about 13 asks. The parameterized classifier yields better results at first but because the number of parameters (2) is much less than the decision tree (about 45 parameters), the decision tree finally better estimates the number of occupants although the parameterized classifier directly minimizes the estimation error and the classification (with C45, classifying in class 2 or in class 3 instead of class 1 has the same impact). Because the structure of parameterized classifier is predefined, the adaptation capability to another context is much less: a relevant structure has to be proposed.

The impact of the modality of interactions through the human machine interface has still to be investigated.

REFERENCES

Agarwal, Y., Balaji, B., Gupta, R., Lyles, J., Wei, M., Weng, T., 2010. Occupancy-driven energy management for smart building automation. In: Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building. ACM, pp. 1–6.

Amayri, M., Ploix, S., Bandyopadhyay, S., 2015. Estimating occupancy in an office setting. ASCE conference 2015, Pittsburgh, USA.

Kashif, A., Dugdale, J., Ploix, S., 2013. Simulating occupants' behaviour for energy waste reduction in dwellings: A multi agent methodology. Advances in Complex Systems 16, 37.

Lam, K. P., Höynck, M., Dong, B., Andrews, B., shang Chiou, Y., Benitez, D., Choi, J., 2009. Occupancy detection through an extensive environmental sensor network in an open-plan office building. In: Proc. of Building Simulation 09, an IBPSA Conference.

Milenkovic, M., Amft, O., 2013a. An opportunistic activity-sensing approach to save energy in office buildings. In: Proceedings of the fourth international conference on Future energy systems. ACM, pp. 247–258.

Milenkovic, M., Amft, O., 2013b. Recognizing energy-related activities using sensors commonly installed in office buildings. Procedia Computer Science 19, 669–677.

Nguyen, T. A., Aiello, M., 2012. Beyond indoor presence monitoring with simple sensors. In: PECCS. pp. 5–14.

Padmanabh, K., Malikarjuna V, A., Sen, S., Katru, S. P., Kumar, A., Vuppala, S. K., Paul, S., et al., 2009. isense: a wireless sensor network based conference room management system. In: Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings. ACM, pp. 37–42.

Page, J., Robinson, D., Scartezzini, J., 2007. Stochastic simulation of occupant presence and behaviour in buildings. Proc. Tenth Int. IBPSA Conf : Building Simulation, 757–764.

Philipose, M., Fishkin, K. P., Perkowitz, M., Patterson, D. J., Fox, D., Kautz, H., Hahnel, D., 2004. Inferring activities from interactions with objects. Pervasive Computing, IEEE 3 (4), 50–57.

Robinson, D., Haldi, F., 2009. Interactions with window openings by office occupants. Energy and Buildings 44, 2378–2395.

Roulet, C., Fritsch, R., Scartezzini, J., Cretton, P., 1991. Stochastic model of inhabitant behavior with regard to ventilation. Technical report.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

ARTIFICIAL INTELLIGENCE, MODELING, NOVEL APPLICATIONS AND TOOLS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Detecting Reports of Mass Emergency Events on Twitter

V. Pekar[1], J. Binner[1], H. Najafi[2], and C. Hale[3]
[1]Business School, University of Birmingham, Birmingham, United Kingdom
[2]Computer Science and Information Systems, University of Wisconsin, River Falls, WI, USA
[3]Electronic Systems Lab, Georgia Tech Research Institute, Dayton, OH, USA

Abstract *The paper addresses the problem of detecting eyewitness reports of mass emergencies on Twitter. This is the first work to conduct a large-scale comparative evaluation of classification features extracted from Twitter posts, using different learning algorithms and datasets representing a broad range of mass emergencies including both natural and technological disasters. We investigate the relative importance of different feature types as well as on the effect of several feature selection methods applied to this problem. Because the task of detecting mass emergencies is characterized by high heterogeneity of the data, our primary focus is on identifying those features that are capable of separating mass emergency reports from other messages, irrespective of the type of the disaster.*

Keywords: text classification, machine learning, social media analysis, disaster management, risk assessment

1 Introduction

Social media data offer a promising possibility to deal with mass emergencies. The present-day ubiquity of mobile devices has meant that during a crisis such as a flood, earthquake or a terrorist attack, social media becomes a primary source of information, publishing eyewitness reports on the events in real-time. This gives an opportunity for emergency services to detect crises at early stages, monitor their development and tackle their consequences more effectively.

The potential of social media analysis for mass emergency management has attracted many Data Mining researchers over the past several years. The problem of detecting eyewitness accounts of emergency events in social media has been primarily approached with text classification methods based on machine learning. Limiting the problem to a narrow domain such as earthquakes or tornados has been shown to produce high classification accuracy (e.g., [3, 4, 8, 10, 13, 20]).

However, mass emergency events differ a lot and a classification method that would cover a wide range of possible disasters would be much more practical. This paper is concerned with the broader task of recognizing emergencies unspecified for a particular type, which could include both natural disasters such as earthquakes, floods and storms, as well as man-made ones such as explosions, collisions and shootings. This is a non-trivial classification problem. Firstly, messages relating to a crisis event include not only actual eyewitness accounts but also those that have to do with official announcements, offers of help, sympathy, criticism, and so on. Olteanu et al. [11] report that of all messages judged to be relevant to one of twenty-six mass emergencies, eyewitness accounts comprise only around 8%. The challenge is therefore in identifying specifically eyewitness reports among messages that talk about largely the same event; this is also a classification problem with a big bias towards the negative class. Secondly, because the automatic classifier is expected to operate on a broad variety of event types, each characterized by its own vocabulary. The data is thus not homogeneous: data instances come from related, but different distributions, and in real-world use cases training data is likely to be insufficiently representative of test data on which the classifier is evaluated.

To address these challenges, we study classification features that can be extracted from Twitter messages, beyond the traditional text-based features, that would be suited specifically to the task at hand. Until now, previous papers on detecting emergency-related messages used their own set of features; a few studies examined their contribution to classifier accuracy, but only within a specific application, often limited to one learning algorithm and one emergency event. In this paper we describe a comparative evaluation of a broad set of features that includes those that were used in previous work as well as those introduced for the first time, conducting experiments on data from 26 different emergency events. We report on features that are robust against data heterogeneity and help achieve better classification accuracy when the classifier is evaluated on data from an emergency event that is different from the events exemplified in the training data.

2 Related Work

There is a considerable body of work on detection of new events in a stream of messages, where the type of the event of interest is not known in advance, and some of these approaches were applied to detecting mass emergency events. Such methods primarily rely on detecting "bursty" keywords [9], i.e. keywords whose frequency increases sharply within a short time window, or bursty message clusters [14]. However, bursty keywords are known to be related not only to new

events, but also recurring events and even non-events. To separate them, Becker et al. [2] used a domain-independent text classifier, before applying keyword burstiness techniques.

Domain-specific methods generally have a greater accuracy than domain-independent ones, and previous work specifically on emergency event detection was concerned with developing domain text classifiers based on machine learning and operating on features extracted from the entire message. Most of this work dealt with specific types of crises such as earthquakes [3, 19, 20], tornados [4, 8], and landslides [10]. Only a few studies developed classifiers that would be applied to more than one type of disasters: Verma et al. [18] evaluated their method on three different types of crises, while Ashktorab et al. [1] on twelve.

Classification features typically include unigrams (e.g., [1, 13]), bigrams [18, 19], message length [10, 13], part-of-speech tags [4, 18], VerbNet categories [4], the proportion of words that are present in a pre-defined vocabulary [10], whether place names are present [10], hashtags [4, 19], if the message is a retweet [4, 19] or a reply [2]. Verma at al. [18] looked at the contribution of three other kinds of features: if the language of the message objective or subjective, if the register is formal or not, if the text is a first-person report or not.

Any direct comparison between previous approaches is difficult, because they used different experimental datasets, different classification algorithms, and the classification tasks were somewhat different. For example, Imran et al. [5] classified messages into "informative" and "non-informative", Ashktorab et al. [1] into those that report damage and those that do not, Verma et al. [18] into those that are related to situational awareness and those that are not.

3 Classification features

In our evaluation we include the following types of features (examples are shown in parentheses):

Lexical:

`Unigrams`: whitespace-separated word tokens (nominal: *please, help, fire*).

`Bigrams`: token sequences with the length of two (nominal: *was_scary, we_complained*).

`NumberOfUnigrams`: the length of the messages, measured in unigrams. Sakaki et al. [13] found that it was a useful class predictor, as in their data eye-witness accounts tended to be short messages (continuous).

Grammatical:

`Verbs&Nouns`: only word tokens that are tagged as verbs and nouns. The intuition behind these features is that events and their participants are usually described with verbs and nouns, and thus events can be more accurately classified by focusing on verbs and nouns found in the message (nominal: *construction, floor, stuck*).

`PartOfSpeechTags`: separate features are created from part-of-speech (PoS) categories, as assigned by a PoS tagger, the reasoning being that the greater incidence of specific parts of speech (e.g., verbs and nouns) may be more indicative of an eye-witness report (nominal: *NNS, JJ, VBD*).

Semantic:

`VerbNetCategories`: VerbNet [6] is a lexical resource encoding English verbs and different semantic information on them, including their semantic categories. Following Imran et al. [4], for each verb found in a tweet, we add a feature corresponding to its VerbNet category in order to generalize the meaning of specific verbs (nominal: *complain-37.8, get-13.5.1*).

`EMTCategories`: Emergency Management Terms [17] is a lexicon containing around 7,000 words and expressions semi-automatically extracted from Twitter messages on different public emergencies. Each item in the lexicon is associated with a semantic category such as "Caution and Advice", "Injured People", "Infrastructure damage". We detect EM terms in the tweets and use their category labels as features (nominal: *T04, T07, O02*).

`NamedEntities`: We map all named entities, as detected and tagged by the PoS tagger, to a category label, and use it as a feature, instead of actual word tokens (nominal).

Stylistic:

`Sentiment`: We process each tweet with a domain-independent sentiment analysis system [12] and create a feature indicating whether the tweet is neutral in terms of sentiment or not; the system detects emoticons and uses them to determine the sentiment of the message (Boolean).

`Personal`: Following Verma et al. [18], we create a feature indicating if the message contains first-person pronouns ("I", "we", "me", "us") or not, expecting that eyewitness accounts of emergencies will be written from a first-person perspective (Boolean).

`All caps`: We create a feature indicating if the tweet contains all-caps words or not, as words spelled all in capital letters are meant to represent shouting, i.e. used when the author wants to attract special attention to the tweet (Boolean).

Twitter metadata:

`Hashtags`: A hashtag is a word or concatenated phrase preceded by the hash symbol, which are used by authors of messages to group tweets on the same topic and indicate important keywords; we create one extra feature for each hashtag found in a tweet (nominal: *#sandy, #haze*).

`ContainsHashtags`: whether or not the tweet contains any hashtags (Boolean).

`Mentions`: A mention is the name of a Twitter account that is included into the message in order to attract that user's attention to the tweet. We hypothesize that in case crises are reported, the tweet would mention the same set of Twitter accounts (e.g., news agencies, police, or government bodies). We create one feature for each mention found in the tweet (nominal: *@newscaster, @News1130radio*).

`ContainsMentions`: whether or not the message mentions one or several Twitter accounts. Becker et al [2] found that presence of mentions correlates with reports of emergency events (Boolean).

`RetweetCount`: the number of times the message has been retweeted. We anticipate that eyewitness accounts are likely to attract more interest than other tweets and thus would be retweeted more (continuous).

`Reply`: whether the message is a reply to a different message. In accordance with Becker et al.'s findings [2], we expect that eyewitness accounts will not be replies to previous messages (Boolean).

`ContainsURL`: whether the tweet contains a URL. We expect that eyewitness accounts will tend not to mention any previously published information such as external URLs (Boolean).

Prior to training and classification, all features are converted to the continuous values.

4 Feature selection

Feature selection is a common step in machine learning scenarios, and in particular in text classification, where the number of features is usually very large. It is performed in order to eliminate noisy features, minimize overfitting of the classifier to the training data and to improve its efficiency. In supervised settings, i.e., when class membership of instances is known, the *filtering* approach to feature selection is commonly followed. For an overview of feature selection methods used in text classification, see [15].

In the context of tweet classification the filtering approach can be formalized as follows. Let us assume that each tweet $t \in T$ of the training set is represented as a feature vector, consisting of features $f \in F$, and that each t is assigned a class label $c \in C$. For each f, from its distribution across C, a certain function computes its informativeness score, specific to each class. From class-specific scores, one can compute its global score by, e.g. averaging local scores of f across classes. The features are then sorted by their informativeness and k top features are selected to represent instances, with k set experimentally. After non-informative features have been removed from the training data, a classifier is learned and evaluated on the test data.

A key decision for feature selection is to choose a function computing $w(f,c)$. Such functions aim to capture the intuition that the best features for a class are the ones that best discriminate between its positive and negative examples. They determine $w(f,c)$ from the distribution of f between c and non-c, attributing greater weights to those f that correlate with c or non-c the most. In the present study we include three such functions widely used in text categorization:

Chi-square. The chi-square (CHI) statistic measures the lack of independence between f and c, and is computed from a two-way contingency table with these two variables.

Information Gain. IG measures the number of bits of information obtained about presence and absence of c by knowing the presence or absence of f.

Information Gain Ratio. IGR aims to overcome one disadvantage of IG which is the fact that IG grows not only with the increase of dependence between f and c, but also with the increase of the entropy of f. IGR removes this factor by normalizing IG by the entropy of c.

5 Experimental setup

5.1 Data

For experimental evaluation we use the labeled part of the CrisisLexT26 dataset [11], which includes tweets on 26 mass emergencies that occurred in 2012 and 2013. The types of emergencies are very diverse and range from terrorist attacks and train derailment to floods and hurricanes. Some examples are Colorado wildfires in 2012, Venezuela refinery explosion in 2012, Boston bombings in 2013.

There are 24,589 labeled tweets in the dataset in total, with 2,193 of them labeled as originating from an eyewitness. The classification task in our experiments consisted of predicting whether a given tweet was an eyewitness report or not.

5.2 Preprocessing

We apply the following preprocessing steps to the data:

Additional metadata. The CrisisLexT26 data contains the Twitter id of the message, its raw content, and its timestamp. Via Twitter Search API we retrieve additional metadata fields: retweet count, reply, hashtags.

Deduplication. Duplicate tweets were removed by measuring similarity in each pair of tweets using cosine and removing one tweet in pairs where the cosine was higher than 0.99.

Tokenization and part-of-speech tagging. Before processing the text of the message with a PoS tagger, the text was normalized: mentions (e.g., @someone) and URLs removed; sequences of hashtags at the start and end of the message removed; hashtags appearing in the middle of the text were kept, but the hash symbol removed from the hashtags; long non-alphanumeric symbol sequences, which tend to be emoticons, were removed; word tokens consisting of digits were replaced with a unique tag. The normalized text was tokenized and tagged with the PoS tagger in the Pattern library [17].

Sentiment analysis. The original text of the tweet was processed with the sentiment analysis system [13]. The system was used in the SemEval ABSA challenge, where it achieved an F-measure of 0.67 and 0.75 on the two evaluation datasets within the sentiment analysis subtask. The system assigned to an input text a sentiment score between -1.0 (negative) and 1.0 (positive); the score was converted to a Boolean value indicating if the tweet is neutral in terms of sentiment (the score was equal 0) or not.

Stopword removal. The usual stoplist was used to remove stopwords.

5.3 Evaluation Metrics

The accuracy of classification is measured in terms of the traditional measures of precision, recall and F1 measure. Because the data is biased towards the negative class, the evaluation metrics averaged over both classes may be misleading, so we report them only for the positive class, i.e. the eyewitness report class.

6 Results and Discussion

6.1 Impact of Data Heterogeneity

In the first experiment, we examined the extent to which data heterogeneity present in the CrisisLexT2 dataset affects classification accuracy. To that end, we evaluated the classifiers in two scenarios. In the first ("Scenario 1"), the entire dataset was randomly split into a train and a test set, in proportion 1 to 9. This ensured, with a large likelihood, that data on the same crisis will be present in both training and test data, and the feature distribution in the test data will be similar to the one in the train data.

The second scenario ("Scenario 2") was meant to better reflect real-world use cases: the train-test split was done in such a way so that the test data contained tweets only on those crises that were not included into the train data, i.e., simulating the conditions when a crisis needs to be detected before any manually labelled data relating to it are available. Specifically, data on 23 crises were used for training and data on 3 remaining crises were used for testing.

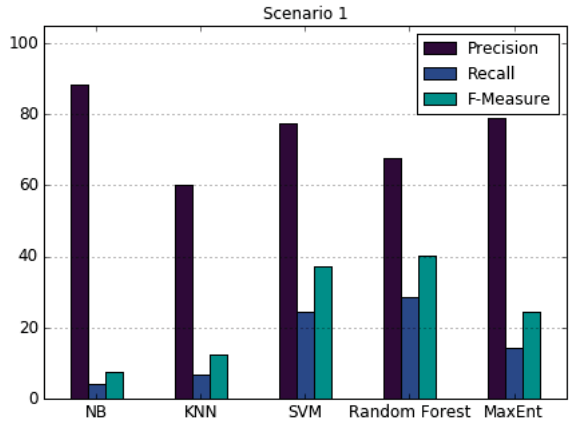


Figure 1. Classifier performance on the full set of features, random train-test split ("Scenario 1").

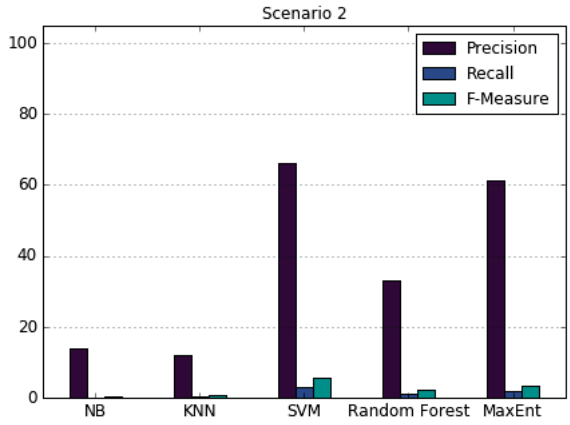


Figure 2. Classifier performance on the full set of features ("Scenario 2").

Training on all the features described in Section 3, we compared the performance of five classifiers – Naive Bayes, k Nearest Neighbors (kNN, k=5), Random Forest, Maximum Entropy (MaxEnt, a.k.a. Logistic regression) and linear Support Vector Machines (SVM), under these two scenarios. The results are shown in Figures 1 and 2

The results show that Scenario 2 is indeed a much harder evaluation task: both precision and recall rates for all the five classifiers drop; the drop is especially big for recall (e.g., for SVM it falls 0.24 from to 0.01). This suggests that, as anticipated, there is more data heterogeneity between different crises than within them. In our subsequent experiments, we used the SVM and MaxEnt classifiers, which fared better than the other three.

6.2 Feature types

To measure the relative utility of each type of features, we ran experiments where each feature type was removed from the full set of features and the change in classifier performance was noted. The results for Scenario 1 are shown in Tables 1 (SVM) and 2 (MaxEnt), for Scenario 2 – in Tables 3 (SVM) and 4 (MaxEnt), the tables show the percent changes in F-measure, precision and recall resulting from removing one feature type.

For Scenario 1 results, we see that the changes are not very significant, except for `Hashtags`, which contribute a lot to the recall of the classifiers (up to 5 points), `Bigrams`, which help precision for Maxent and recall for SVM, and `RetweetCount` and `NamedEntities`, whose removal leads to improvements in all the three metrics, by up to 9 points. Some features increase precision at the cost of recall (`NumberOfUnigrams`), while others, on the contrary, improve recall at the cost of precision (`HashTags`, `EMCategories`).

	F-measure	Precision	Recall
Bigrams	-3.39	0.70	-2.91
Hashtags	-3.10	0.16	-2.53
Mentions	-1.51	-1.47	-1.18
ContainsMentions	-0.83	-0.08	-0.63
EMCategories	-0.78	-0.75	-0.58
Sentiment	-0.39	0.70	-0.39
AllCaps	-0.35	-0.71	-0.22
Personal	-0.34	0.05	-0.29
Reply	-0.23	0.54	-0.21
PosTags	-0.20	-0.98	-0.08
ContainsUrl	-0.16	-0.11	-0.12
Verbnet	0.03	-0.15	0.03
ContainsHashtags	0.22	0.10	0.21
VerbsAndNouns	0.61	-1.19	0.70
NumberOfUnigrams	0.94	-0.80	0.99
NamedEntities	3.10	1.14	2.67
RetweetCount	9.05	-0.22	8.51

Table 1. The effects of removing one feature from the feature set, Scenario 1, SVM.

	F-measure	Precision	Recall
PosTags	-2.66	-0.77	-1.79
Hashtags	-2.42	1.01	-1.62
EMCategories	-1.19	1.29	-0.84
ContainsHashtags	-1.14	-0.16	-0.76
Personal	-0.82	-0.72	-0.54
AllCaps	-0.32	-0.74	-0.16
Mentions	-0.28	-0.55	-0.15
Sentiment	-0.17	-1.33	-0.07
Reply	-0.14	0.0	-0.05
ContainsMention	0.04	0.68	0.02
ContainsUrl	0.04	-1.78	0.13
Verbnet	0.07	-0.64	0.08
VerbsAndNouns	0.44	-0.42	0.32
Bigrams	1.84	-4.92	1.54
NumberOfUnigrams	2.50	-1.80	1.87
NamedEntities	3.61	1.40	2.55
RetweetCount	9.50	2.57	7.04

Table 2. The effects of removing one feature from the feature set, Scenario 1, MaxEnt.

For Scenario 2, the changes in F-measure are not high, but differences between specific features in terms of precision and recall are much more noticeable than for Scenario 1. For both classifiers, `Bigrams` are important for precision, the changes are 11.7 points for SVM and 8.2 for MaxEnt, while `EMCategories` and `Personal` help to improve recall. The use of `PosTags` improves all the evaluation metrics for both classifiers. `RetweetCount`, `VerbNet`, `HashTags` and `VerbsAndNouns` produce an adverse effect on all the three metrics, also for both SVM and MaxEnt. A somewhat unexpected observation is that `PosTags` are positively influencing all the metrics, for all classifiers and scenarios. The changes for other features are less consistent between the classifiers.

	F-measure	Precision	Recall
Bigrams	-1.97	-11.70	-1.08
Personal	-0.97	-5.20	-0.55
NamedEntities	-0.26	-2.48	-0.15
ContainsHashtags	-0.25	6.82	-0.17
PosTags	-0.22	-3.44	-0.12
EMCategories	-0.20	2.20	-0.13
Sentiment	-0.18	-2.44	-0.11
ContainsMention	-0.15	-0.05	-0.09
ContainsUrl	-0.02	1.77	-0.04
Reply	0.0	0.0	0.0
Mentions	0.03	3.71	0.01
AllCaps	0.03	-0.85	0.01
Verbnet	0.22	0.44	0.11
Hashtags	0.53	9.39	0.26
RetweetCount	1.25	-3.49	0.75
NumberOfUnigrams	1.3	-5.16	0.76
VerbsAndNouns	1.58	6.46	0.85

Table 3. The effects of removing one feature from the feature set, Scenario 2, SVM.

	F-measure	Precision	Recall
EMCategories	-1.13	2.24	-0.63
PosTags	-0.97	-38.77	-0.51
ContainsHashtags	-0.68	-2.0	-0.38
Personal	-0.68	5.52	-0.39
ContainsUrl	-0.34	-4.71	-0.19
NamedEntities	-0.31	2.31	-0.19
Sentiment	0.0	0.0	0.0
Mentions	0.0	0.0	0.0
AllCaps	0.14	1.17	0.07
Reply	0.15	1.85	0.07
VerbsAndNouns	0.37	5.27	0.18
ContainsMention	0.40	2.22	0.22
Bigrams	0.62	-8.24	0.36
Verbnet	0.76	2.46	0.41
Hashtags	0.93	2.77	0.50
RetweetCount	0.98	-8.69	0.56
NumberOfUnigrams	1.55	2.45	0.85

Table 4. The effects of removing one feature from the feature set, Scenario 2, MaxEnt.

More generally, it seems that lexical features such as `Bigrams` help to achieve greater precision, while Semantic (e.g., `EMCategories`), Stylistic (e.g., `Personal`) and Twitter-related (e.g., `ContainsHashtags`) ones – greater recall. These characteristics of the features become more prominent in Scenario 2.

6.3 Feature Selection

In the next experiment, we examined the ability of the Chi-Square, Information Gain and Information Gain Ratio to select the most useful classification features. Computing scores for all the features, we experimented with keeping the top 10%, 20%, …, 90% of the most informative features. These results are shown in Figures 3, 4, and 5.

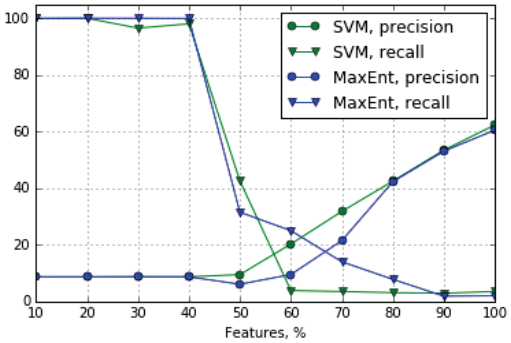


Figure 3. The effect of feature selection based on CHI on precision and recall of SVM and MaxEnt.

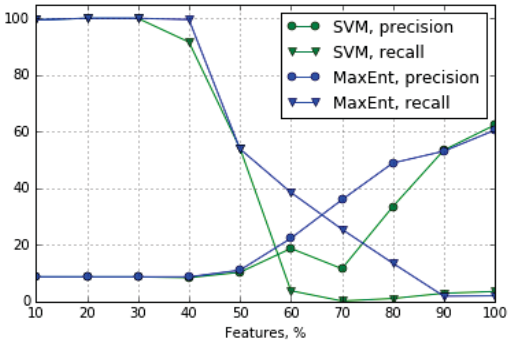


Figure 4. The effect of feature selection based on IG on precision and recall of SVM and MaxEnt.

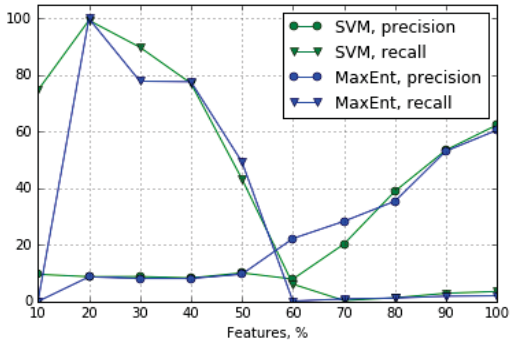


Figure 5. The effect of feature selection based on IGR on precision and recall of SVM and MaxEnt.

We see that the effect of feature selection is largely similar for CHI, IG and IGR. When features are removed drastically (keeping 40% of features or less), the recall improves all the way to 100%, while precision drops to about 10%. The fewer features are removed, the greater precision and the lower recall, with the best precision achieved when keeping 100% of features, both classifiers and all three feature ranking functions.

7 Conclusion

This is the first work to conduct a large-scale comparative evaluation of classification features extracted from Twitter posts, using different learning algorithms and datasets representing a broad range of mass emergencies including both natural and technological disasters. Our key findings can be summarized as follows. We presented empirical results demonstrating that a machine learning classifier tested on data that represents mass emergency events that were unseen at the training stage suffers a significant performance drop, especially in terms of recall, in comparison to testing on data that represents the same types of emergency events as the train data. We furthermore find that when testing the classifier on unseen event types, lexical features help to achieve better precision, while semantic, stylistic, and features derived from message metadata help improve recall. Finally, we examined several well-known feature selection methods, finding that they all produce a similar effect on the classifier: at aggressive levels of feature selection, they lead to better recall; however, they do not help much with precision.

This work has thus produced results that can inform development of applications for automatic detection of social media posts relating to mass emergencies, with regards to the choice of features to be used under different use cases. Future research will focus on ways to exploit the described properties of the features: for example, create different feature subsets constituting different "views" on the data within a semi-supervised learning method.

8 References

[1] Zahra Ashktorab, Christopher Brown, Manojit Nandi, and Aron Culotta. 2014. Tweedr: Mining Twitter to inform disaster response. In Proc. of ISCRAM.

[2] Hila Becker, Mor Naaman, and Luis Gravano. 2011. Beyond trending topics: Real-world event identification on Twitter. Proc. of ICWSM. 438–441.

[3] Cornelia Caragea, Nathan McNeese, Anuj Jaiswal, Greg Traylor, H. Kim, Prasenjit Mitra, Dinghao Wu, A. Tapia, Lee Giles, Bernard J. Jansen, and others. 2011. Classifying text messages for the Haiti earthquake. In Proc. of ISCRAM.

[4] Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, Patrick Meier. 2013. Extracting Information Nuggets from Disaster-Related Messages in Social Media. Proceedings of the 10th International ISCRAM Conference.

[5] Muhammad Imran, Carlos Castillo, Ji Lucas, Patrick Meier, and Sarah Vieweg. 2014. AIDR: Artificial intelligence for disaster response. In Proc. of WWW (Companion). IW3C2, 159–162.

[6] Karin Kipper, Anna Korhonen, Neville Ryant and Martha Palmer. Extending VerbNet with Novel Verb Classes. Proceedings of the Fifth International Conference on Language Resources and Evaluation -- LREC'06. May, 2006, Genoa, Italy: 2006.

[7] Rui Li, Kin Hou Lei, Ravi Khadiwala, and KC-C Chang. 2012. Tedas: A Twitter-based event detection and analysis system. In Proc. of ICDE. IEEE, 1273–1276.

[8] Benjamin Mandel, Aron Culotta, John Boulahanis, Danielle Stark, Bonnie Lewis, and Jeremy Rodrigue. 2012. A demographic analysis of online sentiment during hurricane Irene. In Proc. of the Second Workshop on Language in Social Media (LSM '12). Association for Computational Linguistics, Stroudsburg, PA, USA, 27-36.

[9] Adam Marcus, Michael S. Bernstein, Osama Badar, David R. Karger, Samuel Madden, and Robert C. Miller. 2011. Twitinfo: Aggregating and visualizing microblogs for event exploration. In Proc. of CHI. 227–236.

[10] Aibek Musaev, De Wang, and Calton Pu. 2014. LITMUS: Landslide detection by integrating multiple sources. Proc. of ISCRAM.

[11] Alexandra Olteanu, Sarah Vieweg, Carlos Castillo. 2015. What to Expect When the Unexpected Happens: Social Media Communications Across Crises. In Proceedings of the ACM 2015 Conference on Computer Supported Cooperative Work and Social Computing (CSCW '15). ACM.

[12] Viktor Pekar, Naveed Afzal, and Bernd Bohnet. 2014. UBham: Lexical Resources and Dependency Parsing for Aspect-Based Sentiment Analysis. In Proc. of the Eighth International Workshop on Semantic Evaluation (SemEval 2014).

[13] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes Twitter users: Real-time event detection by social sensors. In Proc. of WWW. ACM, 851–860.

[14] Vincent Schmidt and Jane Binner. 2011. A Semi-automated Display for Geotagged Text. Proceedings of the 2011 International Conference on Software Engineering Research and Practice.

[15] Fabrizio Sebastiani. Machine learning in automated text categorization. ACM Computing Surveys, 34(1):1-47, 2002.

[16] Tom De Smedt and Walter Daelemans. (2012). Pattern for Python. Journal of Machine Learning Research. 13, 2063-2067.

[17] Irina Temnikova, Carlos Castillo, and Sarah Vieweg. 2015. EMTerms 1.0: A Terminological Resource for Crisis Tweets. In Proceedings of the International Conference on Information Systems for Crisis Response and Management (ISCRAM'15).

[18] Sudha Verma, Sarah Vieweg, William J. Corvey, Leysia Palen, James H. Martin, Martha Palmer, Aaron Schram, and Kenneth Mark Anderson. 2011. Natural language processing to the rescue? Extracting "Situational Awareness" tweets during mass emergency. In Proc. of ICWSM.

[19] Jie Yin, Andrew Lampert, Mark Cameron, Bella Robinson, and Robert Power. 2012. Using social media to enhance emergency situation awareness. IEEE Intelligent Systems 27, 6, 52–59.

[20] Andrea Zielinski and Ulrich Buegel. 2012. Multilingual Analysis of Twitter News in Support of Mass Emergency Events. In of the 9th International ISCRAM Conference – Vancouver, Canada, April 2012.

Analysis of the Effect of Distance Metric across Languages on Verse Similarity in the Qur'an

Pan Huang[1], **Amna Basharat**[1], **and Khaled Rasheed**[1]
[1]Department of Comp. Sci, University of Georgia, Athens, GA, USA

Abstract—*Text similarity measures have been widely studied and used in machine learning and information retrieval for many years. However, few applications of text similarity have dealt with multi-lingual translations of a specific document. Additionally, the growing number of texts with more translations being generated increases the challenge of distinguishing or identifying the similarity and differences between texts across different documents. In this article, we employ different text similarity measures to delve into the problem of text similarity in the context of multi-lingual representations of the Qur'an. Four semantic translations of the Qur'an are used for comparative study and analysis. We compare and contrast the effect of applying five similarity measures across these representations. We analyze the results along two classes namely: identical verse pairs and similar verse pairs. Our analysis provides helpful observations about the impact of the five distance metrics for verse similarity in the Qur'an across different languages.*

Keywords: Similarity, Qur'an, Hamming, Manhattan, Jaccard.

1. Introduction

Text similarity measures and their applications have made a significant impact towards text related search and applications in tasks in many different fields including information retrieval, machine learning and semantic web [1]. Text is composed of words. If two words are constructed by similar characters with similar sequence, the words are considered to be lexically similar; if two words represent similar semantics, and are used in the same context, the two words are considered to be semantically similar [1].

In this article, we employ five suitable text similarity or distance measures to investigate the problem of analyzing similarity between the verses of the Qur'an. The Qur'an is one of the most widely read books. It is considered by Muslims as a single-authored text, the direct speech of God (Allah) [2]. The Qur'an contains 6236 verses and around 80000 words [3]. The original data format was spoken Classic Arabic [4]. Recently, Qur'an study has received increasing attention by many researchers, including the "Semantic Qur'an dataset" project described and utilized in [5] and the future research proposed by Eric and his colleagues about a structured large-scale online resource for understanding the Qur'an, which is called "Quranic Knowledge Map" [2]. However, little work has been devoted to the similarity study between verse texts on Qur'an. The QurSim introduced in [6] has some interesting findings, and includes a collection of semantically similar or related verses linked.

The text of the Qur'an presents an interesting dataset in terms of analyzing lexical and semantic similarity between the different verses. We previously undertook a study to compare the similarity of the verses in the Qur'an particularly across the different representations in the Arabic language and an English translation in [3]. We developed a Verse to Verse similarity computation framework for the Qur'an. We applied four similarity measures and three term weighting methods across different representations of the Qur'an. Specifically, we analyzed four datasets: 1) The original Arabic script of the Qur'an with diacritics or case markings, 2) The Arabic script without diacritics, 3) A dataset for the Arabic word roots in the Qur'an and 4) An English translation of the Qur'an. This study resulted in several interesting findings. Our analysis was largely limited to the identical verses in the Qur'an. However, it was evident that there are several verse pairs that are considerably similar and need further analysis. Also based on the analysis of the identical verses alone, it was concluded that such insights could indeed reflect upon the precision and accuracy of a given translation of the Qur'an. Thus, the study provided a basis for further analysis and investigation.

In this article, we extend our previous work to expand our investigation of the similarity amongst the verses of the Qur'an done in [3]. We keep three of the four best similarity computation measures from [3] and add two more measures, the Hamming distance and the Manhattan distance. These two measures are classified as term-based similarity measures according to the survey done by Wael and Aly in [1]. We do this in order to investigate the effect of distance measures on the similarity computation of the verses in the Qur'an in order to derive a conclusion about the most effective distance measure for computing the similarity. More importantly, we apply our similarity measures across four different representations of the Qur'an in three different languages. From the results of the study in [3], we found the Arabic representation without diacritics to give the most accurate results. We therefore use that as the baseline result for this study. We introduce one more English translation, in addition to the one we experimented with earlier, to give us better grounds for analyzing similarities across different translations in the same language. We also introduce another translation of the Qur'an, in the Urdu language, as one of the datasets, in order to analyze the similarities across different translations of the Qur'an in different languages. Research shows that similarity analysis within and across languages always obtains more interesting and valuable information [7]. We therefore aim to expand our insights about the similarity patterns in the Qur'an through this investigation. Moreover, the experiments done in [3] mainly focused on the analysis of identical verse pairs. In this article, we extend our investigation to similar verse pairs in the Qur'an and perform an in-depth comparison across the thr-

Fig1: Methodology Flow Chart

-ee different translations of the Qur'an.

2. Methodology

Table 1: Dataset sample and description

Abbreviation	Description	Example (first verse)
Q-Arabic	Arabic text without diacritics	"الرحيم الرحمن الله بسم"
Q-E-SAHIH	English text translated by the Saheeh team	"In the name of Allah the Entirely Merciful the Especially Merciful"
Q-E-YUSUF	English text translated by Yusuf Ali	"In the name of Allah\Most Gracious\ Most Merciful"
Q-Urdu	Urdu text	" شروع اللہ کا نام لے کر جو بڑا مہربان نہایت رحم والا ہے"

This section presents the methodology introduced in [3] for implementing the process of verse similarity evaluation for the different representations of the Qur'an. We also describe the five different similarity measures that we implemented for this study.

2.1 Dataset with different representations of the Qur'an

A Qur'an database was created for this study which contains four different representations of the Qur'an. Table 1 shows the information of the four representations and a sample text in each representation of the Qur'an.

2.2 Qur'an text preprocessing

Each dataset is complete raw data without any preprocessing; each text is simple and plain. Figure 1 shows the framework that has been designed in [3]. The preprocessing step removes all the stop words and punctuations for each verse in the corresponding representation of the Qur'an.

2.3 Feature selection and verse-vector representation

According to our previous work in [3], assume that V = $\{v_1, v_2, v_3 \ldots v_n\}$ represents the set of verses in the Qur'an, and T = $\{t_1, t_2, \ldots t_m\}$ represents the set of the unique terms that construct each v in V. Each verse v is considered as an m-dimensional vector $\overline{v}$= $\{a_1, a_2, a_3 \ldots a_m\}$, where a_i represents the weight of the i_{th} term in the vector $\overline{v}$. The next step is to generate the verse-vector matrix. In the matrix, each row represents each verse of the Qur'an and each column represents a unique term of the vocabulary which constitutes the verses in Qur'an. The verse-vector matrix is structured based on the verse order from the Qur'an, and each element is calculated in accordance with one of two term weighting techniques: term frequency–inverse document frequency (TF-IDF) or Frequency (F). We keep these two out of the three term weighing techniques from [3]. At first, the verse set is processed to select the features. After that, the verse-vector matrix where each verse is represented as weighted vector is generated based on the features.

2.4 Similarity computation and similarity analysis

In the framework, the similarity computation is the most important step of our experiment. We use the similarity computation module to analyze the correlation between verses by calculating the distance or similarity of the corresponding vectors. More importantly, selecting an appropriate similarity computation measure is the most crucial part. The different vector representation also affects the similarity accuracy. Therefore, we believe that in order to obtain the best result, each combination of similarity measure and vector representation (Frequency and TF-IDF) should be applied to the verses in the Qur'an. Three similarity measures are described in [8] and [9], and two of them are described in [10] and [11] respectively. Table 2 shows the relevant equation and brief description for each similarity measure. In the equations, $\overline{v}_a$ and $\overline{v}_b$ are the term vectors corresponding to the two verses v_a and v_b respectively. T = $\{t_1, t_2, \ldots t_m\}$ represents the weight of each term occurring in V.

The similarity computation step generates the verse to verse similarity matrix which contains all the similarity values of each verse pair. The similarity analysis module extracts the relevant similarity values from the matrix for analyzing the identical and similar verse pairs.

3. Experimentation and results

3.1 Evaluation measures for identical verses

The Q-Arabic dataset contains 775 identical verse pairs. Those verse pairs serve as the ground truth for our identical verse evaluation across the different translations. Each similarity ranges from 0 to 1. A value of 1 means the two verses are lexically identical. Our analysis first focuses on the verse pairs with similarity value 1.

For our analysis we still use the same evaluation metrics used in our previous work [3], including Precision, Recall and F1score. Equations (1-3) show the formulae of those three measures.

$$P(Precision) = \frac{TP}{TP + FP} \quad (1)$$

$$R(Recall) = \frac{TP}{TP + FN} \quad (2)$$

$$F1(F - Measure) = \frac{2 \times P \times R}{P + R} \quad (3)$$

TP represents true positive, TN represents true negative, FP represents false positive, FN represents false

Table 2: Equations and brief description for 5 different similarity measures

Similarity Measures	Equations and brief description
Cosine Similarity: measures the similarity between two texts by obtaining their normalized dot product. The range of Cosine similarity is bounded in [0, 1].	$S_C(\vec{v}_a, \vec{v}_b) = \frac{\vec{v}_a * \vec{v}_b}{\lvert\vec{v}_a\rvert * \lvert\vec{v}_b\rvert}$
Manhattan Distance: computes the sum of difference in each dimension of two vectors in n dimensional vector space. It is the sum of the absolute differences of their corresponding components [10]. We invert the distance as $1/D_m$ to form the value bounded in [0, 1]	$D_M(\vec{v}_a, \vec{v}_b) = \sum_{i=1}^{n} \lvert t_{ai} - t_{bi}\rvert$
Pearson Correlation: computes the linear correlation between two objects. It obtains the correlation coefficient by computing the ratio of the covariance of the two objects and the product of their standard deviations. The value is also bounded in [0, 1].	$S_P(\vec{v}_a, \vec{v}_b) = \frac{m\sum_1^m t_{a1} \times t_{b1} - \sum_1^m t_{a1} \times \sum_1^m t_{b1}}{\sqrt{(m\sum_1^m (t_{a1})^2 - (\sum_1^m t_{a1})^2) \times (m\sum_1^m (t_{b1})^2 - (\sum_1^m t_{b1})^2)}}$
Hamming Distance: computes the minimum number of substitutions from one string changes to another string. For example: The Hamming distance between 1234567 and 1233497 is 3. We inverse the distance as 1/D_H to form the value bounded in [0, 1].	$D_H(\vec{v}_a, \vec{v}_b) = \sum_{i=1}^{n} \lvert t_{ai} - t_{bi}\rvert$ (if t_{ai} is different from t_{bi}, $t_{ai} - t_{bi} = 1$)
Jaccard Similarity: computes the similarity between sets. This method is defined as the quotient between the intersection and the union of the entities. The above equation is modified based on the general definition for different cases. The value range is in [0, 1].	$s_J(\vec{v}_a, \vec{v}_b) = \frac{\vec{v}_a * \vec{v}_b}{\lvert\vec{v}_a\rvert^2 + \lvert\vec{v}_b\rvert^2 - \vec{v}_a * \vec{v}_b}$

negative. In order to obtain the above three measures, we treat verse pairs with similarity value 1 which are identical verses in the Qur'an as TPs which are correctly classified by the model. We treat non-identical verse pairs with similarity value that is not equal 1 as TNs. FPs are non-identical verse pairs classified as identical pairs (value 1). FNs are identical verse pairs classified as non-identical pairs (value less than 1).

3.2 Experimental results

In our experiments, four datasets including one original Arabic script and three translations of the Qur'an are analyzed. We apply five similarity measures to these datasets. In addition, two term weighting techniques have been used. Therefore, 4 * 5 * 2 = 40 combinations in total are implemented. Every experiment scheme is defined by an abbreviation as follows: dataset representation used - term weighting technique applied - similarity measure employed. For instance, *Q-Urdu-M-F* indicates *Manhattan* distance measure is implemented on the *Q-Urdu* dataset, and the term weighting technique is *Frequency*. All the experimental results are shown in Table 3.

3.3 Analysis for identical verses

The Arabic text of the Qur'an generates the most perfect experimental results when focusing on the identical verse pairs' similarity. This is clearly indicated by the precision and the recall values given in Table 3. In our previous study [3] we analyzed three different representations of the Arabic Qur'an namely: Arabic script with diacritics (case markings), an Arabic script without diacritics and a dataset that includes the Arabic word roots. We also included the Q-E-YUSUF in our comparative analysis. We presented an in-depth analysis for the identical verse pairs in this study. From this study, the Arabic script without diacritics provided the best results.

Therefore, the results obtained from this dataset were chosen to be the baseline results for the experimentation and results presented in this article. We focus our analysis on the relative comparison between the two English translations: Q-E-SAHIH and Q-E-YUSUF and the Urdu translation of the Qur'an (Q-E-URDU). We also provide a comparative analysis across these different representations.

3.3.1 Analysis on Q-E-YUSUF and Q-E-SAHIH and comparison between them

The Q-E-SAHIH and Q-E-YUSUF representations are two different translations of the Qur'an, translated by different English translators. Compared to the original Arabic representation, both produce a relatively lower recall of around 93% for Q-E-SAHIH representation and 85% for Q-E-YUSUF representation.

From the table, we find that both representations generate a considerable number of FN (false negatives), which is a significant indicator when it comes to the identical verses [3]. As we have stated above, in terms of our experiments, false negatives represent identical verse pairs classified as non-identical. In the Q-E-YUSUF representation, the FNs reach up to 114 or 115, and it is exactly twice the FNs of the Q-E-SAHIH representation. That can be considered as a significant indicator to establish that the Q-E-SAHIH's translation quality is closer in terms of accuracy and precision to the original Arabic script, as compared to the Q-E-YUSUF. Finding the number of FNs and comparing it with the most original Arabic Qur'an script is one of the measures to evaluate the translation quality.

We found some verse pairs that are translated differently even though they are identical in the Arabic Qur'an script. Verse 3:182 and verse 8:51 are identical verses in the Q-Arabic representation. However, these two verses are appearing slightly different in the two English translations. In the Q-E-SAHIH translation, compared to verse 3:182, verse 8:51 has two extra words "of" and "devil" in the middle of the verse, which causes these two verses to be detected as non-identical by our framework. Likewise, in the Q-E-YUSUF translation, semantically, the verses describe the same content, however, the structure and the length of the verse is slightly different. Besides this sample case, we also found another two verses, which are translated identically in one of the English translations, but differently in the other one. Verse 54:17 and verse 54:22 are identical verses in the Q-Arabic representation. The Q-E-SAHIH translated these two verses to identical English verses. However, in the Q-E-YUSUF, the corresponding translated verses are not the same because the first initial is "and" in the verse 54:17, however, it is the word "but" in the verse "54:22".

Table 3: Experimental results

		Cosine(C)		Jaccard(J)		Pearson(P)		Manhattan(M)		Hamming(H)	
		TF-IDF	F	TF-IDF	F	TF-IDF	F	TF-IDF	F	TF-IDF	F
Q-ARABIC	**TP**	775	775	775	775	775	775	775	775	775	775
	FP	2	1	2	1	2	1	2	1	2	1
	FN	0	0	0	0	0	0	0	0	0	0
	P	0.997	0.999	0.997	0.999	0.997	0.999	0.997	0.999	0.997	0.999
	R	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
Q-E-SAHIH	**TP**	718	718	718	718	718	718	718	718	718	718
	FP	12	11	12	11	12	11	12	11	12	11
	FN	57	57	57	57	57	57	57	57	57	57
	P	0.985	0.985	0.985	0.985	0.985	0.984	0.985	0.985	0.985	0.985
	R	0.926	0.926	0.926	0.926	0.926	0.926	0.926	0.926	0.926	0.926
	F	0.955	0.955	0.955	0.955	0.955	0.954	0.955	0.955	0.955	0.955
Q-E-YUSUF	**TP**	661	660	661	660	661	660	661	660	661	660
	FP	12	10	12	10	12	10	12	10	12	10
	FN	114	115	114	115	114	115	114	115	114	115
	P	0.982	0.985	0.982	0.985	0.982	0.985	0.982	0.985	0.982	0.985
	R	0.853	0.852	0.853	0.852	0.853	0.852	0.853	0.852	0.853	0.852
	F	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913
Q-URDU	**TP**	535	535	535	535	535	535	535	535	535	535
	FP	7	6	7	6	7	6	7	6	7	6
	FN	240	240	240	240	240	240	240	240	240	240
	P	0.987	0.989	0.987	0.989	0.987	0.989	0.987	0.989	0.987	0.989
	R	0.690	0.690	0.690	0.690	0.690	0.690	0.690	0.690	0.690	0.690
	F	0.812	0.813	0.812	0.813	0.812	0.813	0.812	0.813	0.812	0.813

Similarly, verse 7:108 and 26: 33 are translated to the same English verses in Q-E-YUSUF, but to different English verses in Q-E-SAHIH with two extra words occurring in verse 26:33.

FP (false positives) is another significant indicator to analyze the identical verses. From the perspective of our experiments, false positive cases represent non-identical verse pairs which are classified as identical. In the result table, with different term weighting techniques, 12 or 10 FPs are obtained in the Q-E-YUSUF and 12 or 11 FPs in the Q-E--SAHIH. Both versions contain much fewer FPs as compared to the number of FNs. Arabic is a rich morphological language; in particular, the language of the Qur'an is considered to be precise and the slightest variation or alteration in the arrangement or morphological manifestation of the word implies something significant, which the translation often fails to capture [3]. Table 4 shows three cases, which reflect the difference for specific verse pairs among the three representations of the Qur'an. Case 1 demonstrates two verses which are not identical in the original Arabic script but are translated identically in the two English Qur'an translations. The reason for the difference is that the word "إنا" is widely used in Arabic as an intensifier to emphasize the speech. Verse 37:110 includes this word, but it is missing in verse 37:121. However, this difference is not captured by the two English translations. Similarly, Case 3 shows another two non-identical verses in Arabic. The word "و" is included in verse 20:9, but is not appearing in verse 79:15. This character is a conjunction word in Arabic referring to "and" in English. As a result, the Q-E-SAHIH translation correctly captures this difference; however, it has been ignored in the Q-E-YUSUF. Case 2 shows two verses constructed with same words in the Arabic Qur'an, but three of these words are in different orders within the two verses, which results in the two verses not being lexically identical. This is captured by Q-E-YUSUF translation, but not by the Q-E-SAHIH translation.

For our experiments, we intentionally used two versions of English translation in order to illustrate that the various versions of the Qur'an translated in one language will invariably have subtle differences. In addition, this analysis also proves that our framework is helpful in evaluating the translation's quality and investigating the comparison and contrast among different versions of the Quran that are translated in one language.

3.3.2 Analysis of identical verses in Q-URDU

The result table shows that the Q-URDU representation obtains a recall of around 69%, which is much lower than the other three representations. Besides that, the F1 score is also the lowest compared to the others. Similarly, this representation generates both FNs and FPs. The number of FNs is almost one third of the number of identical verse pairs. On the other hand, the number of FPs is smaller than the numbers generated by the two English representations.

For the Q-URDU representation, 240 identical verse pairs are classified mistakenly as non-identical. 240 is a big number in terms of the influence of FNs reflecting the relative precision or accuracy of the translation relative to the original script. Within the 240 false negatives, most verse pairs have a fairly common small difference which causes the misclassification. For instance, verses 5:10 and 5:86 are identical in the Arabic script but not identical in the Urdu version. The difference is strongly visible, where the word "جنہوں" in verse 5:10 is different from the " جن لوگوں" in verse 5:86. However, "جنہوں" is another way of expressing "جن لوگوں", so basically this change doesn't affect the semantics of the verse. It indicates that the translator has just expressed the same meaning with a different expression. Another representative verse pair is indicated by the verse 7:121 and the verse 26:47. There are a few differences between them. First, the word "تمام" is absent in verse 7:121, and present in verse 26:47, but this is a minor difference. Second, the word "پروردگار" used in verse 7:121 means Lord or God which accurately captures the Arabic meaning; the word "مالک " used in ver-

Table 4: 3 cases for analyzing the false positives on three Qur'an representations

Case 1	**Verse Text(Q-Arabic)**	**Verse Text(Q-E-SAHIH)**	**Verse Text(Q-E-YUSUF)**
37:110	"إنا كذلك نجزي المحسنين"	"Indeed We thus reward the doers of good"	"Thus indeed do We reward those who do right."
37:121	"كذلك نجزي المحسنين"	"Indeed We thus reward the doers of good"	"Thus indeed do We reward those who do right."
Case 2	**Verse Text(Q-Arabic)**	**Verse Text(Q-E-SAHIH)**	**Verse Text(Q-E-YUSUF)**
23:83	"لقد وعدنا نحن وآباؤنا هذا من قبل إن هذا إلا أساطير الأولين"	"We have been promised this we and our forefathers before this is not but legends of the former peoples"	"Such things have been promised to us and to our fathers before! they are nothing but tales of the ancients! "
27:68	"لقد وعدنا هذا نحن وآباؤنا من قبل إن هذا إلا أساطير الأولين"	"We have been promised this we and our forefathers before this is not but legends of the former peoples"	"It is true we were promised this we and our fathers before (us): these are nothing but tales of the ancients."
Case 3	**Verse Text(Q-Arabic)**	**Verse Text(Q-E-SAHIH)**	**Verse Text(Q-E-YUSUF)**
20:9	"وهل أتاك حديث موسى"	"And has the story of Moses reached you"	"Has the story of Moses reached thee?"
79:15	"هل أتاك حديث موسى"	"Has there reached you the story of Moses"	"Has the story of Moses reached thee?"

-se 26:47 means King or owner which is less accurate. Nevertheless, the semantic content of both verses is still same. Third, the word "لائے " and "آئے لے " are separately used in two verses. Again, these two words are generally still meaning the same thing in the two verses. From the analysis of the 240 FNs, we can deduce that the differences in translation of identical verses occur for two main reasons. Firstly, Urdu has a tendency to be a more verbose language compared to Arabic. The same verse expressed using few words in Arabic requires more words in Urdu in order to convey the meaning. Also, the choice of prepositional and conjunction words can cause lexical differences in the expression, while maintaining the semantics. The second reason is that of using a different synonym for a significant word, which may imply subtle semantic differences. This may also be reflective of the precision of the translation. Whether identical verse pairs ought to be translated differently or not is a question in itself.

Table 5: 2 Cases for analyzing the false positives from Q-URDU

Case 1	**Verse Text(Q-Arabic)**	**Verse Text(Q-Urdu)**
37:34	"إنا كذلك نفعل بالمجرمين"	"ہم گنہگاروں کے ساتھ ایسا ہی کیا کرتے ہیں"
77:18	"كذلك نفعل بالمجرمين"	"ہم گنہگاروں کے ساتھ ایسا ہی کیا کرتے ہیں"
Case 2	**Verse Text(Q-Arabic)**	**Verse Text(Q-Urdu)**
52:11	"فويل يومئذ للمكذبين"	"اس دن جھٹلانے والوں کے لئے خرابی ہے"
77:15	"ويل يومئذ للمكذبين"	"اس دن جھٹلانے والوں کے لئے خرابی ہے"

As we can see from the table, the Q-Urdu representation also generates some FPs (i.e. verse pairs which are different in the original Arabic script but found to be identical in Q-Urdu). Table 5 shows two such FP cases. In case 1, there is an additional word at the beginning of verse 37:34 in Arabic, which is present for emphasis. However, this additional emphasis is not reflected in the translation for both verses. Case 2 shows a rather subtler difference of a single character at the beginning of verse 52:11. However, it should have been captured by the translator, but was not. The analysis of the FPs provides a good basis for determining cases where subtle differences in the Arabic script are not reflected in the translation.

3.3.3 Comparative analysis across different languages

We considered an additional thread of analysis across the three different translations of the Qur'an by finding common FNs or FPs among them. Table 6 shows the number of common FNs or FPs between two or three different translations of the Qur'an. For the convenience of analysis, all the statistics from Table 6 are based on the FNs and FPs which are generated by the Cosine method with frequency as term weighing measure. Since the generated FPs from each translation are few, as shown in Table 6, the numbers of common FPs between various translations are much smaller than FNs. Between the two English translations, there are 3 verse pairs which are FPs for both translations. They are 37:80&37:110, 37:110&37:121 and 37:110&37:131. If we look more closely at these verse pairs, all of them involve the same verse 37:110. The original verse 37:110 is composed of 3 simple Arabic words. However, the other 3 verses which are similar to verse 37:110 include one extra emphasis word "إنا" at the beginning of the verse in the Arabic Qur'an script. Since this difference is not reflected in the English translations, the 3 verse pairs become false positives in both Q-E-SAHIH and Q-E-YUSUF. Interestingly, the only one common FP between all three translations is 37:80&37:110. However, the difference in the other two verse pairs is captured by the Urdu translation. It is not clear why that difference is ignored in one of the verse pairs in the Urdu translation. It is nevertheless obvious that the number of false positives falls within a small range. The major reason that the false negatives still exist is because the translator more or less ignored the emphasis word from some original Arabic verses.

Table 6: The FPs and FNs in common between relevant translations.

Comparative Cases	**FPs in common**	**FNs in common**
1. Q-E-YUSUF vs Q-E-SAHIH	3	46
2. Q-E-YUSUF vs Q-URDU	1	99
3. Q-E-SAHIH vs Q-URDU	1	56
4.Q-E-YUSUF vs Q-E-SAHIH vs Q-URDU	1	45

Indeed, each translation of the Qur'an generated many

FNs, especially in the Q-URDU. In case 1, the common FNs are 46 which is almost 80% of FNs of Q-E-SAHIH. There are another 11 FNs unique to Q-E-SAHIH and 69 FNs unique to Q-E-YUSUF. In case 2, the common FNs are 99, which is almost 86% of FNs of Q-E-YUSUF. There are another 16 unique FNs to Q-E-SAHIH and 141 unique to Urdu Qur'an. In case 3, the common FNs are 56; the FNs of Urdu include all the FNs generated by Q-E-SAHIH. From these three cases, we may draw at least one conclusion that there are many FNs in common between the different translations. In other words, a certain number of identical verse pairs in the original Qur'an are consistently translated as different verse pairs by the translators with different reasons. The case 4 is an example, 45 common FNs among three translations. Compare this number with the number in case 1, only one difference, which means there is one identical verse pair correctly captured by the translators of Urdu Qur'an, but not captured by the translators of both English translations. One of the reasons behind this could be the criteria of translation set up by different language. Therefore, based on case 4, if we add one more translation of the Qur'an, the number of common FNs may be fewer than 45 because of the new criteria of the additional translation.

Finally, if we evaluate the quality of a translation of the Qur'an based on recall, F1 score or the number of FNs or FPs without considering the language itself and the criteria of translation set up by different languages, the Q-E-SAHIH definitely is the closest to the Arabic Qur'an based on the comparative analysis on identical verse pairs.

3.4 Analysis for similar verses

Table 7: Selected verses for similarity analysis

Verses	Verse Text (Q-Arabic)
1:3	"الرحمن الرحيم"
1:1	"بسم الله الرحمن الرحيم"
27:30	"إنه من سليمان وإنه بسم الله الرحمن الرحيم"
11:61	وإلى ثمود أخاهم صالحا ۚ قال يا قوم اعبدوا الله ما لكم من إله غيره ۖ هو أنشأكم من الأرض واستعمركم فيها فاستغفروه ثم توبوا إليه ۚ إن ربي قريب مجيب
11:84	وإلى مدين أخاهم شعيبا ۚ قال يا قوم اعبدوا الله ما لكم من إله غيره ۖ ولا تنقصوا المكيال والميزان ۚ إني أراكم بخير وإني أخاف عليكم عذاب يوم محيط

Although, much of our analysis is based on the identical verse pairs in the Qur'an, these pairs are a handful compared to the total number of verse pairs. There are several verse pairs, which are similar to a great extent but not identical. It is therefore worthwhile to explore the similarities of those verse pairs. We aim to establish the effectiveness of the various similarity measures employed in our experiments for the analysis of similar verses across the difference representations of the Qur'an.

One of the challenges with analyzing similar verse pairs, unlike the identical verse pairs, is the lack of any baseline standard similarity values or thresholds. It is therefore impossible to investigate and analyze all similar verse pairs; instead, we manually selected a few representative verses to study from the Arabic Qur'an. As shown in Table 7, it is clear that the first three verses are similar to each other. Specifically, verse 1:1 contains all the content of verse 1:3, and verse 1:27 contains all the content of verse 1:1. The last two verses lexically contain a portion of common words. In looking at the first three verses from Table 7, we clearly see that the content of verse 1:3 is around half that of verse 1:1 and around a quarter of that of verse 1:27; verse 1:1 is also around half of verse 1:27. Table 8 shows relevant similarity values obtained for these verse pairs across the different representations of the Qur'an. For the Q-Arabic version, the Jaccard method gives the most logical similarity values with both F and TFIDF term weighing measures for these three pairs. Since the Jaccard method is one of the term-based similarity measures, it is not surprising for it to return the best results. We also studied the other four methods, and compared the relevant similarity values to that of Jaccard's. The Hamming method returns the same results with the two different term weighting measures, which are close to the Jaccard's evaluations, especially on verse pair 1:1 and 27:30. The Manhattan method obtained the same results as the Hamming's with Frequency as weighting measure and different results with TFIDF as the weighting measure, but overall, the results still approach the Jaccard's values. On the other hand, the results obtained with the other two weighting measures, Pearson and Cosine, are not reflecting the observed similarity values.

The values obtained for these two methods vastly exceed the Jaccard's values. Contrary to the first three verses in Table 7, the last two verses do not contain only one contiguous common portion but rather a few common portions. Therefore, it cannot be intuitively evaluated. However, after analyzing the first 3 verses, we conclude that Jaccard method with Frequency as the term weighing measure is the most suitable combination to evaluate our cases. Look into this pair, Jaccard with F returns 0.28, which we consider intuitively close to the true value. Among the other combinations, only Cosine method with TFIDF and Jaccard method with TFIDF are close to this result, so we conclude that the rest of the combinations do not reflect the true similarities. To sum up, the Jaccard method with F as the weighting measure can objectively and appropriately reflect the intuitive values of all cases based on our selected verses. The Manhattan, Hamming and Cosine methods are applicable to some cases. As for the Pearson method, the overall evaluation results are much higher than any intuitive reasonable values.

As for the other language representations, for the first three verses, the relevant similarity values generated by the Jaccard method with F in the Q-E-SAHIH are most similar to that of Arabic. The relevant similarity values from the other Qur'an representations also reflect the same relationships among these three verses. On the other hand, for verses 11:61 and 11:84, the relevant similarity values are 0.37, 0.4, and 0.35 respectively from the three representations, which are all close. Therefore, from what has been discussed above, considering the similarity value comparison of the given cases, we may reasonably conclude that the translation accuracy for similar verses of Q-E-SAHIH is better than that of other representations. Also, its translation quality is the best among the three translations. Overall, the pattern of similarity is maintained in the other three representations of the Qur'an, which are based on translations. This pattern follows the similarity patterns of the original Arabic script. However, determining the extent to which this conclusion can be generalized needs further investigation and validation of the ground truth measures.

Table 8: Relevant similarity values for selected verse pairs

Q-ARABIC	Manhattan		Hamming		Jaccard		Pearson		Cosine	
	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF
1:3 VS 1:1	0.66	0.38	0.66	0.66	0.5	0.53	0.85	0.84	0.71	0.68
1:3 VS 27:30	0.39	0.23	0.39	0.39	0.25	0.28	0.75	0.75	0.5	0.51
1:1 VS 27:30	0.48	0.28	0.48	0.48	0.5	0.53	0.85	0.87	0.71	0.75
11:61 VS 11:84	0.18	0.09	0.18	0.18	0.28	0.21	0.73	0.64	0.46	0.29
Q-E-YUSUF	Manhattan		Hamming		Jaccard		Pearson		Cosine	
	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF
1:3 VS 1:1	0.43	0.40	0.43	0.43	0.18	0.58	0.89	0.89	0.74	0.79
1:3 VS 27:30	0.27	0.22	0.28	0.28	0.24	0.30	0.77	0.77	0.53	0.54
1:1 VS 27:30	0.34	0.28	0.36	0.36	0.53	0.51	0.84	0.84	0.72	0.68
11:61 VS 11:84	0.14	0.09	0.14	0.15	0.37	0.26	0.64	0.64	0.62	0.29
Q-E-SAHIH	Manhattan		Hamming		Jaccard		Pearson		Cosine	
	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF
1:3 VS 1:1	0.43	0.40	0.43	0.48	0.55	0.66	0.95	0.93	0.87	0.85
1:3 VS 27:30	0.26	0.22	0.28	0.29	0.3	0.36	0.84	0.80	0.67	0.61
1:1 VS 27:30	0.32	0.26	0.36	0.36	0.55	0.55	0.89	0.86	0.77	0.71
11:61 VS 11:84	0.15	0.10	0.12	0.17	0.40	0.27	0.84	0.64	0.68	0.28
Q-URDU	Manhattan		Hamming		Jaccard		Pearson		Cosine	
	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF	F	TFIDF
1:3 VS 1:1	0.34	0.24	0.34	0.34	0.38	0.46	0.81	0.84	0.62	0.68
1:3 VS 27:30	0.22	0.18	0.24	0.24	0.21	0.33	0.70	0.78	0.41	0.57
1:1 VS 27:30	0.27	0.22	0.28	0.29	0.48	0.59	0.85	0.86	0.71	0.73
11:61 VS 11:84	0.12	0.08	0.11	0.14	0.35	0.24	0.82	0.64	0.63	0.29

4. Conclusion and future work

In this article, we investigated the effect of five similarity measures across four different representations of the Qur'an, in three different languages. We analyzed the results for identical and similar pairs of verses within these representations. We concluded that the Q-E-SAHIH representation demonstrates the most accurate result with highest F1 score among the different translations of the Qur'an. We also concluded that the Jaccard similarity method proves to be effective for each of our tested verse pairs. In addition, the similarity values returned by the Jaccard measure intuitively reflect the observed similarities. We also found that the verse pairs which are similar in the original Arabic script are more or less lexically similar in the three translations.

Regarding future work, this research can be improved and extended in various aspects. First, we can apply the framework developed as part of our research for analyzing similarities within and across other religious texts such as the Bible, to see how the approach scales for larger texts. Second, we aim to adopt more term weighting methods for our future experiments, such as LTC, and relative frequency which are introduced in [12]. Finally, we plan to reduce the time complexity of future experiments by applying more efficient data structures or algorithms to reconstruct our term weighing matrix in the data preprocessing step, or the whole process, using parallel processing techniques.

References

[1] W. H. Gomaa, A. A. Fahmy, "A survey of text similarity approaches", International Journal of Computer Applications(0975-8887) vol. 68 -No.13 pp. 01-06, April 2013.

[2] 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.

[3] A. Basharat, D. Yasdansepas, and K. Rasheed, " Comparative study of verse vimilarity for multi-lingual representations of the qur'an", Proceedings on the International Conference on Artificial Intelligence (ICAI), 2015.

[4] 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.

[5] M. A. Sherif and A.-C. Ngonga Ngomo, "Semantic quran - a multilingual resource for natural-language processing," Semantic Web, 2009.

[6] A.-B. M. Sharaf and E. Atwell, "Qursim: A corpus for evaluation of relatedness in short texts." in LREC, 2012, pp. 2295–2302.

[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] A. Huang, "Similarity measures for text document clustering," in Pro- ceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand, 2008, pp.49-56.

[9] 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.

[10] Hasnat, S. Halder, A. Hoque, D. Bhattacharjee, M. Nasipuri, "A fast fpga based architecture for measuring the distance between tow color images using manhattan", International Journal of Electronics and Communication Engineering & Technology(IJECET), vol. 4, pp. 01–10, May -June, 2013.

[11] A. Bookstein, V. A. Kuliukin, T. Raita, " Generalized Hamming distance", Information Retrieval Journal, Vol 5, pp.353-375, Octorber, 2012.

[12] 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. 513-5

Using Artificial Intelligence to Automatically Customize Modern Car Infotainment Systems

Ashraf Gaffar and Shokoufe Monjezi
School of Computing, Informatics, & Decision Systems Engineering
Ira A. Fulton Schools of Engineering
Arizona State University, Phoenix, AZ, USA

Abstract –*In the last decade the automotive industry added digital features to their new cars, including advanced interactive infotainment system. Drivers execute several secondary tasks simultaneously besides the essential driving tasks. More complex infotainment system needs more cognitive attention causing more driver distraction. A consensus from multiple driver distraction research is that multi-tasking causes impaired performance. Although drivers know that distraction is a major reason of car accidents, they continue doing distracting activities during driving. In this paper we introduce a customized user interface instead of a standard user interface to reduce response time and hence driver distraction. While it is hard to directly measure cognitive-related distraction, events that take longer response time by the driver indicate a higher cognitive load as the user needs more time to perceive, analyze and comprehend when compared to simple events. Our hypothesis is that a reduction in the driver's response time while using infotainment system can indicate a reduction in driver distraction*

Keywords: human car interaction, response time, driver distraction, cognitive load, cognitive distraction

1 Introduction

The primary task of driving can be challenging for the drivers and requires full attention and focus on the road, continuously watching other cars, street and traffic signs, and pedestrians. Drivers also need to keep an eye on their own car's instrument panel as well as side and rear views. Additional tasks include operating necessary car controls to turn on lights, windshield wipers or turn signals. While not essential for proper car operation, driver entertainment was added early on due to reduce the boredom of driving. A simple car radio evolved rapidly in features then a CD was added, and eventually a complete digital system with advanced screen display was emerging. As modern cars become predominantly digital in most of its systems, with a lot of information available, industry needed to communicate some of this information to the driver using an information system. Combining this "**Info**rmation" system with a complex enter**tainment** system and a significantly larger screen, it is now commonly known as the "**Infotainment** System, occupying the mid console of many cars.

In the last few years, the infotainment system of modern cars saw a rapid growth in complexity as manufacturers kept adding advanced interactive ditgital features to the cockpit [1]. With the avilability of a large screen, a full computing architecture slimila to Von Neuman maching, including multi-core microporcessors, memory, data bus, I/O devices and advanced networking capabilitiy as well as an operating system and software, the infotainment system opened the door to bringing virtually every application that was originally developed for the desktop computing environment into the car cockpit.

This migration was seen before as the same trend took place with smart phones evolving into an advanced computing platform with similar features as a desktop. We call this the "First Digital Migration Wave, Wave I". The "Second Digital Migration Wave, Wave II" is now happening as the infotainment system is evloving in the same way. Unfortunately, the context of use in both waves is dramatically different. While users can generall afford to have full focus on their smartphone screens, with few exceptions, drivers can not. This Wave II evolutionary trend is therefore causing complications not seen in Wave I.

Besides the essential driving tasks, some drivers execute several secondary tasks such as texting, listening to the music or checking email simultaneously. These secondary tasks need cognitive attention while some of them also need visual or physical attention, requiring drivers to take their eyes off the road, or their hands off the wheel, causeing distraction. Modern cars with advanced infotainment system often need more cognitive attention, causing more distraction. A growing trend in the research community is to find better ways to manage distraction by improving the interaction between the car and the driver using design principles from Natural User Interfaces (NUI) and Human Computer interaction (HCI) [1]. NUI systems have to adapt to the human behind the wheel instead of forcing people to adapt to the system. Although some of these systems are less

cognitively demanding, they all cause driver distraction [2]. Driver distraction is broadly defined as "The diversion of attention away from activities critical for safe driving toward competing for activities" [3].

Several known brain areas are activated during driving-related tasks such as visual area, auditory, and spatial working memory. Theta is a brain wave that has a direct correlation with increase drivers errors. When the amount of theta increases in any brain's areas the result is more driver distraction and more errors during driving. Using devices such as mobile phone increases theta activity so it causes more driver distraction [4][5].

According to fatal accidents report issued by NHTSA [6], in-vehicle distraction was an important factor in fatal accidents from 1985 to 1995 in England. National Highway Traffic Safety Administration (NHTSA), estimates that in 25% of accidents in the United State of America driver distraction is the main reason of accident. This means 1.2 million incidents each year happen because of driver distraction.

Four kinds of distraction were identified by NHTSA [7]:

I. **Visual Distraction:** Happens when the driver takes his eyes off the road, for example when reading text messages

II. **Manual Distraction:** Happens when the driver's one or two hands are not on the steering wheel

III. **Audio Distraction:** Any sound such as music, engine's sound and passengers' conversation cause audio distraction.

IV. **Cognitive Distraction:** Anything that absorbs thoughts and reduces the driver's attention.

In general, these four kinds of distractions rarely happen in isolation. A deeper analysis of distraction allows us to provide a simple model with specific attributes and relationships.

Composition: Most of the time, distraction takes place as a combination of two or more of these four types happening together. We call this the "composite" distraction.

Association: Visual, Manual, and Audio distraction is often associated with cognitive distraction. This "Association" can be weak as in the case when a noise is common to the driver. The car's own engine noise would have weak association to cognitive distraction as the driver is probably used to it. However, a strange or a sudden sound would have a strong cognitive association as the driver tries to think if that was a blown tire, a broken muffler, or something else.

Rank: Driving tasks are inherently distracting. Even a perfect driver cannot focus on the road ahead 100% of the time. A good driver needs to distribute attention wisely between the road, the car instrument panel, the management of car driving-related controls (gas and break pedals), utility controls (lights, turn signals, wipers, windows…), as well as the side view and the rearview mirrors. We call these items to cause "*essential distraction*". Other distractions like interaction with a navigation system, radio/CD, or managing additional information or features on the infotainment system or other interactive gadgets in the cockpit are considered non-essential or "*dispensable distractions*".

There is clearly a complex and intertwined relationship between tens of distractors and their effect on driver distraction. What makes it harder to come up with an objective evaluation is that the driving activity inherently has varying demand of user attention, and rarely requires 100% attention 100% of the time. At certain moments, drivers can partially distribute their attention between different tasks with no harm. In other driving situations, the same behavior can lead to a major accident.

The common conclusion is that multi–tasking during driving a car causes impaired performance. The researcher found that doing second activity during driving causes a competition for cognitive resources among different tasks, the result of this competition is driver distraction. For example listening to sentences while driving reduces the spatial processing by 37% [8].

Although driver distraction is a major contributor to car accidents, car drivers continue doing distracting activities. For example, they regularely use mobile phones, navigation or other features of infotainment system [2]. 75% of drivers believe that hands-free devices are safe to use during driving, they believe this kind of devices doesn't make distraction but according to the American Automobile Association (AAA) they have a false sense of security [8].

The basis for all distraction detection algorithms is measured and registered during driving. They include driver behaviour (e.g eye movement), driving behaviour (e.g speed) and other data. After collecting these data, some features such as gaze direction are extracted. Driver distraction detection algorithms use these data to estimate driver distraction. For example, "glance away" from the road is a glance put outside the place relevant to driving. When the driver looking away from the road for too long or too often the driver is distracted [2].

An experiment was made by AAA to measure cognitive distraction. In this experiment, a 'one' rating means there isn no distraction and 'five' means workload is a maximum workload that a human can handle, indicating maximum distraction. They found out that even if the perfect

technology is used in car's infortainment system it ranked 3 for cognitive distraction [8].

Development of driver assistant systems can reduce the number of car accidents that happen because of driver distraction but sometimes these driver assistant systems don't work efficiently in the real driving environment. For example, using nonvisual and multi-sensing collision warning system reduces the number of rear-end collisions but In the real driving environment different kinds of noise such as passengers' conversations , music, and the sound of car's engine cover the warning noise and sometimes it can make an audio distraction for the driver [9].

The workload system in a car can reduce the distraction that is made by driver assistant system. For example, the lane changing warning is unnecessary when the driver is aware it only causes an audio distraction in this situation. The workload can reduce false alarms or unnecessary alarms by sensing driver inattention to the road and it uses warning about the situation when the driver is truly unaware [10].

Reducing the response time is an appropriate solution to reduce the driver distraction. In this paper, we discuss the details of an experiment that aim to reduce the response time of the driver while she is using infotainment system. In the experiment, the effect of customized user interface in reducing driver distraction was measured.

2 Customized User Interface

The use of application in the car center stack differs from person to person. Six features has been used in our design and each feature includes several commands. Although diversity kinds of commands are available the modern's car center stack, most of the users only use few commands so a customized center stack would be really helpful since the most frequent commands has been bolded in the customized interface so the user can find them easily and the response time of users that use customized user interface is less than users that use standard user interface.

Using setting to customize the user interface is not convenience for all drivers. Some drivers don't know how to change the setting of their car's system so it is better to use AI methods to customize the user interface since it is more convenience also the driver doesn't need to learn how to use setting of his car's infotainment setting.
Each driver has a specific behavior's pattern during using his car's infotainment system. This pattern can be found and used to customized the user interface.

2.1 Experiment

We conducted this experiment to show customized interactive interface has a better response time compared to standard user interface. The experiment was conducted in a Drive Safety research simulator DS-600s (Figure 1). The DS-600c is a fully integrated, high performance, high fidelity driving simulation system which includes multi-channel audio/ visual systems, a minimum 180° wraparound display, full-width automobile cab (Ford Focus) including windshield, driver and passenger seats, center console, dash and instrumentation, and real-time vehicle motion simulation. It renders visual imagery at 60 frames per second on a sophisticated out-the-window visual display with horizontal field-of-view. It also includes three independently configurable rear view mirrors.

An android application was developed to display the user interface. The application was hosted on the Android v4.4.2 based Samsung Galaxy Tab4 8.0 which was connected to Hyper Drive simulator.

Figure 1: Hyper Drive Simulator in the Experiment

45 people belonging to all sectors participated in this experiment. We asked them to drive on a previously programmed route and we gave them special instructions. Response time and the number of driver's errors have been measured throughout the experiment.

I. Response Time: This is the total time taken by a person to do the navigation operation. Specific action (like play FM) was prompted to the user and the response time was recorded.
II. Numbers of Error: This is the number of errors that has been done by each driver during the experiment. Errors can be as simple as not following traffic rules or accidents that has been made by each volunteer driver.

Combination of these two factors should help us quantify the distraction level.

2.2 Steps of Execution

Each participant was given few minutes of using the simulator to get familiar with the system. Once everyone completed their preparatory test drives, actual experiment was started. They were asked to drive two times. The first time, when they are driving, which is the primary task, they were asked to do the secondary task like operating the button on customized UI. Response time was recorded. They were

asked to drive the second time and interact with the buttons in the same way by using the standard UI. Response time was noted again. The average was calculated.

The sequence of execution of different steps:

I. User would take a test drive in which he would be briefly informed about actions he is expected to perform.
II. User starts driving and the user uses customized user interface in this step.
III. He was prompted to play FM when he reached four different locations. (Table1) he should navigate to specific FM page and play a song.
IV. He repeats the operation three more times.

During the execution of experiment the response time would be recorded and tabulated for each subject.

Table I

User prompt	Location
Play FM	Very first left turn when the cars would be coming from different directions.
Play FM	When the cars suddenly change on the freeway
Play FM	On the last left turn of the simulation when the cars bump into each other on the side
Play FM	When pedestrians are crossing the road

2.3 Analysis Methods

We used two methods to analysis this experiment. The first method is "comparing two alternatives". The statistics "comparing between two alternatives" is to compute the mean of differences of the paired measurements. The confidence interval C1 and C2 are determined with a confidence level, say 95%. Equation 1 shows the confidence interval. This is computed using the information of the mean of the differences and standard deviation. If the interval has zero, then the measurements are not statistically significant else we can statistically say that the mean lies within C1 and C2 with 95% confidence. This statistics helps us to estimate if the results of the experiment are statistically significant.

$$c_1 = \bar{d} - z_{1-\alpha/2}\frac{s_d}{\sqrt{n}}$$

$$c_2 = \bar{d} + z_{1-\alpha/2}\frac{s_d}{\sqrt{n}} \quad (1)$$

If c1<0 and c2 > 0, then the measurements are statistically insignificant. If c1 and c2 >0 or c1 and c2 < 0, then we can conclude the measurements are statistically significant and if c1 and c2 are positive then we can say alternative 2 is better than alternative 1 and vice versa.

The second method that has been used to analysis this experiment is analysis of variance (ANOVA). Analysis of variance is a technique to divide the total variation observed in an experiment into different meaningful components. This technique assumes that the errors in the measurements for different setting are independent with normal distribution.

Using ANOVA we can statistically compute:

I. The variation observed due to the effect of changing the alternatives.
II. The variation observed due the errors in the measurement.

Ideally the measurements are significant if the variation observed due to the change of alternatives is greater than the variation due to errors.

The variation due to alternatives SSA and variation due to errors SSE are calculated using equation 2 and equation 3. Where (K) being the number of alternatives. In our case k = 2 and n is the number of users in the experiment.

$$SST = \frac{\sum_{j=1}^{k}\sum_{i=1}^{n}(Yij \cdot Yij)}{kn} - \frac{(Yij \cdot Yij)}{kn} \quad (2)$$

$$SSA = \frac{\sum_{j=1}^{k}(\sum_{i=1}^{n}(Yij \cdot Yij))2}{n} - \frac{(Yij \cdot Yij)}{kn} \quad (3)$$

The estimate of variance of alternatives and errors are calculated using equation 4 and equation 5:

$$S^2_a = \frac{SSA}{k-1} \quad (4) \qquad S^2_e = \frac{SSE}{k(n-1)} \quad (5)$$

F value is computed using equation 6:

$$F = S^2_a / S^2_e \quad (6)$$

This calculated F value is compared with the value $F_{[1-\alpha;(k-1),k(n-1)]}$ obtained from the table of critical F values. Then we can say the variation due to alternatives is greater than variation due to errors with a confidence level of (1 -α).

We will use this statistical analysis to prove that the variation of response time is significantly due to the change in alternative (changing the UI setting of Center stack) rather than the variation due the different users. This also gives a statistical estimate of the effect on the response time due to change in alternative and change in user.

2.4 Results

The results of using "computing two alternatives" method (1) to analysis the measured response times in this experiment are as follows (7)(8):

$$C1 = 1366.64581 \quad (7)$$

C2 = 1700.420857 (8)

Since C1 and C2 are positive, and '0' doesn't lie in between C1 and C2, the measurements are statistically significant. With 95% confidence level we can statistically say that response time with customized user interface is better than response time with standard user interface. Figure 2 shows the response time with two different kinds of user Interface. You can see in this figure that the response time of driver with customized interface is less than the response time of this driver with standard interface.
The results of using "computing two alternatives" method to analysis the number of errors in this experiment are as follows (9),(10):

C1 = 0.168580822 (9)

C2 = 0.720308066 (10)

Figure 2: Response Time with Two Modes of UI in Milliseconds

Figure 3 shows the number of errors in standard user interface and customized interface. It is obvious that the number of driver's errors with customized interface was less than the number of driver's errors with standard user interface.

Since C1 and C2 are positive, and '0' doesn't lie in between C1 and C2, the measurements are statistically significant. With 95% confidence level we can statistically say that the number of errors with customized user interface is less than the number of errors with standard user interface.

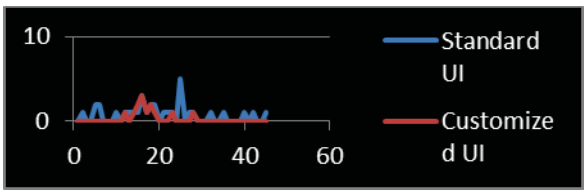


Figure 3: Number of Errors in Two Modes of UI

We applied ANOVA on response time measured (Table 4). Thus we can say 74.5% (SSA/SST) of total variation is due to difference among the two alternatives options and 25.5% (SSE/SST) of total variation is due to noise in the measurements of different user. Since F-computed is greater than F-tabulated, at 95% confidence we can say that the differences seen in the response time by changing the alternatives is statistically significant.

We similarly applied ANOVA on the errors measured and we found that 26.9% (SSA/SST) of total variation is due to difference among the two alternatives options and 73.1% (SSE/SST) of total variation is due to noise in the measurements of different user(Table 6). Since F-computed is greater than F-tabulated, at 95% confidence we can say that the differences seen in the response time by changing the alternatives is statistically significant (Table 5).

Table 4

SSA	381919524.6
SST	512608213.3
SSE	130688688.7
SSA/SST	0.745051512
SSE/SST	0.254948488
F-computed	257.1677663
F-tabulated	3.9846

Table 5

SSA	24.19432099
SSE	65.7997037
SST	89.99402469
SSA/SST	0.268843638
SSE/SST	0.731156362
F-computed	32.35729231
F-tabulated	3.9846

According the results of our experiments we can say that using customized user interface reduced the driver distraction significantly since it decrease the number of errors also it reduced the response time. Using customized interface causes less cognitive load because the most frequent commands has been bolded and driver can find them easier so it makes less cognitive distraction. In addition, it reduces the response time that means driver's eyes off-the-road time reduces by using customized user interface so the visual distraction reduces too.

3 Conclusions

We conducted the experiment with two scenarios, one with Standard user interface found in many cars and another with customized user interface. Response time was measured in both scenarios. In the first case average response time to operate the user interface turned out to be 3.8s to do a specific operation. When we made the participants to drive the car using customized user interface, average response

time was reduced to 1.7s which is less than half of the first scenario. The response time is directly proportional to the distraction caused since it is the amount of time the driver keeps working on the secondary task rather than focusing on the road. Reducing the response decreases the driver distraction. We also measured number of errors made during driving in both scenarios. We found that the ratio between numbers of errors made by the driver when operating with the standard user interface to customized user interface is 7:3. There was a reduction of errors by 57% in customized user interface. From the analysis of readings taken during experiment, we can say with 95% confidence level statistically that response time with customized user interface is better than standard user interface. Also errors with customized user interface are less than standard user interface. Hence Customized UI would be easier to interact and reduces driver distraction drastically so the driver distraction with customized user interface is less than standard user interface.

4 References

[1] Bastian Pfleging, Tanja Doring and Ignacio Alvarez. "Auto NUI : 2nd workshop on Automotive Natural User interface ", AutomotiveUI '12, pp.37-38, October, 2012.

[2] Christer Ahlstrom and Katja Kricher. "Review of real time visual driver distraction", 7th International Conference on Methods and Techniques in Behavioral Research, article No 2,August, 2010.

[3] Michael A.Regan, John D.Lee and Ignacio Alvarez. "Driver distraction theory, effects and mitigations",CRS press , 2009.

[4] Mustapha Mouloua, Amber Ahren and Edward Rinalducci."the effect of text messaging on driver distraction: A bio behavioral analysis ", proceedings of the human factors and ergonomics society 54th annual meeting ,Vol No. 19,pp.1541-1545,2010.

[5] Mustapha Mouloua, Daniela Jaramillo, and Janan Smither."The effect of iPod use on driver distraction", proceedings of the human factors and ergonomics society 55th annual meeting, Vol No.41,pp.5886-5888,2011.

[6] Nicola Dibben and Victoria J Williams."An exploratory survey of in-vehicle music listening", Sychology of music society for education ,music and sychologyresearch, Vol No. 35(4),pp. 571–589,2007.

[7] www.esurance.com/info/car/3-types-of-distracted-driving, accessed April 16th, 2015.

[8] www.speechtechmag.com , accessed April 10th, 2015.

[9] Fanxing Meng, Rob Gray, Cristy Ho, Mujthaba Ahtamad and Charles Spence. "Dynamic vibrotactile signals for forward collision avoidance warning systems", Vol.57,No.2,pp.329-346,March 2015.

[10] Paul Green. "Driver distraction telematics design and workload manages: safety issues and solutions", SAE International,No. 2004-21-0022, 2004.

Controlling Immune Memory Generated by Antibody Dynamics

Chung-Ming Ou[1]
[1]Department of Information Management, Kainan University, Taoyuan, Taiwan

Abstract— *Immune memory of antigens is formed as a limit behavior of antibody dynamics inspired by one-dimensional chaotic systems. Associativity of immune-memory mechanism can be investigated on the basis of dynamics determined by affinity index of antibody chains. The proposed model provides some control strategy based on the OGY method to steer immune-memory formation which is non-associative to associative memory formation which is shown by some stable periodic orbit.*

Keywords: chaotic dynamical system, associative memory, antibody dynamics

1. Introduction

Immune memory is one of the major feature of immune systems, particularly, for secondary and cross reactive immune responses. Short-living memory cells which form some immune networks is an accepted model for immune memory [1]. Immune memory mechanism can be explained through Jerne's immune network theory by regarding these as complex dynamical systems [2]. On the other hand, rather than a centralized architecture, immune memory belongs to a class of sparse and distributed associative memory [3]. Based on the above considerations, we are looking forward to building a mathematical model of immune memory mechanism which can describe such paradigm of short-lived memory cells with network dynamics.

Our idea is the following. First the traditional idiotypic immune networks (IINs) are too complicated to define network dynamics. Therefore we simplify such IINs to some network structure, namely, antibody chains. We then analyze the network dynamics of antibody chains. Morita has proposed non-monotonic dynamics for associative memory [4]. It introduced the autocorrelation matrix for such dynamics. We are inspired by such statistical dynamics for antibody chain, namely, discrete chaotic dynamical system defined by logistic function. In details, the memory format of the corresponding antigen can be defined as limiting behavior of such network dynamics. One advantage of antibody chains is its simplicity to define the network dynamics rather than population dynamics. Namely, the state transitions generated by such antibody dynamics is more suitable for population dynamics from immune memory viewpoints.

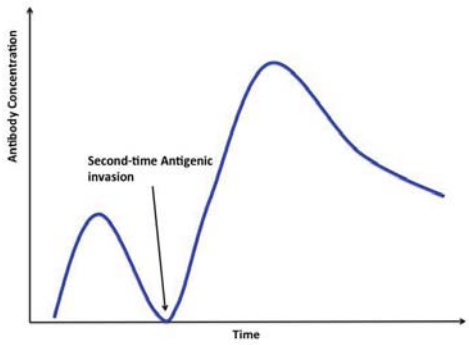
Another challenge for network dynamics of antibody chain is as follows. Is antibody dynamics inspired by logistic function a reasonable model of antigenic memory? One aspect of this question is the dynamics of cross-reactive immune response, namely, the mechanism of associative memory. In this research, such associativity can be characterized by the affinity index, which defines the degree of connection for the corresponding antibody chain. The latter can be the bifurcation parameter of the logistic function. One advantage of this model is the ability of controlling non-associative immune memory by way of perturbation structure of antibody chain successively. Moreover, such controlling technique of successive perturbations is proposed by Ott, Grebogi and York [5], or the OGY method.

The major goal of this research is to control the forming of non-associative immune memory based on antibody dynamics. There are two aspects for the latter: (1) associative memory based on chaotic immunodynamics inspired by [6][4]; (2) controlling the non-associative memory based on the OGY method [5]. One contribution of such approach is to establish the method of controlling immune memory inspired by chaotic dynamics. This research is a continued work for [7] and also inspired by Abbattista et al. [8].

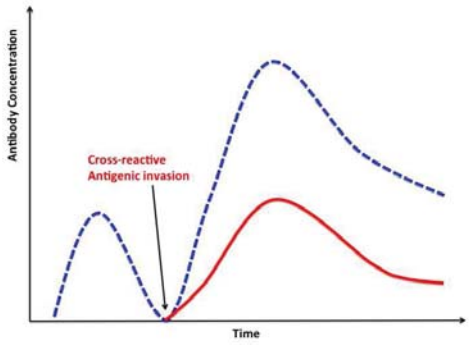
2. Research Background

2.1 Immune Memory

While a specific antigen invades human body, the immune system will respond by producing some antibodies which can eliminate this invaded antigen. The typical pattern of such immune responses can be illustrated by Figure 1(a) and Figure 1(b) ; it has three phases, namely, first immune response, second immune response and the cross-reactive response. For the first invaded antigen Ag, the immune systems will produce large amount of the antibody Ab_1 which binds to Ag. Therefore the amount of antigen clones for Ag will be reduced rapidly after they had reached the population peak. Simultaneously, the amount of Ab_1 is also declined to some constant value; we say this immune system is turned into the memory state. For the second phase, the same antigen Ag invades and the *remaining* antibody Ab_1 will take less time than that of first immune response to reach the population peak. Therefore this antigen cannot proliferate immediately as it did at the first phase. This is the reason why we seldom get sick at this stage. On the other hand, if an antigen Ag' which is very similar to Ag invades, then the same antibody Ab will also proliferate soon and eliminate Ag' clones. The second and the third stages are all related to immune memory mechanism to recall the stored



(a) First and Second Immune Responses



(b) Cross-reactive Immune Responses

Fig. 1: Three Stages of Immune Responses. Each peak represents the highest proliferation amount of antibody in three different stages

information of Ag. We will analyze the state transitions of the invaded antigen patterns between each phase based on network structure of immune systems, in particular the idiotypic immune networks and antibody chains. We are particularly interested in the third stage which shows an associative memory mechanism for the immune response to similar antigens. The first antibody Ab_1 binding to Ag in the immune response process is determined by the clonal selection mechanism of immune systems. Ab_1 secreted by some activated B-cell is the one with the highest affinity among those candidates of antibodies which are generated by hypermutation.

Immune system will react rapidly to the same or similar, even mutated antigens which had invaded the same human body before. This phenomenon implies that immune system can "memorize" associatively the formations of previously invaded antigens. One major evidence for such immune memory mechanism is that it is strongly affected by the concentration of soluble antibodies in the blood. Therefore some variables related to the immune memory might be correlated to the antibody concentration from the aspect of computational biology.

Immune memory mechanism is not fully understood so far. According to Smith et al. [3], at the end of an immune response, when the antigen clones are cleared, the B cell concentration decreases, leaving a persistent population of memory cells. The newer viewpoint of memory cells is that they are not lived longer than virgin cells; their life cycles depend on the persistence of antigens [9]. Besides memory cell mechanism, researches related to immune network theory imply the immune memory mechanism is formed by cyclic idiotypic immune networks (CIINs) than specific memory cells [10]. According to vaccinations, immune systems can react immediately to antigens which are similar to previously invaded ones. However, such theory is mainly dealt with the diversity of antibodies for antigenic recognitions by affinity matching. It is not sufficient to explain how memory recalls are activated for similar antigenic invasions. Therefore, for such associative memory mechanism of immune systems, it is worth of considering the immune network theory.

2.2 Antibody Chain

Idiotypic network theory implies that immune systems will emulate the presence of antigens even after they are eliminated [3]. The epitope of antibody molecule is called an "idiotope". An epitope of antigen Ag is recognized by the antibody molecule Ab_1 and by the receptor molecule on the lymphocyte of LU_1. The antibody Ab_1 and the receptor of LU_1 have the idiotope which is recognized by antibody Ab_2 and the receptor on the lymphocyte of LU_2. Continuously, we reach an antibody Ab_N, while the antibody Ab_1 and the receptor on the lymphocyte of LU_1 also recognize idiotopes on antibody Ab_N. Ab_N constitutes an *internal image* of the antigen Ag. Immune network is formed by interactions between lymphocyte interactions. These LU_i form an closed loop, namely, closed idiotypic immune network (CIIN).

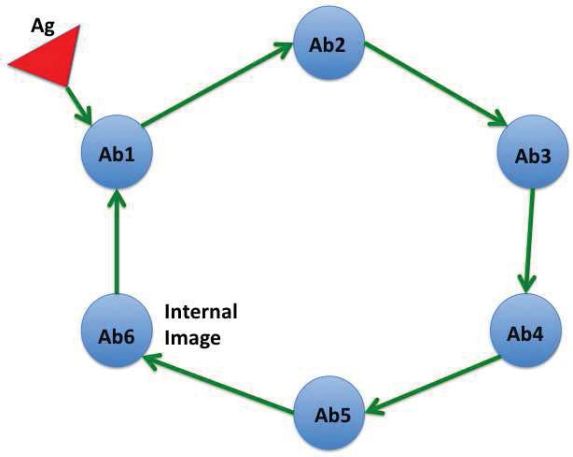


Fig. 2: Antibody Chain

The advantage of this simplified network, namely, *antibody chain*, is the following. The immune response, in

particular, immune memory, can be exploited by state transitions determined by such simplified networks. There are two stable states for idiotypic immune network; for the first state, the antibody Ab_1 is not produced. For the second stable state, Ab_1 is produced and antigen may or may not be completely eliminated [11]. The memory of such antigen is thus established. Network models of immune responses described by dynamical systems can be referred to [12][13][14].

Given λ a positive integer. The relationship "Ab_k recognizes Ab_i", denoted by $Ab_i \rightarrow_\lambda Ab_k$, is defined if Ab_i and Ab_k are λ-complementary match; namely, there are λ different digits between Ab_i and Ab_k. Frequently we simply write $Ab_i \rightarrow Ab_k$ by omitting λ if the latter is known and fixed. For simplicity, it is reasonable that we consider antibodies from both network dynamics perspective. Certain immune network models can be contributed to this antibody dynamics such as the one proposed in [10]. The antibody chain, written as $AC = \{Ab_1, Ab_2, \cdots Ab_N\}$ is defined as follows. (1) $Ab_i \in LU_i$, for all $i = 1, 2, \cdots N$; (2) The idiotype of Ab_i can be recognized by the partatope of Ab_{i+1}, namely, $Ab_i \rightarrow Ab_{i+1}$, for all $i = 1, 2, \cdots N-1$; (3) $Ab_N \rightarrow Ab_1$.

2.3 One-dimensional Chaotic System

One-dimensional chaotic systems have shown many applications in highly nonlinear systems which are sensitively dependent on initial conditions. One the best-known one-dimensional chaotic systems is the logistic function which is defined by

$$F(x, \alpha) = \alpha \cdot x(1-x) \tag{1}$$

where $\alpha \in [0, 4]$ is the bifurcation parameter of (1). Such parameter is the major character for dramatic change of system behavior. (1) also generates a one-dimensional discrete dynamical system defined as follows.

$$x_{n+1} = F(x_n, \alpha) = \alpha \cdot x_n(1-x_n) \tag{2}$$

Fig. 3 is the bifurcation diagram of (2). It shows that the values of fixed points for (2) for varied $\alpha \in [0, 4]$. While $1 < \alpha < 3$, the dynamics has two fixed points, one is 0, the other is $x = \frac{\alpha-1}{\alpha}$. As $\alpha = 3.5$, periodic orbit with periodicity 2 appears. As α increases, periodic orbits with higher periodicities appear. As $\alpha > 3.65$, (2) shows chaotic behavior (Fig. 3).

Definition 1: Consider the discrete dynamical system $x_{n+1} = F(x_n)$. The Lyapunov exponent of x_0 is defined by $\lambda(x_0) = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln |f'(x_i)|$, if the limit exists. If $\lambda(x_0) > 0$, then the dynamics of (2) with an initial condition x_0 is chaotic.

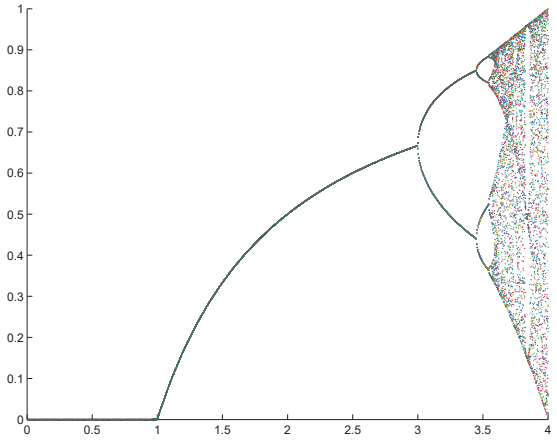


Fig. 3: Bifurcation Diagram of (2)

3. Results

Major results in this paper include the modeling of dynamics of immune memory mechanism from one-dimensional dynamical system (2). This immune memory mechanism can explain the associativity property by the chaotic property of (2).

3.1 Mathematical model of Immune Memory Mechanism

We establish a mathematical model of the immune memory mechanism. First is the definition of the immune memory function as follows.

Definition 2: The immune memory forming function $f : \mathbb{R} \rightarrow \mathbb{R}$ is a real-valued continuous function defined on $\mathbb{R}$ with the following condition. There exists some nonempty set $E \subset \mathbb{R}$ such that $\lim_{k\to\infty} f^k(x)$ exists for all $x \in E$.

According to immune memory forming function, we can define immune memory format for antigens. The format of each antigen is the binary vector of length n. Each antigen can be defined by some rational number $x \in [0, 1]$ with finitely many digits.

Definition 3: Fixed k a positive integer. Given an immune memory forming function f, the k-level immune memory of antigen Ag is defined by $f^k(Ag)$. The immune memory of Ag is defined by $\lim_{k\rightarrow} f^k(Ag)$, if the latter exists.

Remark 1: If F is a logistic function, then $f_\alpha(x) = F(x, \alpha)$ is a immune memory function, for $\alpha \in (0, 1]$.

3.2 Network Dynamics of Antibody Chains

We propose a network dynamics F of antibody chains inspired by the logistic function (1). Such dynamics of state transitions can generate immune memory of any given antigen. The idea of this model is as follows. Some character of the antibody chain AC can influence the forming of

immune memory. If AC has strong affinity for adjacent antibodies, it will incur strong associative memory for antigens "similar" to previously invaded antigen Ag. Therefore, we will describe such character at first. We first define the affinity between two molecules.

Definition 4: Let n be the length of molecules X and Y. The the affinity between X and Y, $\gamma(X,Y)$, is defined by $1-\frac{d(X,Y)}{n}$, where d is the Hamming distance.

Now we can define the affinity index of any given antibody chain. For an antigen Ag, we also assume that there exists an antibody chain $AC=\{Ab_i\}_{i=1}^N$ such that it is a CIIN. Such AC can be characterized by affinity index which describes the degree of aggregation for AC.

Definition 5: The affinity index L_{AC} induced by the antibody chain AC is defined by

$$L_{AC}=4\cdot(1-\frac{1}{N-1}(\sum_{i=1}^{N-1}\gamma(Ab_{i+1},Ab_i)) \quad (3)$$

Therefore L_{AC} is a real number in $[0,4]$. Given an antigen Ag^T (column vector), we transform Ag^T to a real number $x(0)=\sum_{i=1}^n Ag_i 2^{-i}$ between 0 and 1, where Ag_i is the i-th component of Ag. For example, if $Ag=$"11001", then $x(0)=2^{-1}+2^{-2}+2^{-5}=0.78125$.

Affinity index represents the degree of connection for the corresponding antibody chain. The smaller the index, the more associative the immune memory. Based on the affinity index, the network dynamics F derived by L_{AC} is the following.

$$x(n+1)=F(x(n),L_{AC})=L_{AC}\cdot x(n)(1-x(n)) \quad (4)$$

$n=0,1,2,\cdots$.

3.3 Memory formed by Antibody Chains

The network dynamics f generated by antibody chains activated by some antigens can generate memory formats of invaded antigens. It is similar to the dynamics of attractor networks originally proposed by [15]. Let $\mathcal{S}$ be the set of all binary molecule formats with finite lengths. The network dynamics F defined on $\mathcal{S}$ generates antigenic format of Ag by $\lim_{k\to\infty} f^k(Ag)$. If $f^k(Ag)\to s^1$, for some $s^1\in\mathcal{S}$, then f generates an attractor dynamics (as s^1 is exactly an attractor of F). s^1 is also called a memory format of antigen Ag (Figure 4). The domain of attraction of s^1, $\mathcal{A}(s^1)$, is defined as $\{s\in\mathcal{S}: F^k(s)\to s^1\}$. The associative memory of antigens can be defined by stable fixed points of such network dynamics.

For an immune network AC activated by antigen Ag with network dynamics F, if there exists a binary molecule format s^1 which is an attractor of f, then we say the immune network AC can memorize Ag. if some antigen Ag' is fallen into the domain of attraction of s^1, then Ag' can invoke proliferation of the same antibody (Ab_1) immediately.

Definition 6: Let $\gamma(\cdot,\cdot)$ represents the affinity between two binary molecule formats. An immune network $AC=\{Ab_i\}_{i=1}^N$ activated by an antigen Ag is equipped with associative memory, if there exists some $\delta>0$, such that whenever a new antigen Ag' with $\gamma(Ag,Ag')>\delta$ implies that $Ag'\to_\lambda Ab_1$.

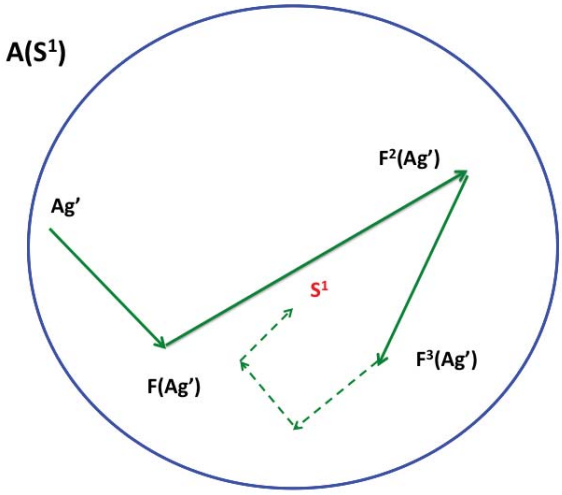


Fig. 4: s^1 is the memory format of the antigen Ag. One mutated antigen Ag' is fallen into the domain of attraction of s^1. This Ag' will activate the cross-reactive immune response.

According to this definition, if a new or mutated antigen Ag' which is very similar to the previous invaded antigen Ag invades, then the antibody Ab_1 which binds Ag can also bind Ag'. Therefore A'_g invokes cross-reactive immune response.

The network dynamics (3)-(4) may induce attractors such as stable fixed points or stable periodic orbits or even strange attractors. These attractors represent memory formats for different antigens. Memory formats are exactly some molecule attributes with dimension n, if attractors of (3)-(4) are stable equilibrium states. As for periodic orbit, memory format is composed by l binary molecule formats, where l is the periodicity of such orbit. For example, $l=2$ represents 2-periodic orbit

Given an antigen Ag with activated antibody chain $AC=\{Ab_i\}_{i=1}^N$ with dynamics (4), then the memory format of Ag is defined by the limit of Ag by F, namely, $\lim_{k\to\infty} F^k(Ag)$, where F is the network dynamics (4), see Figure 4.

Once the same or a similar" antigen to Ag, say Ag', invades the immune system again, the memory recall process of the immune systems will be activated by comparing the $F^k(Ag')$ with S^1, the memory format of Ag. According to system dynamics, such similar antigens are elements of domain of attraction of the dynamics f, namely $Ag\in\mathcal{A}(s^1)$. Suppose the same antigen Ag invades the immune systems again, then $f^k(Ag)$ will immediately converges to s^1. On the other hand, if some similar antigen Ag' invades, the

network dynamics $F^k(Ag')$ converges to s^1, if $Ag' \in \mathcal{A}(s^1)$. Therefore, the domain of attraction of antigen memory format is the major criterion whether similar or mutated antigens will activate the original antibody Ab_1 and the same antibody chain AC. In this way, the dynamics of antibody chains is the key for cross-reactive immune responses which show the associative memory mechanism for the immune systems.

Proposition 1: Let $L^- \subset [0,1]$ be the region of negative Lyaponov exponents of (2). Then the immune memory of an antigen x_0 exists, if $(\alpha, x_0) \in L_-$.

3.4 Controlling Associativity of Immune Memory Forming

we propose the method of controlling associativity of immune memory forming based on the OGY method. Given $(\lambda, x_0) \in L^+$. By given successive perturbations $\Delta\alpha_n$ such that the dynamics $x_{n+1} = F(x_n, \alpha + \Delta\alpha_n)$ is equal to either $\{p, q\}$. The latter is some stable periodic orbit. According to OGY method, $\Delta\alpha_n$ can be derived as follows.

$$\Delta\alpha_n = \alpha_0 \frac{[2p-1][x_n - p]}{p[1-p]} \tag{5}$$

3.5 Simulations

We simulate memory forming process via network dynamics (3)-(4). In this way, we can analyze memory formats of mutated antigens Ag' which are similar to the previously invaded and memorized antigens Ag with only a few different attributes (m is the number of mutated attributes). We give a complete list of parameters as (Table 1) with $iterno = 100$, $n = 30$ and $m = 5$ for every simulation. Memory formats can be represented as real number between 0 and 1.

Table 1: Parameters for Immune Networks

Para.	Description
L_{AC}	Affinity Index of Antibody Body AC
λ	Affinity Threshold
m	Number of mutated attributes from Ag
$iterno$	Numbers of Iterations
α	bifurcation parameter for logistic function
n	Length of Ag, Ab
N	Length of antibody chain AC

3.5.1 Simulation One: $3 < L_{AC} < 3.6$.

According to varied simulations, we observe that AC is difficult to form if affinity threshold $\lambda \geq 0.7$. Therefore we will simulate the memory format for $\lambda \geq 0.7$ by assigning L_{AC} values directly to (4) without generating antibody chains.

Fig 5 shows that memory format for Ag is a stable periodic orbit 0.56, 0.76 with periodicity equal to 2. $L_{AC} = 3.1026$. As for the memory format of mutated antigen Ag', Fig. 6 illustrates a better view that two memory formats are identical (after 30 iterations). The mutated antigen Ag' induces a cross-reactive immune response activated by original antibody chain, as its memory format is also convergent to that of Ag.

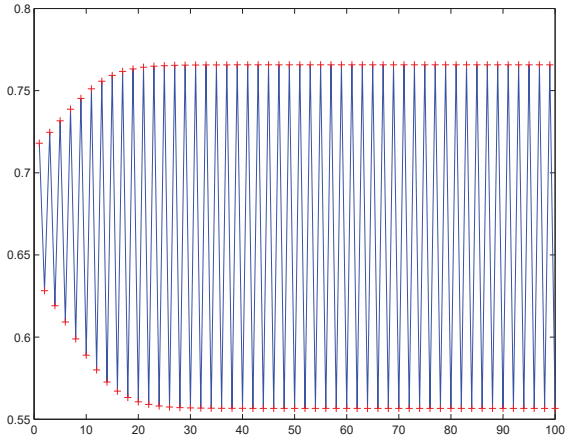


Fig. 5: Memory Formats of $L = 3.1026$

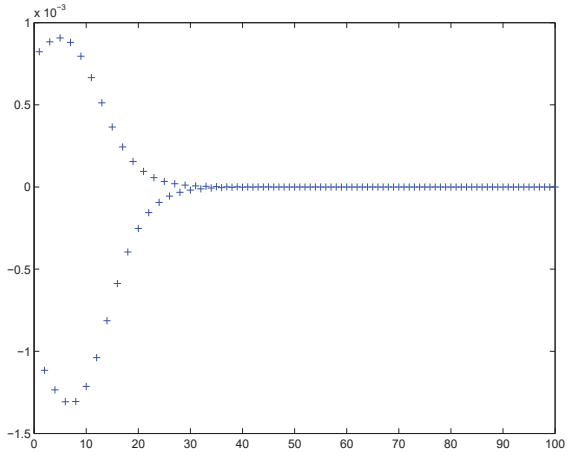


Fig. 6: Difference of Memory Formats between Ag and Ag'

3.5.2 Simulation Two: $3.6 \leq L_{AC} < 4$

Fig/ 7 shows that memory format for Ag is chaotic with $L_{AC} = 3.8$. As for the memory format of mutated antigen Ag', Fig. 8 illustrates a better view that two memory formats are completely different. In this case, the corresponding AC cannot activate a cross-reactive immune response to Ag'; Ab_1 cannot effectively eliminate Ag' clones.

For $L_{AC} = 3.8$, the unstable fixed point of (4) is equal to 0.7368. By calculating $p = f(q)$ and $q = f(p)$, where

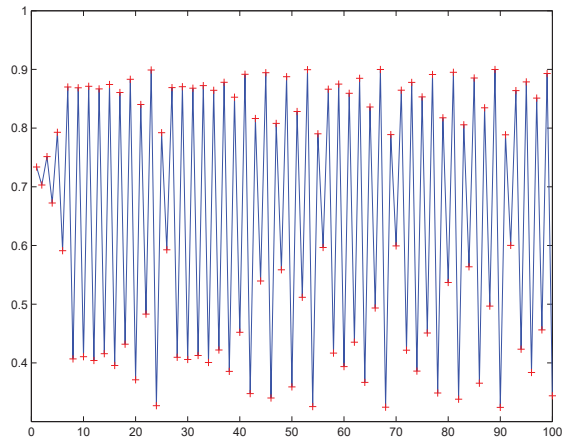


Fig. 7: Memory Format of Antigen (L_{AC}=3.6)

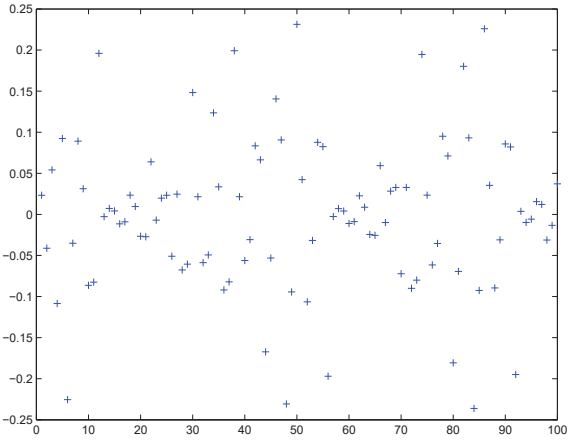


Fig. 8: Difference of Memory Formats between Ag and Ag' ($L = 3.6$).

$f(x) = F(x, 3.8)$, we obtain the stable periodic orbit $\{p, q\}$, where $p \approx 0.3737$ and $q \approx 0.8894$. The successive controls $\Delta\alpha_n$ of (4) is given by (5). Therefore, the antigen x_0 will generate a memory format $\{p, q\}$ which activates an associative immune memory. This memory format is also similar to Fig.5.

4. Conclusions

We propose an antibody dynamics model based on one-dimensional chaotic system. This model can describe the associativity of immune memory mechanism. The affinity index of antibody chain is the bifurcation parameter of logistic function F. Moreover, for higher affinity index, the antibody dynamics can generate associative memory for mutated antigens. Our model focuses on the memory forming process which is unique while comparing to other researches. For example, Anderson et al. studied intensively about the immune network model with antibody dynamics based on Cayley tree [16]. Our model also proposes a way of controlling non-associative memory forming based on the OGY method.

5. Acknowledgment

The author would like to thank support from the Ministry of Science and Technology of Taiwan under the grant number MOST-104-2221-E-424-006.

References

[1] A. de Castro, "A network model for clonal differentiation and immune memory," *Physica A*, vol. 355, pp. 408–426, 2005.
[2] N. Jerne, "Towards a network theory of the immune system," *Ann. Immunol. (Inst. Pasteur)*, vol. 125C, p. 373, 1974.
[3] D. Smith, S. Forrest, and A. Perelson, "Immunological memory is associative," *Artificial Immune Systems and Their Applications,The international Conference on the Multi-Agent Systems Workshop Notes, Kyoto, 1996*, pp. 62–70, 1999.
[4] M. Morita, "Associative memory with nonmonotone dynamics," *Neural Networks*, vol. 6, pp. 115–126, 1993.
[5] E. Ott, C. Grebogi, and J. York, "Controlling chaos," *Phys Rev Lett.*, vol. 64(11), pp. 1196–1199, 1990.
[6] S.-I. Amari and K. Maginu, "Statistical neurodynamics of associative memory," *Neural Networks*, vol. 1, pp. 63–73, 1988.
[7] C.-M. Ou, C. R. Ou, and Y.-T. Wang, "Immune memory with associativity: Perspectives on dynamical systems of networked lymphocyte cells," *Advanced Science Letter*, vol. 19(11), pp. 3179–3183, 2013.
[8] F. Abbattista, G. D. Gioia, G. D. Santo, and A. Fanelli, "An asspciative memory based on the immune networks," in *Proceedings of the IEEE Internatinal Conference on Neural Networks, 1996*, 1996, pp. 519–523.
[9] P. Matzinger, "Memories are made of this ?" *Nature*, vol. 369 (23 June), pp. 605–606, 1994.
[10] T. Sonoda, "Formation and stability of a memory state in the immune network," *J. Physical Society of Japan*, vol. 81(4), pp. 1408–1424, 1992.
[11] G. Parasi, "An introduction to the immune network," *Arvix*, vol. 87, pp. 1–7, 2008.
[12] A. Coutinho, "A walk with francisco varela from first- to second generation networks: In search of the structure, dynamics and metadynamics," *Biol. Res.*, vol. 36, pp. 17–26, 1993.
[13] P. Varela and A. Coutinho, "Second generation immune networks," *Immunol. Today*, vol. 12, pp. 159–166, 1991.
[14] G. Weisbuch, R. DeBoer, and A. Perelson, "Localized memories in idiotypic networks," *J. Theor. Biol.*, vol. 146, p. 483, 1990.
[15] T. Kohonen, *Self-Organization and Associative Memory*. Springer-Verlag, 1983.
[16] R. Anderson, A. Neumann, and A. Perelso, "A cayley tree immune network model with antibody dynamics," *Bulletin of Mathematical Biology*, vol. 55, pp. 1091–1131, 1993.

A Method for the Automatic Determination of Seats in Non-territorial Offices

Naoki KAWATA[1], Mitsunori MIKI[2], Katsuya ITO[1], Daichi TERAI[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 performed the proposal and examination of the method which determine a seat by random number and seat decision rules in Non-territorial offices. When the verification experiment of the subject point in a free-address office was conducted, the problem of fixation of a seat and a next seat member's fixation was checked. Moreover, it became clear that many work people have the low consciousness which tries to aim at communication with people. Then, when experimented using the seat determination system incorporating this technique, it was checked that those problems are solved and the freedom of a user's seat selection can be secured.*

Keywords: Office, Non-territorial Office, Free Address Office

1. Introduction

The currently most common office seating layout in Japan is what is called island office, whereby desks are arranged into one or more rows ("islands") consisting of an even number of desks facing each other (and sometimes with a single desk at the end of the row). This layout was originally devised on the basis of concepts of management, division of labor, and hierarchization. Since these concepts were important in the period of high economic growth, the layout spread widely and has been prevalent to this day.

It was high information processing capacity that was required of offices in the period of high economic growth. Workers were supposed to perform the duties assigned to each of them and to consult with each other if they had to within the team assigned to each row (island). The team leader was supposed to manage the progress of work and issues in the team. For work for which communication is closed within a team, island office was an effective office layout. Owing to the development of IT and OA equipment in recent years, however, administrative processing has been simplified, and the content of work required of offices has shifted to planning and creation of ideas, which require greater originality. This resulted in a change of a working style focusing on information processing to one focusing on communication that leads to an improvement in intellectual productivity. As a result of a change in working style, workplaces are also required to change. Above all, free address offices and non-territorial offices are drawing much attention.

A non-territorial office refers to a type of office designed by a method whereby individuals do not have their dedicated seat and equipment and fixture are shared by multiple people. A free address office refers to a type of office subsumed under non-territorial offices where any worker can take any seat to work. The study by Allen [2]. revealed that users were able to choose their seat in light of their preference and mood in a non-territorial office and that opportunities for communication with more people increased in a non-territorial office relative to an office with fixed seat assignment. Furthermore, an improvement in intellectual productivity is expected owing to seating customized to individual character and preferences and an increase in communication.

The concept of the free address office dates back to the 1970's. The purpose of a free address office was to save space in an office for cost reduction. Although it was adopted by many offices during the 1990's, it then lost popularity and remained unpopular for some time as it caused competition for seats, due to the insufficient number of seats, and the lack of collaboration in an office. In the middle of the 2000's and after, as changes in workers' style of working due to the development of information technology and the adoption of open seating, an effective use of office space, etc., came to be considered as important, more companies have introduced a non-territorial office whose concept is broader.

There are, however, several issues of concern with the use of the shared section of a free address or non-territorial office. In this paper, we made an automatic determination of seats (seat assignment) in accordance with random number and predetermined rules as a method for resolving those issues. Furthermore, we considered a method for facilitating communication more effectively and verified issues newly arising in automatically assigning seats.

2. Problem in Non-territorial Offices

In a non-territorial office, a meeting space suitable for engaging in collaborative work and a refreshing space suitable for taking a break can be taken as a working space to facilitate communication with various people at various working places. There are some issues concerned with free seating offices such as free address and non-territorial offices [2], which are shown below.

- Rigidification of seating
- Rigidification of people sitting together

The first issue, the rigidification of seating, refers to such an issue that a particular worker always takes the same seat and/or puts his/her belongings on it to indicate that it is his/her seat, preventing others from using it. This goes against the purpose of the effective use of office space. It has also been indicated that a user at a higher position or age tends to occupy the same seat, which leads to the diminished degree of freedom for their subordinates or younger users to choose a seat.

The second issue of the rigidification of people sitting together refers to such an issue that workers belonging to the same group tend to sit close to each other. Furthermore, there are two types of rigidification of people sitting together. One is that, with the rigidification of seating, those who prefer the same seat come to sit close to each other. Another is that, as workers belonging to the same group consider sitting close to a particular worker in choosing a seat, the same group always sit close to each other in any area. Either type of rigidification inhibits the advantage of having opportunities to interact with various users in other departments or those in other occupational categories, which is one of advantages of the non-territorial office.

3. Examination of automatic seat assignment

3.1 Examination of rigidification issues

To address two issues of the rigidification of seating and people sitting together, we attempt to solve the issue of the rigidification of seating by assigning a seat to each worker randomly by using a system. Furthermore, we examine a method for facilitating communication more effectively by variably managing the interaction among workers by adding various rules for seat assignment (seat assignment rules or SA rules for short) to the system mentioned in the above idea. An SA rule refers to a rule for narrowing candidate seats in automatic seat assignment so as to facilitate interaction effectively. The addition of an SA rule enables augmenting the opportunities for the interaction between particular persons and/or groups and prohibiting a particular way of using seats. For instance, if it is intended to solve the problem of the rigidification of seating and people sitting together by using SA rules, it suffices to add the following SA rules.

- A worker should not be assigned to the same seat that he/she occupied the last time.
- A worker should not be assigned close to a person who took a seat in the same area the last time.

3.2 Examination of the variable management of interaction opportunities

The result of the questionnaire taken at the preliminary experiment conducted in advance revealed that many workers put a greater priority on what kind of seat to take than on what kind of person they sit near. Therefore, it is considered possible to facilitate active interactions and the construction of new human relationships by supporting workers by the system with regard to whom to sit near and interact with. We describe how to realize this by SA rules. Workers are classified into groups in advance by their attributes such as affiliation and age. It becomes possible to manage interaction opportunities variably by assigning seats to workers in such a way that workers having the same or different attributes sit close to each other. Such an SA rule is called a worker interaction changing SA rule. Examples of the attributes used in the worker interaction changing SA rule.

- Age
- Gender
- Organizational unit
- Project

If the interaction among workers can be controlled, the construction of smooth human relationships can be supported in a non-territorial office also. For instance, if the attribute used is set to age, vertical interaction can be facilitated by assigning seats to workers in such a manner that managers and senior employees sit close to younger employees. It is also possible to enhance information sharing and a sense of competition among contemporaries by assigning seats in such a way that those in the same generation are seated nearby. Furthermore, in training employees through OJT（On-the-Job Training）, which is also useful in new employee education, assigning seats so as to make sure that a superior in charge of teaching and subordinates are seated nearby enables subordinates to learn how to do job by looking at how their boss works and to communicate closely with their boss, which, in turn, are expected to result in such effects as greater training efficiency.

4. Automatic seat assignment system

4.1 Outline of the automatic seat assignment system

Fig.1 shows how the prototype automatic seat assignment system looks like. This system is composed of a card reader, a screen for entering a preferred seat, and a seat display screen. You hover an IC card over the card reader and select a preferred seat on the screen for entering a preferred seat. A seat randomly chosen from seat candidates matching SA rules is assigned to you and output to the seat display screen.

Seat assignment under this system is made by narrowing seat candidates down by one or more SA rules and choosing

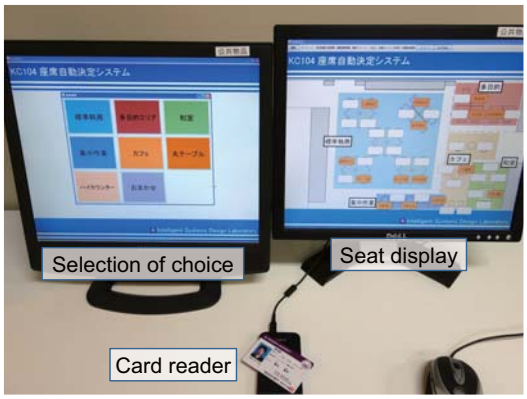


Fig. 1: User Interface of the system

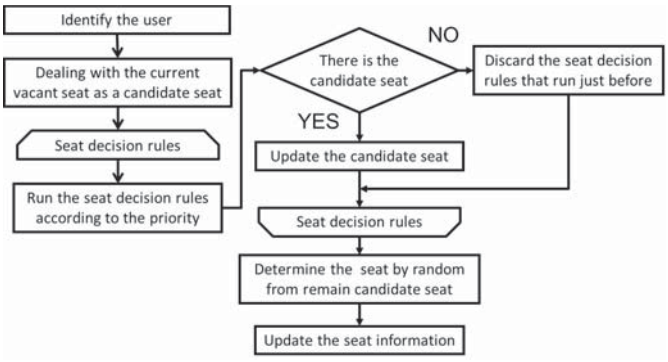


Fig. 2: Algorithm of automatic determination of seats system

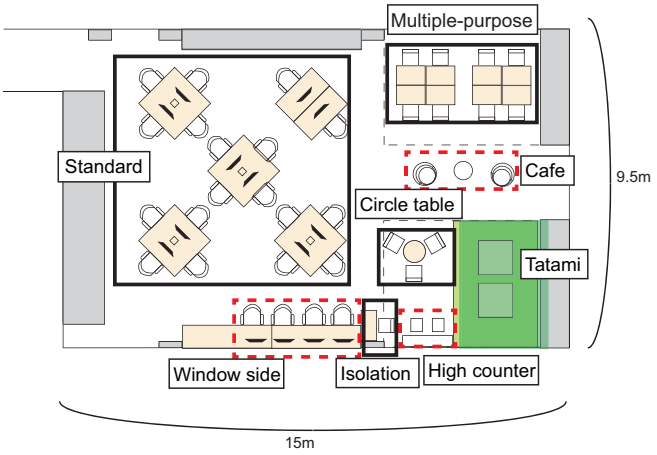


Fig. 3: Experiment environment

one from among vacant qualified seats by random number. When multiple SA rules are used in combination, a priority is assigned to each rule. SA rules of this system are shown below in the descending order of priority.

1) It is prohibited to assign an individual to the same seat as he/she took the previous time.
2) It is prohibited to assign an individual to any seat near any person who is seated in the same area in which the seat which the said individual took is located.
3) Seat preference
4) Priority among interaction opportunities
5) Density of assigned seats

Section3.1 described Rule 1 on the prohibition of an assignment to the same seat as previously taken and Rule 2 on the prohibition of an assignment to a seat near a person previously taking a seat in the same area. Rule 3 on seat preference is a rule for selecting a working environment in a broader environment into which the system is introduced. Rule 4 on priority among interaction opportunities is an SA rule, described in Section3.1, for changing particular opportunities for interaction. Rule 5 on the density of assigned seats is a rule for determining the density of persons assigned to a single table. The greater such a density is, the more an increase in opportunities for interaction is expected. Workers' satisfaction and working efficiency can be reduced due to a decrease in the working area under such a circumstance, however. On the other hand, the opposite effect is expected to occur if the density is less. As this experiment focused on the investigation of an impact on interaction opportunities, seats were assigned so as to make tables as densely seated as possible in this experiment.

4.2 Algorithm for the automatic seat assignment system

As a solution to the two issue of the rigidification of seating and people sitting together, Free-address Office Support System and Program [3] by Ozaki et al. has been available as patented technology. By inputting the individual ID of a worker and having the system randomly assign a seat to each worker, communication opportunities in a non-territorial office can be increased while solving the rigidification issues. Fig.2 shows the algorithm for the system.

By using this algorithm, candidate seats for a worker are figured out in the order of priority per SA rules. If there is no candidate seat, the last SA rule applied is discarded to inhibit the occurrence of problems such as being unable to assign a seat to a worker.

5. Experiment verifying the effects of the worker interaction changing SA rule

5.1 Outline of the verification experiment

An experiment verifying the effects of the worker interaction changing SA rule was conducted. The experiment was conducted in a university laboratory in the form of a non-territorial office. Fig.3 shows the experimental environment.

As subjects were students at a university laboratory, the attribute was set to grade (in university or graduate school).

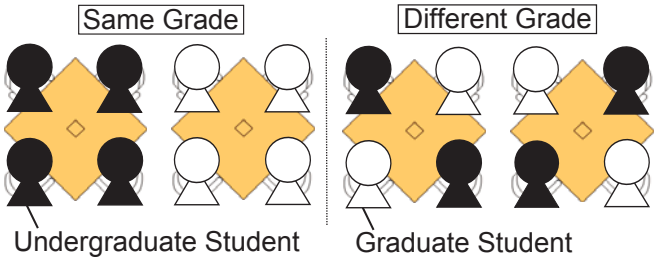


Fig. 4: Seating configuration of the grade communication

The following two SA rules were proposed as those increasing opportunities for interaction between them.

- Horizontal interaction facilitating SA rule
- Vertical interaction facilitating SA rule

The horizontal interaction facilitating SA rule facilitates horizontal interaction by assigning undergraduates and graduates to different tables. On the other hand, the vertical interaction facilitating SA rule facilitates vertical interaction by assigning the same number of undergraduates and graduates to the same table. Fig.4 shows the conceptual diagram for seat assignment by horizontal and vertical interaction facilitating SA rules

Users of this laboratory were composed of 20 undergraduates and 23 graduates in the academic year 2014 and 17 undergraduates and 23 graduates in the academic year 2015. Table.1 and Table.2 show experiment periods and SA rules which is about grade communication. A questionnaire was conducted each term to survey changes in the degree of intimacy and interaction opportunities.

Table 1: Experiment schedule(2014)

Experimental period	Grade communication rule
2014/4/4 ～ 5/6	Same
2014/5/7 ～ 6/1	Different
2014/6/2 ～ 6/30	Same
2014/7/1 ～ 8/3	Different

Table 2: Experiment schedule(2015)

Experimental period	Grade communication rule
2015/4/4 ～ 5/3	Different
2015/5/4 ～ 5/31	Same
2015/6/1 ～ 7/5	Different
2015/7/6 ～ 8/3	Different

5.2 Result of the verification experiment and discussion

Fig.5, 6, and 7 show the result that respondents to the questionnaire regarding changes in interaction opportunities by the effect of seat assignment answered that interaction

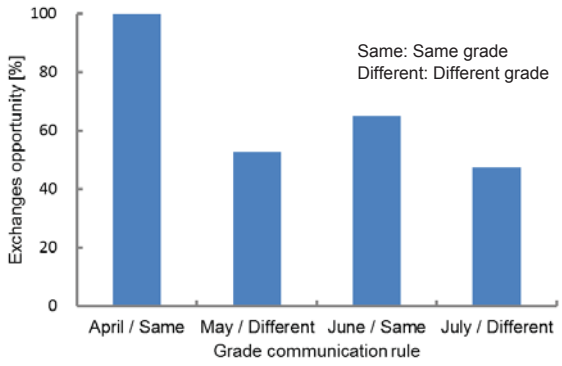


Fig. 5: The history of exchanges between undergraduate students: more than 50% of members increased(2014)

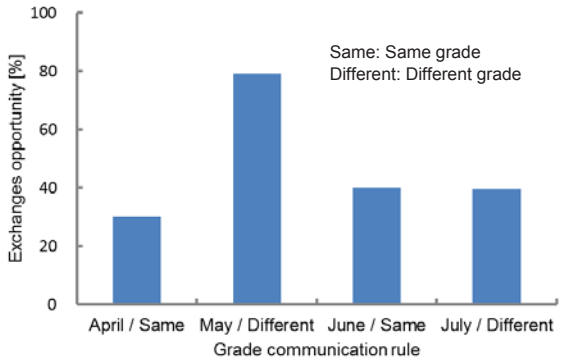


Fig. 6: The history of exchanges of undergraduate and graduate students: more than 50% of members increased(2014)

opportunities with the majority of other persons increased. Fig.5, 6, and 7 respectively shows a change in interaction opportunities among undergraduates in 2014, between undergraduates and graduates in 2014, and among undergraduates in 2015.

Let us discuss the horizontal interaction facilitating SA rule. If you pay an attention to April and May in 2014 in Fig.5, it is found that interaction opportunities were controlled as the value for May, in which seats were assigned to facilitate vertical interaction, was lower than that for April, in which seats were assigned to facilitate horizontal interaction. If you pay attention to April and June, in both of which seats were assigned to facilitate horizontal interaction, the effect by the SA rule decreased in June as interaction had been facilitated for two months since April. If, however, the result for June is compared with that for July, in which seats were assigned to facilitate vertical interaction, it is found that interaction opportunities were controlled.

Next, vertical interaction facilitating SA rule is discussed. In 2014, as shown in Fig.6, the value for May, in which seats were assigned to facilitate vertical interaction, was

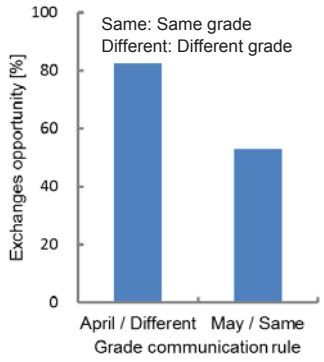


Fig. 7: The history of exchanges between undergraduate students: more than 50% of members increased(2015)

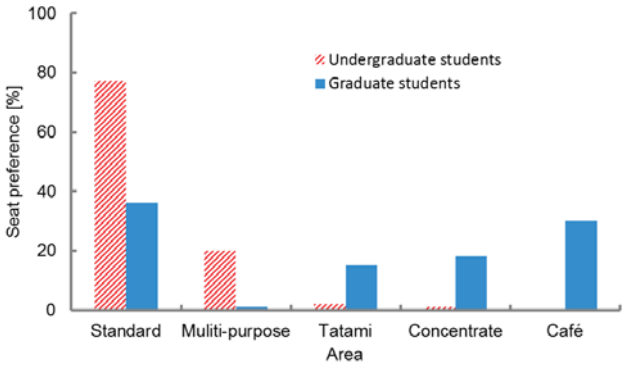


Fig. 8: The history of the seat chosen in April 2015

higher than that for April, in which seats were assigned to facilitate horizontal interaction. It is thus found that interaction opportunities were controlled. If, however, you pay attention to the value for July, in which seats were assigned to facilitate vertical interaction, it is lower than that for May, and there is no change from the value for June. As interaction had been facilitated for two months since May, the effect of seat assignment decreased in July. This is considered to be because there were then less users who can share the same table with those subject to vertical interaction facilitation under the vertical interaction facilitation SA rule.

In 2015, in which the SA rule used in April was different from that in 2014, interaction opportunities among undergraduates increased significantly in April, as shown in Fig.7, despite vertical interaction facilitating seat assignment. Fig.8, which shows the record of seats taken by undergraduates and graduates in April 2015, enables us to confirm that seats taken by undergraduates and graduates were extremely apart. This is considered to be caused by the fact that, in a new environment of this laboratory, undergraduates were assigned near other undergraduates who are in the same year as them and with whom they are acquainted rather than graduates who are in a different year and with whom they are not acquainted. In May, in which horizontal interaction was facilitated, the effect by seat assignment decreased as horizontal interaction increased in April. Therefore, for undergraduates newly assigned to a laboratory, it is preferable to make horizontal interaction facilitating seat assignment first.

6. Conclusion

In recent years, non-territorial office and a free address office are attracting attention again. These office has a issue of rigidification of seating and rigidfication of people sitting together. Also, we found that office worker don't choose the seat to take the communication by the examination. So, we proposed a method for the automatic determination of seats in non-territorial offices by using random number and SA rules to solve the problem and to make the communication environment.

We build a system and examine it by using the worker interaction changing SA rule in non-territorial office type laboratory. In result, we controlled the changing particular opportunities for worker interaction. But, if we don't choose suitable SA rule in suitable period, SA rule may give a little effects.

The proposed method can support the construction of human relations in non-territorial office. It is also possible to increase the worker interaction without the movement of furniture. In addition, it is also useful in new employee education and OJT. This system can expand the insight by looking at work of the various people and can take the communication with training employees, expected to effect rising the training efficiency.

References

[1] Thomas J. Allen and Peter G. Gerstberger.A field experiment to improve communications in a product engineering department: the non-territorial office.*the Human Factors and Ergonomics Society - Human Factors*,Vol. 15, No. 5, pp. 487–498, 1973.

[2] Xiaoyu Qu, Xinnan Zhang, Takashi Izato, Junzo Munemoto, and Daisuke Matsushita.Behavior concerning choosing workstations in non-territorial offices.*Journal of Asian Architecture and Building Engineering*, Vol. 9,No. 1, pp. 95–102, 2010.

[3] Tsukasa Ozaki, Japan patent2010-079585（2008.09.25）.

News Selection Method Considering the User's Interests for an Intelligent Conversation System

Eriko Yoshimura[1], **Misako Imono**[2], **Seiji Tsuchiya**[1] **and Hirokazu Watabe**[1]
[1] Dept. of Intelligent Information Engineering & Sciences, Faculty of Science and Engineering
Doshisha University, Kyo-Tanabe, Kyoto, Japan
[2] Dept. of Information Systems Computer Science Course, School of Informatics,
Daido University, Minami-ku, Nagoya, Japan

Abstract - *This paper proposed the news selection method for computer conversation. The method is user's matched news selection based user's personal information and the method is high topicality news selection considering generally importance. In this proposed method, we used association mechanism is not based only the notation information of the word. The proposed system conceptualized several information and news by using this mechanism and it was able to calculate association quantitatively between information and news.*

Keywords: News Selection Method, Association mechanism, degree of association, concept base, Conceptualized method, User's Interests

1 Introduction

Studies on the self-directed discourse of intelligent robots have been conducted from a variety of approaches. One area of study is the processing of replies using specialized methods developed under the rules of task-type conversations. Examples of purpose designed conversation systems include the DARPA Airline Travel Information System (ATIS) [1], a hotel reservation system [2], and a secretary system [3]. These systems guide the conversation toward fulfilling a target task and are extremely effective when used for carrying out clerical work. As a result, conversational systems have been developed for numerous specific purposes and some are already in practical use. Such systems can already grasp the state of a conversation as a transition diagram and develop effective conversational rules for accurately obtaining the information necessary for the task—without allowing the conversation to wander to other topics. However, such systems are unable to handle any issues other than the target task or to understand words that have not been registered. Thus, they are exceptionally poor when it comes to the development of systems for chatting or small talk.

Examples of chatting systems include Eliza [4], ALICE, Ractor, and Verbot. These systems have been improved to the point where they "learn" responses from their conversational partners. By continual placement in an environment in which they chat with humans, these systems can develop a growing repository of response patterns. A large volume of research on interactive conversation that is not task-driven, such as chatting, is being carried out (see, for example, [5] [6] and the interactive voice system described in [7]). These and other similar systems acquire conversational rules through inductive learning using genetic algorithms based on an enormous repository of interactive conversation samples. As a result, they demonstrate better performance than Eliza-type systems. These systems generate responses by imitating conversations contained in the abovementioned repositories and are very good at chat-type conversations. However, when a system simply learns previously used patterns, the processing of the intelligent conversation consists of robots listening to a user's talk and giving useful information to the user. In this case, it is necessary that the system can access new information as a source that is not arranged and accumulated beforehand, as in the task-type conversation, and is not accumulated by pattern learning of past knowledge, as in non-task type conversation.

This paper uses news articles as a source of information, but conversations that simply choose a random news article decrease the user's interest. In this paper, the authors propose a news selection method that considers the individual interests of a user for the automatic generation of conversation sentences by a robot.

2 Summarizing information

When using news articles as the information source in a conversation, the news contents must be summarized. Studies of article summaries have been carried out in recent years, for example, in [8] and in one subtask of an NTCIR evaluation type workshop[8]. As reported in the studies on article summaries, the most important information is the news title and the first few lines of the article. This paper uses news titles that express the contents plainly for conversation sentences. We propose a method that chooses news of interest for a user based on his or her personal information and interests and by selecting news about popular topics. To measure the relation between a user's interest and popular news, we use an association method that is not based only on

the notation information of the word. The method proposed in this paper uses mechanisms of human association and the between words and replies based on knowledge and common sense.

3 Association mechanism

For machines to exhibit human-like conceptual association abilities, we must first develop a quantitative characterization of the semantic similarities between words. In addition to familiar relations that exist between words—such as exact synonyms, near synonyms, and inclusion relations—we also have co-occurrence relations, partial relations, and other relations that are intuitively clear but not systematically formalized. Thus, for robots to achieve concept associations close to the intuitive, common sense capacity of humans, it is necessary first to construct a quantitative representation of the semantic similarities between words. Spurred by this challenge, Okamoto [9] and Watabe [10] studied the relations between concepts and proposed a concept base.

A concept base is a knowledge base comprising words (concepts) that have been mechanically auto-constructed from multiple dictionaries, newspapers, and other sources, along with a glossary of words expressing their meanings (attributes) and concepts. There is approximately 90,000 concepts. Sets of concepts and attributes are weighted to express the importance of the set. To understand concepts and weighting, let us define a given concept A as a paired set consisting of an attribute a_i that expresses the meaning of the concept and a weight w_i that expresses the importance of attribute a_i in expressing concept A. Attribute a_i is called a primary attribute of A, while an attribute for which a_i is the concept is called a secondary attribute of A, as shown in the following:

$$A = \{(a_1, w_1), (a_2, w_2), \cdots, (a_N, w_N)\}$$

Calculation of the degree of association [11], which is the strength of the association between two concepts, is evaluated quantitatively with the concept base. The two concepts are expanded as far as their secondary attributes, and the calculation is carried out using the weight to determine the optimum combination of the primary attribute and to evaluate the weight of the matching attributes. The degree of association is a real number between 0 and 1, where higher values indicate a stronger association.

Table 1 shows an example in which the calculation of the degree of association was carried out on concept A and concept B. As shown, cherry blossom has a much higher degree of association with flower than car. Sunny has a much higher degree of association with weather than school.

Table 1: Example of calculation of the degree of association

Concept A	Concept B	Value for degree of association
Flower	Cherry blossom	0.224
Flower	Car	0.001
Weather	Sunny	0.446
Weather	School	0.002

4 The news selecting method

We proposed a method for choosing a user's "interested" news (news of interest) using the user's information, user's interests, and popular topics in the news. For news sources, we used asahi.com [12], mainichi jp [13], and Yomiuri Online [14].

The proposed method uses the degree of association to calculate the relation between words. To calculate the degree of association, the target must be a concept having attributes and weights. Then, the method conceptualizes the user's information and news. The method can use an association word that is not written directly by the calculated degree of association.

4.1 Conceptualized news

When the news is conceptualized, the news headline becomes the concept and independent words in the text become the attributes. Each attribute is given a weight: a TF-IDF value of the news texts over a period of one month. Table 2 shows an example of conceptualized news.

Table 2: Example of conceptualized news

News headline and an extract of the text

Headline	Japan soccer representative, defeated by Belarus
Text	Japan soccer representative defeated by Belarus on 15th….

↓ conceptualized

Concept	Japan soccer representative, defeated by Belarus
Attributes	Japan soccer representative (0.22), Belarus (0.18)…

4.2 Personal information and conceptualized interests information

In this paper, a user's personal information is divided into objective profile information and subjective interested information. Profile information includes items such as a user's name, residence, and occupation. Interested information includes items about a user's subjective likes and dislikes. The user might input several items and several words for each item.

In the application of the method, we would prefer to get their information in a conversation, but this paper does not consider the automatic acquisition of personal information from conversations.

After the personal information is obtained, the interested information is conceptualized. The interested information is divided into likes and dislikes. Division of like and dislikes is conceptualized. The attributes are the inputted interested words and the weight is the inverse of the total number of inputted interested words. Table 3 shows personal information items and Table 4 shows an example of conceptualized interested information.

Table 3: Personal information items

Profile information	Interested information (likes and dislikes)
Name, Birthplace, Living place, Occupation, School(Employment), Nationality, Pet, Age, Gender	Food, Drink, Color, Sports, Country, Animal, Artist, Character, Book, Writer, Movie, Drama, Place

Table 4: Example of conceptualized interested information

Like	Food	Steak, Egg
	Sports	Soccer
Dislike	Food	Ginger
	Sports	Swimming

↓ conceptualized

Concept	Like
Attribute	Steak (0.2), Egg (0.2), Soccer (0.2)…

Concept	Dislike
Attribute	Ginger (0.2), Swimming (0.2)…

4.3 News score

4.3.1 News score considering personal information

The system under the proposed method gives a score of the relation between the news and the profile information as well as a score of the relation between the news and the interested information. The system calculates the news score by setting the personal information as the sum of the two scores of the abovementioned relations. The score of the relation between the news and the profile information is the ratio of the same profile word in the news, because profile words are special words. The score for the relation between news and interested information is the value resulting from subtracting the degree of association between the news and the dislike concept from the degree of association between the news and the like concept. By this method, the interested news is more likely to appear, and the disliked news is less likely to appear.

4.3.2 News score considering general importance

The news score considering general importance includes the news frequency and the score considering the interested words according to gender and age generation.

First, we explain the news frequency. Frequently reported news has higher importance. The system judges frequency by checking the number of times that nouns in a news title are described in all news titles of one day, and then makes a list of the news titles of one day. To obtain this list, the system gets all the nouns in the news titles and counts the number of times that each noun appears in the list. The total number of nouns is denoted as h in news title H, and the number of times that each noun appears in the list is n_i. We define the degree of frequency F of news H as follows:

$$F(H) = \frac{\sum_{i=1}^{h} n_i}{9h}$$

Next, we explain the score considering the interested words for each gender and generation. We use the Shunkan Ranking [15] in our method. Shunkan Ranking ranks 20 words from popular topics using a search engine that gathers search words for each gender and each generation. The generations range from ages in the 10s to ages in the 50s. The system obtains the concept from the interested category and the keyword concept according to each user's gender and generation from the last week of the Shunkan Ranking. The score for interested words, with consideration of the user's gender and generation, is given by the degree of association between these concepts and the news. To express the meaning of the Shunkan Ranking keywords, based on the user's gender and generation, the system gathers words in a web network. To gather the meanings of these words, the system uses the Auto Feedback method [16]. If a word has many similar meanings, the attribute word contains the user's interest. Therefore, the system obtains 20 words listed in decreasing order of the number of appearances. A category concept has the attributes of the 20 words and the weight value, which is the number of appearances normalized to 1. A keyword that appears in many days is important. Therefore, a keyword concept has attributes appearing over multiple days in the last week of the Shunkan Ranking. The weight value of the keyword concept is the overlap number normalized to 1. Figure 1 shows an example of the category concept and the keyword concept for Japanese males in their 20s.

The score considering the interested words for a user's gender and generation is the value of the degree of association between the news and the keyword concept as well as the

degree of association between the news and the category concept.

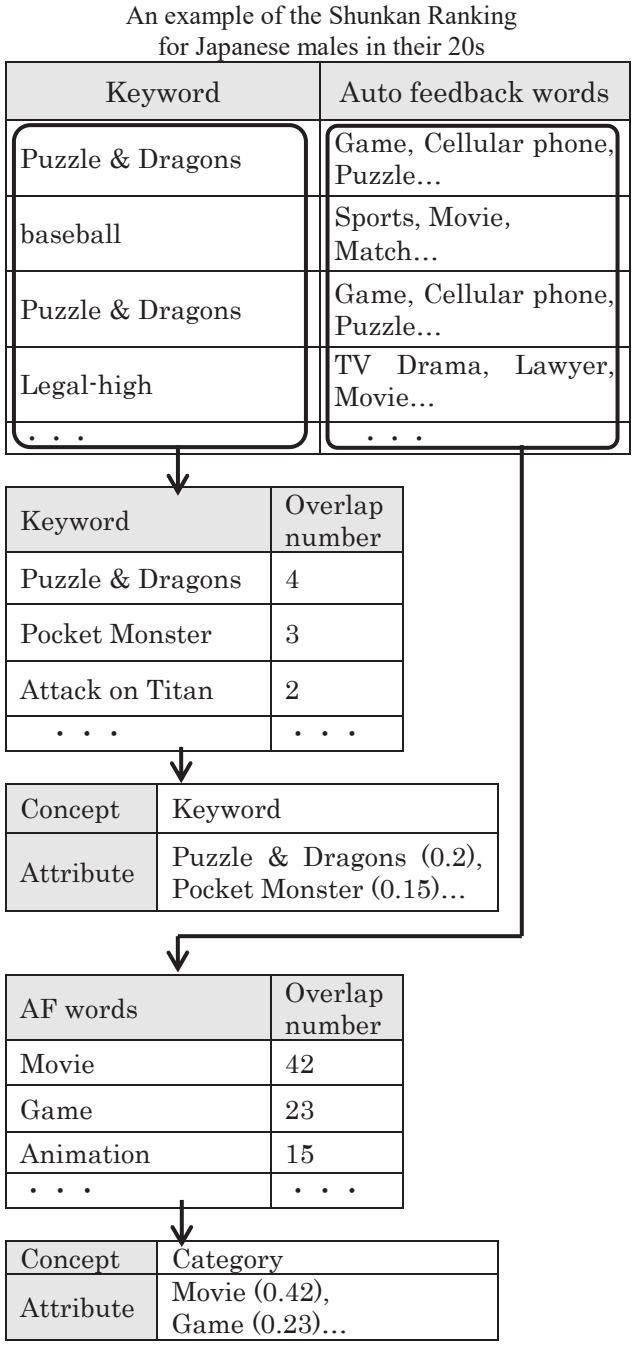


Figure 1: Example of category concept and keyword concept

The news score defined with the general importance uses the news frequency and the score with consideration for the interested words for a user's gender and generation. If the news score considering the interested word is high the general importance is high. Therefore, the news score considering general importance is calculated by multiplying the frequency degree and the score considering interested words for a user's gender and generation.

4.4 Selection of news

The news score is the total of the news score considering personal information and the news score considering general importance. The system lists the news scores in decreasing order.

5 Conversion to a conversation form

The system uses news titles for selecting conversation sentences. A news title has many unique descriptions and is characterized by its lack of a post-positional particle and a declinable word. For conversation, it is necessary to insert the missing particle and declinable word. Therefore, the system converts news titles into a natural conversation form with the "*Computer-Generated Conversation Based on Newspaper Headline Interpretation*" [17].

6 Evaluation and discussion

6.1 Evaluation

As preparation for an evaluation experiment, five participants provided their personal information beforehand. The participants read news titles and articles for the tested days and they expressed their interest or disinterest for each news article. For the evaluation, the output of the proposed system was compared with the news that the participants considered interested news. The evaluation is the ranking of the correct news rank in the output of the proposed system. The "correct" news was the participants' actual interested news.

The experiment was conducted for 3 days. The news articles were gathered over 3 days and the Shunkan Ranking was derived for the week preceding the day of the experiment. There was an average of 229 articles per day during the experiment and the total number was thus 687. This paper shows the results of the participants, who were five Japanese men in their 20s. For comparison, this paper shows the results for the proposed method using only the scores that consider the participant's personal information and the proposed method using only the scores considering general importance and random output. The methods for calculating the position degree of the correct news rank in the output of the proposed system are the average precision (*AP*), which was calculated for each day of the experiment, and the mean average precision (*MAP*). The *AP* is calculated as follows:

$$AP = \frac{1}{S}\sum_{i=1}^{n}\frac{Z_i}{i}(1+\sum_{k=1}^{i-1}Z_k)$$

where Z_i is a variable that is 1 when the news of rank i of the proposed system output is correct, and 0 when the news is incorrect. S is the number of correct news articles, and n is

the number of news articles of the day of the experiment. *MAP* is the average of multiple *AP* s. *MAP* is calculated as follows:

$$MAP = \frac{1}{T}\sum_{h=1}^{T} AP_h$$

where the output number of the participants is T. The average precision for every output is AP_h (h = 1,…, T). The *MAP* is the average of AP_h. Figures 2 through 4 show the results of the *AP* for the five participants (A–E) for each of the three days. Figure 5 shows the results of the *MAP* as the average of all results.

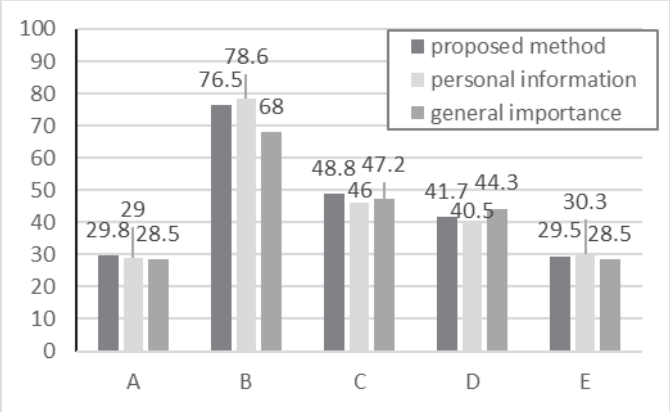


Figure 2: *AP* result in DAY 1

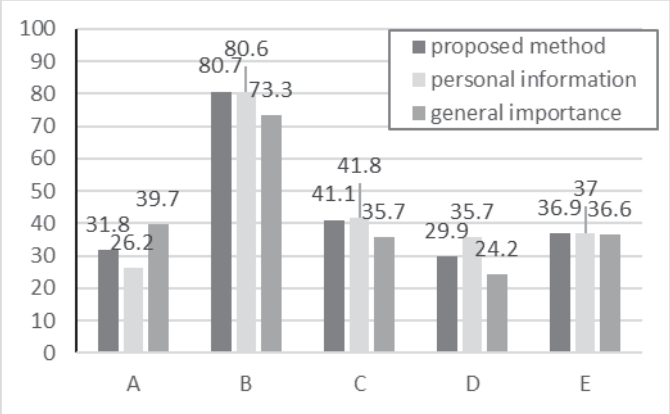


Figure 3: *AP* result in DAY 2

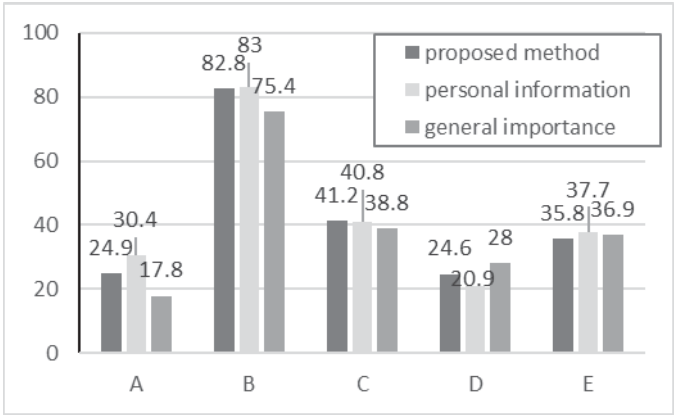


Figure 4: *AP* result in DAY 3

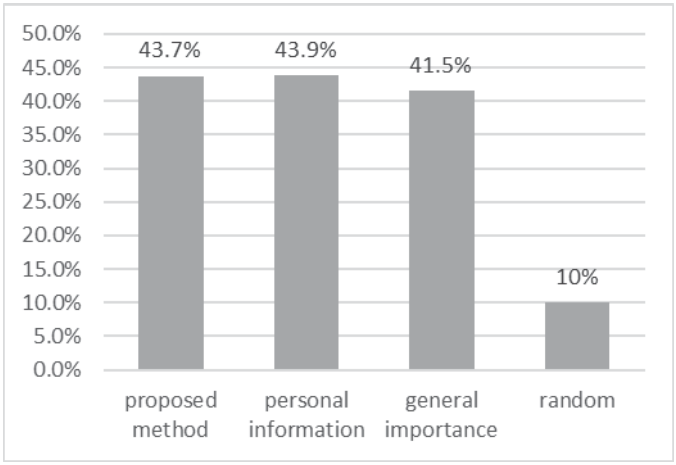


Figure 5: *MAP* result

6.2 Discussion

The *MAP* values for random news output is 10% (the *AP* result is similar). The results of the proposed method for the participants matched the news better than did the random news. The *MAP* of the proposed method using only the score considering the participant's personal information is 43.9% and the MAP of the method using only the score considering general news importance is 41.5%. The proposed method, using personal information and general news importance, shows good effect. The *MAP* of personal information and general news importance and the *MAP* of the proposed method (using the personal information and the general news importance) did not show a significant difference. We suggest the reason for the insignificance of the difference is the synergistic effect of the participant's personal interests and the general news importance being almost the same. In general, the main benefits of the proposed method are its use of the generation and the gender of the participants.

7 Conclusions

This paper proposed a news selection method for composing computer conversations. The method matches news topics with a user based on that user's personal information, personal interests and news of general importance. In the proposed method, we use an association mechanism that is not only based on the notation information of the word. The proposed system conceptualizes several items of personal information and news with this mechanism, and was able to quantitatively calculate the association between information and news. In the evaluation, the mean precision of the proposed method was 43.7%, which was considered to be a good effect.

Acknowledgment

This work was partially supported by JSPS KAKENHI Grant Number 16K00311.

References

[1] Kaufmann, M.: Proceedings of the Third DARPA Speech and Natural Language Workshop (1990)

[2] Y.Iida, K.Umedu, : Development of a Reservation System for Japanese Style Hotels using Aspect-Oriented Programming, The Journal of the Faculty of Science and Technology, Seikei University, Vol.43, No.1, pp.9-15 (2006)

[3] K.Tanigaki, G.Chkraborty, N.Shiratori, : Trainable Natural Language Interface for Intelligent Agent, The Special Interest Group Technical Reports of IPSJ, 1994-DPS-069, pp.73-78 (1995)

[4] Weizenbaum, J.: ELIZA — A Computer Program For the Study of Natural Language Communication Between Man and Machine, Communications of the Association For Computing Machinery, Vol.9, No.1, pp.36-45 (1965)

[5] R.Tokuhisa, R.Terashima,: An Analysis of 'Distinctive' Utterances in Non-task-oriented Conversational Dialogue, Transactions of the Japanese Society for Artificial Intelligence AI22, pp.425-435(2007)

[6] M.Inaba, N.Isomura, F.Toriumi, K.Ishii,: Evaluation method of Non-task-oriented dialogue system using Semantic Network, IPSJ SIG Technical Report, ICS2009(16), pp.29-34(2009)

[7] Y.Kimura, K.Araki, Y.Momouchi, K.Tochinai,: Spoken Dialogue Processing Method Using Inductive Learning with Genetic Algorithm, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, J84-D-II(9), pp.2079-2091(2001)

[8] NTCIR, : http://research.nii.ac.jp/ntcir/

[9] J.Okamoto, S.Ishizaki, : Construction of Associative Concept Dictionary with Distance Information, and Comparison with Electronic Concept Dictionary, Journal of Natural Language Processing, Vol.8, No.4, pp.37-54 (2001)

[10] H.Watabe, N.Okumura, 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)

[11] T.Araki, N.Okumura, H.Watabe, T.Kawaoka, : Dynamic Calculation Method of Degree of Association Considering the Common Attributes of Target Concepts, The science and engineering review of Doshisha University, Vol.48, No.3, pp.14-24 (2007)

[12] asahi.com, : http://www.asahi.com/

[13] mainichi.jp, : http://www.mainichi.jp/

[14] YOMIURI ONLINE, : http://www.yomiuri.co.jp/

[15] BIGLOBE Search Shunkan Ranking, :http://search.biglobe.ne.jp/ranking/

[16] Y.Tsuji, H.Watabe, T.Kawaoka, : The Method of Acquisition of The New Concept and Its Attribute Using The World Wide Web, The Japanese Society for Artificial Intelligence, JSAI04(0), pp.146-146(2004)

[17] E.Yoshimura, M.Imono, S.Tsuchiya, H.Watabe,:Computer-Generated Conversation Based on Newspaper Headline Interpretation, KES2010 LNAI(Lecture Notes in Artificial Intelligence) 6276, pp.400-409(2010)

A Fast Quantization Tree Based Image Retrieval Method

X.L. Wang[1], X. Wang[2], and A.N. Hou[2]
[1]School of Information Engineering, Changan University, Xi'an, Shaanxi, China
[2]School of Software Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi, China

Abstract - *Traditional content-based image retrieval technology expresses the content of each image by feature vectors. Then image retrieval process begins by calculating the similarity between the image to search and the images in the database in terms of the corresponding feature vectors, next ranks the images in the database by a descending order of similarity, and finally outputs the desired top ones. This method shows good accuracy and high efficiency when the image databases are not very big. For modern large image databases, however, these methods will not satisfy users' requirements for retrieval time and accuracy. To meet these challenges, in this paper, a new content-based image retrieval method is proposed which is similar to the Scalable Vocabulary Tree image retrieval method but with important variations. Experimental results show our method can more efficiently deal with larger image databases while with similar retrieval accuracy.*

Keywords: Content-based Image Retrieval, Visual Vocabulary, Vocabulary Tree, Clustering Algorithm, Random Quantization Tree

1 Introduction

The development of information technology gives an impetus to the development of multimedia technology, and image databases grow rapidly in an exponential scale. People begin to care about how to manage these images scientifically, legitimately and effectively, and how to identify needed images quickly and accurately from huge image databases. Therefore, image retrieval technology becomes an active research area.

Content-based image retrieval is a high-level learning task and can be pursued at different levels of visual properties such as appearance and shape etc.. Since recognizing an object by its shape is computationally much more expensive, researchers try to represent object by learning appearance features of objects, such as color and texture, in the form of local descriptors which are extracted from image local regions and form the basis of many appearance-based object discovery applications.

Published by David Lowe in 1999, Scale-Invariant Feature Transform (SIFT) is an algorithm in computer vision to detect and describe local features in images which are invariant to color, rotation, and translation etc. [1]. With its receptive field-like image operations, SIFT based image descriptors have opened up an area of research on image-based matching and recognition with numerous application areas. Being based on theoretically well-founded scale-space operations or approximations thereof, these approaches have been demonstrated to allow for robust computation of image features and descriptors from real-world image data.

For modern large databases, traditional sequential brute force retrieval methods are of low efficiency and, therefore, unable to meet the requirements of users. Different retrieval methods have been proposed to improve retrieval efficiency. For fast image retrieval, Scalable Vocabulary Tree (SVT) is a way of both improved retrieval accuracy and better retrieval efficiency. In this mechanism, SIFT descriptors as image characteristics are first extracted as the content features to depict an image. Next, the method of hierarchical *k*-means clustering algorithm is applied on the extracted image features to generate a visual word vocabulary tree. Finally, a hierarchical Term Frequency Inverse Document Frequency (TF-IDF) scoring using hierarchically defined visual words that form the vocabulary tree is proposed to allow much more efficient lookup of visual words. The image retrieval based on vocabulary tree saves time for image matching, especially for large image libraries. However, due to the "curse of dimensionality" problem of high-dimensional space, the establishment of the vocabulary tree consumes a considerable amount of time, and, therefore, can not meet the users' requirements. To partially circumvent this problem, through studying the vocabulary tree, we propose a random quantization tree (QT) method to retrieve similar images. The experimental results show that the random quantization tree is significantly better than vocabulary tree in the tree construction efficiency while maintaining similar retrieval accuracy to that of SVT.

The rest of the paper is organized as follows. The vocabulary tree method as a solution to the content-based image retrieval problem is described in Section 2. An introduction to our proposed quantization tree-based method is given in Section 3. Experiments are designed in Section 4 to explore the suitability of our method as a solution for the current image retrieval problem. Finally, conclusions are made in Section 5. Before proceeding, a brief introduction to the SIFT image descriptor is given in the below.

1.1 Scale-Invariant Feature Transform

To recognize any object in an image, interesting points (which are termed as keypoints in the SIFT framework) on an object can be extracted from a training image to provide a

feature description of the object, which can then be used to identify the object when attempting to locate it in a test image containing that object. To perform reliable recognition, it is important that the features extracted from the training image be detectable even under changes in image scale, noise and illumination.

Lowe's method for image feature generation transforms an image into a large collection of feature vectors, each of which is invariant to image translation, scaling, and rotation, partially invariant to illumination changes and robust to local geometric distortion. There are mainly four steps involved in the SIFT algorithm.

In the first step, that is, Scale-Space Extrema Detection, an image is convolved with Gaussian filters at different scales, and the difference of successive Gaussian-blurred (DoG) images is taken. Keypoints are then identified as local maxima/minima of the DoG images that occur at multiple scales. This is done by comparing each pixel in the DoG images to its eight neighbors at the same scale and nine corresponding neighboring pixels in each of the neighboring scales. If the pixel value is the maximum or minimum among all compared pixels, it is selected as a candidate keypoint. Scale-space extrema detection can produce too many keypoint candidates, some of which are unstable. In the second step, that is, Keypoint Localization, the SIFT algorithm discards low contrast keypoints and filters out those locating on edges. To do so, Taylor series expansion of the scale space is used to get more accurate location of extrema, and if the intensity at this extrema is less than a threshold value, it is rejected. Further, DoG has higher response for edges, so edges also need to be removed. By eliminating any low-contrast keypoints and edge keypoints, strong interest points remain. It can happen that keypoints have the same location and scale, but different directions. In the third step, that is, Orientation Assignment, an orientation can be assigned to each keypoint to achieve invariance to image rotation. In a neighbourhood taken around a keypoint location depending on the scale, the gradient magnitude and direction are calculated so as to generate an orientation histogram with 36 bins covering 360 degrees. The highest peak in the histogram is taken and any peak above 80% of it is also considered to calculate the orientation. Finally, in the fourth step, Keypoint Descriptor, a 16x16 neighbourhood around the keypoint is taken. It is divided into 16 sub-blocks of 4x4 size. For each sub-block, 8-bin orientation histogram is created. So a total of 128-bin values are available. It is represented as a vector to form a keypoint descriptor. Additionally, several measures are taken to achieve robustness against illumination changes and rotation etc.. These steps ensure that the keypoints are more stable for matching and recognition.

The SIFT descriptor is based on image measurements in terms of receptive fields [2][3][4][5] over which local scale invariant reference frames [6][7] are established by local scale selection [8][9][7]. These features share similar properties with neurons in the inferior temporal cortex that are used for object recognition in the primate vision [10]. A general theoretical explanation about this is given in the Scholarpedia article on SIFT [11].

Fig. 1. SIFT-based image matching

For content-based image retrieval, as illustrated in Fig.1, keypoints between the input image and the images in the databases are matched by identifying their similarity. However, for modern large databases, these methods do not scale well with the size of the image databases, and can not select a small number of matching images out of the databases in acceptable time. To speed up, Lowe used a modification of the K-d tree algorithm called the Best-bin-first (BBF) search method [12] that can identify the nearest neighbors with high probability using only a limited amount of computation. However, due to the "curse of dimensionality" problem, its search efficiency could degenerate to that of a sequential search in high-dimensional space. Therefore, rapid image retrieval algorithm is in immediate need.

2 Scalable recognition with a VT

Trees present an efficient way to index local image regions. Scalable Vocabulary Tree (SVT) is a two-stage recognition scheme that can handle a large number of objects and enable extremely efficient retrieval. The scheme builds upon popular techniques of indexing descriptors extracted from local regions which are hierarchically quantized in a vocabulary tree and scales efficiently to a large number of objects.

Being highly distinctive in performance evaluation [13], SIFT keypoints of objects are first extracted from a set of training images in large databases and stored. An object is recognized in a new image by individually comparing each feature from the new image to those from databases and finding candidate matching images based on similarity measure of their feature vectors.

In the offline training stage, SVT provides a more efficient training of the tree by defining a hierarchical quantization that is built by hierarchically *k*-means clustering on a large set of representative descriptor vectors. Instead of *k* defining the final number of clusters or quantization cells, *k* defines the branch factor (i.e., the number of children of each node) of the tree. More specifically, an initial *k*-means process is run on the training data, defining *k* cluster centers. The training data is then partitioned into *k* groups, where each group consists of the descriptor vectors closest to a particular cluster center. The same process is then recursively applied to each newly generated group of descriptor vectors, recursively

defining quantization cells by splitting each quantization cell into k new parts. The tree is determined level by level, up to some maximum number of levels, and each division into k parts is only defined by the distribution of the descriptor vectors that belong to its parent quantization cell.

In the online phase, each descriptor vector is simply propagated down the tree by comparing it to the k candidate cluster centers (represented by k children in the tree) at each level and choosing the closest one. This is a simple matter of performing k dot products at each level which is very efficient if k is not too large. The path down the tree can be encoded by a single integer and is then available for use in scoring. The nodes of an SVT are the centroids determined by hierarchical k-means clustering of database feature descriptors. The determination of the relevance of a database image to the query image is based on how similar the paths down the vocabulary tree are for the descriptors from the database images and for those from the query image. Using the SVT, information about a feature set is condensed to node visit counts. Suppose an SVT has N nodes. During the training phase, all feature descriptors extracted from the i^{th} database image are classified through the SVT, using a greedy nearest-neighbor search. After classification, it is known how many times, $n_i(j)$, the j^{th} node in the SVT is visited by the i^{th} database feature set, resulting in a vector of node visit counts $d_i = [n_i(1), n_i(2), \ldots, n_i(N)]$. Similarly, when a query feature set is received on the server, its descriptors are classified through the SVT, generating its vector of node visit counts $q = [n_q(1), n_q(2), \ldots, n_q(N)]$. The dissimilarity between d_i and q is given by,

$$D_i(d_i, q) = \left\| \frac{W \bullet d_i}{\|W \bullet d_i\|} - \frac{W \bullet q}{\|W \bullet q\|} \right\| \quad (1)$$

where W = diag $(w_1, \ldots, w_N)$ is a matrix used to assign different entropy-related weights to different nodes. The Euclidean distance can be the dissimilarity measure to use. The database feature sets with the smallest D_i values are further considered for a more elaborate comparison.

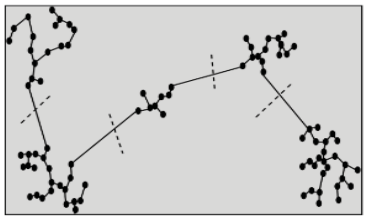
Scalable image retrieval systems have the merit of retrieval efficiency and usually give impressive results, but the resulting visual vocabulary representation usually faces two crucial problems. First of all, hierarchical quantization errors and biases exist in the generation of "visual words". Secondly, because of involving hierarchical quantization of local image descriptors, they are computationally expensive owing to the cost of hierarchical k-means clustering during training. Particularly, when the backend database is updated incrementally, the vocabulary tree model has to be re-generated entirely from the overall dataset. To lessen system computational cost, in the following, we describe an unsupervised optimization strategy in generating the hierarchy structure of visual vocabulary, which produces a more effective and adaptive retrieval model for large-scale search. By our approach, efficient and effective transfer of a retrieval model across databases of different sizes is feasible.

3 Scalable recognition with a QT

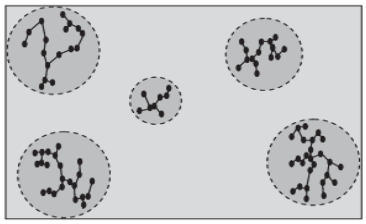
The most significant property of the SVT scheme is that the tree directly defines the quantization in the sense that the quantization and the indexing are fully integrated, essentially being one and the same. Different from SVT, our approach defines a visual vocabulary non-hierarchically, and then devises an approximate nearest neighbor search scheme in order to find similar visual words efficiently.

3.1 Visual vocabulary generation

In order to understand and manipulate the world, labeling local descriptors extracted from training images to target categories through machine learning methodology is often a major task in the building of intelligent systems. Each feature vector appears as a point in the feature space and patterns pertaining to different classes will fall into different regions of the feature space. The label-learning process is to partition the feature vectors into sensible clusters based on some properties in common. To ease the labeling task by grouping unlabeled feature vectors, unsupervised learning (also known as clustering in data mining literature) have gained prominence and many approaches have been developed for it, including hierarchical, partition-based, density-based, model-based, and graph-based approaches. An extensive study and comparison of the state-of-the-art unsupervised object discovery techniques are reported in [14]. As illustrated in Fig.2, being a graph-based approach, MST-based clustering algorithms can outperform the classic methods such as k-means clustering algorithm when the boundaries of the clusters are irregular. To this end, a fast MST-based clustering algorithm, proposed in [15], has obtained impressive segmentation results for challenging image data obtained from an indoor environment [16]. We use this method in this research for an outdoor environment based on SIFT image local descriptors.



(a) A minimum spanning tree (MST)



(b) MST-based clusters

Fig. 2. Illustration of MST-based clustering

3.2 Nearest neighbor based matching

Matching an object model against object data from some other source, object recognition capability is fundamental in many practical robotic systems and can be performed by nearest neighbor search. An image retrieval algorithm based on the plane nearest neighbor matching of feature sets is described in the following as,

1) label manually all U training images in a database with known W categories, and extract their SIFT descriptors to form two sets, U feature sets, $F = \{F_i, 1 \le i \le U\}$, and the corresponding label set, $L = \{ L_i, 1 \le i \le U \mid L_i \in L_j, 1 \le j \le W\}$;
2) given a coming new image, extract its SIFT feature set of size V, $T = \{t_m, 1 \le m \le V\}$;
3) for each SIFT feature, t_m, calculate its Euclidean distance with every feature in F, if its nearest neighbor is in F_o, $1 \le o \le U$, assign F_o's label, L_o, to t_m;
4) repeat Step 3) until all the feature vectors are processed;
5) a label set is obtained for the query image, and use the majority of the labels in the set as the label for this image.

Though simple and elegant, nearest neighbor search in the presence of a large database of object models is not very efficient. To support efficient image matching, multidimensional index structures, especially various types of tree-based index structures, have been developed to reduce the computation cost.

3.3 Quantization tree

Hierarchies are a natural way to organize concepts and data. For a long time, research on object recognition has aimed at building hierarchical models [17]. As a very interesting alternative to *k*NN search algorithm, decision tree construction is a classic technique for classification of a set of data records of known classes. Standard splitting rules usually involve a single attribute (e.g., applying a threshold to obtain locally optimal decisions at each node) [18]. Branches from the root to the leaf nodes represent conjunctions of features that lead to those class labels. To organize objects in the real world into a hierarchy based on their semantic meaning, in our approach, a tree-structured vector quantization method is used to generate a 3-way approximate nearest neighbor search tree, which is illustrated in Fig.3. Given the obtained training set, a tree root node exists at the first level which includes all the indices to the database feature elements. Then an initial set of 3 representative patterns, called centers, are randomly selected from the elements of the root as its children cluster centers (or tree nodes) at the second level, and the whole set of the elements is clustered into three subsets by assigning each feature vector to its closest representative cluster center according to a selected distance measure (say Euclidean distance). At the third level, for each of the 3 clusters obtained at the second level, three feature vectors are selected randomly as its children cluster centers and elements are clustered correspondingly, resulting in 9 cluster tree nodes at the level. This procedure continues until either all the elements in a node belong to the same object class (a pure node) or the number of the elements in a node is below some limit, say, one hundred. Every feature vector in the leaf nodes has a percept landmark associated with it.

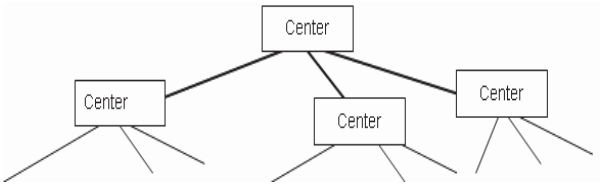


Fig. 3. Illustration of database tree structure

Next, to search through the tree, given a new feature vector, its distances to the randomly chosen cluster centers at the second level are calculated and the winner is the center that the feature vector is nearest to. At the third level, the feature vector's distances with the three children centers of the winner at the second level are computed and the corresponding new winner is selected. This procedure continues until coming to a leaf node. If it is pure, then stop. Otherwise, do a nearest-neighbor search, and the winner is the training feature vector which gives minimum distance according to the chosen metric. Our final scoring mechanism follows the same procedure as that of the SVT's.

4 Experiments

In this section, we present the results of experiments conducted to evaluate the performance of our proposed recognition method. The method was tested by performing queries on a database consisting of 2794 images with 9 known visual categories. A subset of 114 images is chosen as the training set, for which SIFT descriptors are extracted and clustered. 72 images are chosen as the test set. The database is queried with every image in the test set. We implemented all the algorithms in C++. All the experiments were performed on a computer with Intel Core i5-3470 3.20 GHz CPU and 4 GB RAM. The operating system running on this computer is Windows 7. We use the timer utilities defined in the C standard library to report the CPU time.

Fig.4 shows some typical scenes of images used in the experiments. The query results by our method and the corresponding query efficiency are compared with those of two other methods, namely, brute force (BF, named for nearest neighbor based matching) method and SVT, are summarized in Fig.5 and Table 1, respectively. For retrieval effectiveness performance measure, we focus on the right images returned at the top for the query and especially how many percent of the other images in each category are found perfectly.

Fig.5 shows the retrieval results for the test images. There are nine sets of three lines in the graph, representing

the performances of three methods on nine categories of the images. In each three-vertical-line set, the left vertical line denotes the results for brute force method, the middle line denotes the results for our method, and the right line denotes the results for the SVT method. Clearly, the retrieval effectiveness of our algorithm agrees with the other two methods for 5 out of 9 cases and has retrieval accuracy rate above 80% for 8 out of 9 cases. Overall, it can be seen that our algorithm performs similarly well as the other two retrieval methods.

Fig. 4. Some typical scenes of images used.

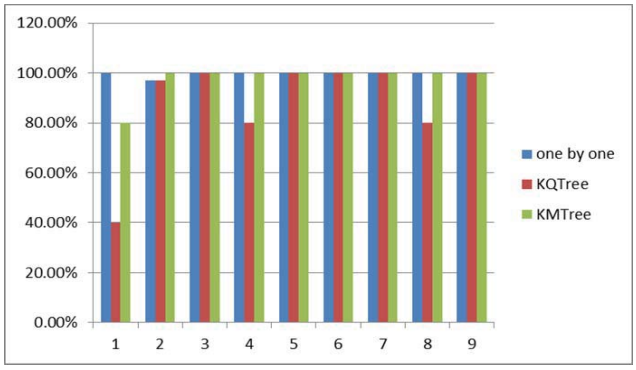


Fig. 5. Percentage of detected right images for 72 test query images over the nine categories of the image database.

The running time performances of our algorithm, the brute force method (BF), and the SVT method are summarized in Table 1. There are three rows in the table, representing the average running time performance of tree construction, single query image retrieval, and their sum, respectively, over 10 trial runs of three algorithms. From the table, we can see that our algorithm outperforms the other two algorithms and is faster than SVT by a factor of 30 in running time performance.

Table 1. Running time performances.

	BF (hour)	SVT (s)	KQTree (s)
Tree Construction	0	15122	465
Image Retrieval	4	4	3
Total	4	15126	468

5 Conclusions

Content-based image retrieval demands for efficient search structures and algorithms. Design of searching methods that scale well with the size of the database and the dimensionality of the data is a challenging task. Trees present an efficient way to index local image regions, making the nearest neighbor search more efficient by pruning. A fast image retrieval approach with an indexing scheme significantly more efficient than the state-of-the-art SVT method has been presented. The approach is built upon a quantization tree that hierarchically organizes labeled descriptors from image keypoints. Experimental results have demonstrated the improved efficiency of the proposed method while with a similar effectiveness to that of the state-of-the-art SVT.

Acknowledgments

The authors would like to thank the Chinese National Science Foundation for its valuable support of this work under award 61473220 and all the anonymous reviewers for their valuable comments.

References

[1] D.G. Lowe. "Object recognition from local scale-invariant features". In: Proceedings of the International Conference on Computer Vision, pp.1150-1157, Sept 1999.

[2] T. Lindeberg. "A computational theory of visual receptive fields". Biological Cybernetics, 107(6):589-635, Sept 2013.

[3] T. Lindeberg. "Generalized axiomatic scale-space theory". Advances in Imaging and Electron Physics, Elsevier, 178:1-96, June 2013.

[4] T. Lindeberg. "Invariance of visual operations at the level of receptive fields". PLoS ONE, 8(7):e66990, July 2013.

[5] T. Lindeberg. "Scale selection". Computer Vision: A Reference Guide, (K. Ikeuchi, Editor), Springer, pp.701-713, June 2014.

[6] B. Sirmacek and C. Unsalan. "Urban area and building detection using SIFT keypoints and graph theory". IEEE

Transactions on Geoscience and Remote Sensing, 47(4): 1156–1167, April 2009.

[7] S. Se, D.G. Lowe, and J. Little. "Vision-based mobile robot localization and mapping using scale-invariant features". In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp.2051-2058, 2001.

[8] T. Lindeberg. "Scale invariant feature transform". Scholarpedia, 7(5):10491, 2012.

[9] T. Lindeberg and L. Bretzner. "Real-time scale selection in hybrid multi-scale representations". In: Proceedings of the 4th International Conference on Scale-Space 2003, Isle of Skye, UK, Springer Lecture Notes in Computer Science, 2695:148-163, June 2003.

[10] J. Beis and D.G. Lowe. "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces". In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, San Juan Puerto Rico, pp.1000-1006. June 1997.

[11] L. Bretzner, I. Laptev, and T. Lindeberg. "Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering". In: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, Washington, DC, USA, pp.423-428, May 2002.

[12] S. Lazebnik, C. Schmid and J. Ponce. "Semi-local affine parts for object recognition". In: Proceedings of the British Machine Vision Conference, 2:779-788, 2004.

[13] J.L. Bentley. "Multidimensional binary search trees in database applications". IEEE Transactions on Software Engineering, 5(4):333–340, July 1979.

[14] T. Tuytelaars, C.H. Lampert, M.B. Blaschko, and W. Buntine. "Unsupervised object discovery: a comparison". International Journal of Computer Vision, 88(2):284-302, June 2010.

[15] X. Wang, X.L. Wang, and D.M. Wilkes. "A divide-and-conquer approach for minimum spanning tree-based clustering". IEEE Transactions on Knowledge and Data Engineering, 21(7):945–958, July 2009.

[16] X. Wang, M. Tugcu, J.E. Hunter, and D.M. Wilkes. "Exploration of configural representation in landmark learning using working memory toolkit". Pattern Recognition Letters, 30(1): 66-79, Jan 2009.

[17] K. Fukushima. "Neocognition: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". Biological Cybernetics, 36(4):193-202, 1980.

[18] J.R. Quinlan. "C4.5: Program for Machine Learning". Morgan Kaufmann, 1992.

A Seat Occupancy/Vacancy Detection Method Using Smartphone and High-Resolution Infrared Sensors in a Non-territorial Office

Daichi Terai[1], Mitsunori Miki[2], Sota Nakahara[1], Naoki Kawata[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 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 and energy saving deterioration the Intelligent Lighting System. On the other hand, in a non-territorial office, however, as any individual's seat is not fixed, it is not easy to manage information on which seat workers occupy. Then We propose the method to detect a status of seat occupancy/vacancy automatically using a smartphone and high-resolution infrared sensors.*

Keywords: infrared sensor, intelligent lighting system, smartphone, position estimation

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[4]. 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. As its effectiveness was acknowledged, demonstration tests have been conducted in real office environments.

If a seat is not occupied by a worker, the Intelligent Lighting System judges that it does not need brightness. By dimming or turning off lighting for an area around a seat unoccupied by a worker in such a manner as not to affect working space of other workers, the system realizes even higher energy saving. The results of demonstration tests at real offices, however, showed that workers did not appropriately change the occupied/vacant status of their seat. Those results confirmed that there were lighting fixtures which provided brightness more than required even though there was no worker.

We thus proposed a method to automatize the toggling of the occupied/vacant status of a worker's seat in an office with fixed seats by using a high-resolution infrared sensor[7]. At present, the Intelligent Lighting System expanded the scope of its application and being introduced into non-territorial offices. A non-territorial office refers to a type of office designed by a method whereby individuals do not have their dedicated seat and equipment and fixture are shared by multiple people. As opportunities for interaction between workers in an office increase and each worker can work at a place he/she prefers in a non-territorial office, it is expected to improve intellectual productivity and comfort. Also partly because of its excellent space saving property, it has been introduced into many companies. In a non-territorial office, however, as any individual's seat is not fixed, it is not easy to manage information on which seat workers occupy. This study thus examines a method for managing information on whether workers are seated or not in a non-territorial office in which the Intelligent Lighting System has been introduced by enabling the identification of a worker by using a smartphone each worker carries with him/her.

The proposed method determines whether each worker is seated or not and identifies the location of a seat the worker has taken. Whether each worker is seated or not is determined on the basis of his/her smartphone's angle to the horizontal plane calculated by an accelerometer embedded in the smartphone. The location of the seat in which each worker is seated is identified by using the accelerometer and geomagnetic sensor embedded in a smartphone in combination with a high-resolution infrared sensor. Combining a smartphone and a high-resolution infrared sensor enables the automatic management of information on whether workers are seated or not under the Intelligent Lighting System introduced in a non-territorial office. If the target illuminance for each worker is registered with a database, the worker does not have to set his target illuminance each time, and the Intelligent Lighting System will operate quickly to adjust illuminance to the worker's preference. In addition, managing information on which worker is seated in which seat in

Table 1: Specification of the infrared sensor with high resolution

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

a non-territorial office is expected to increase opportunities for interactions.

2. High-Resolution Infrared Sensor

A high-resolution infrared sensor is a 16 × 16 noncontact MEMS 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.

If a high-resolution infrared sensor is installed on the ceiling 3 m from the floor, assuming that the distance from the floor to the desktop is 1 m, the distance from the desktop to the ceiling is 2 m. The detection range will then be 3.6 m x 3.6 m. This detection range is divided into 256 sections for 16 elements × 16 elements, and the average temperature is output for each section. It is then possible to identify the number and position of thermal sources on the basis of the temperature distribution in the detection range of the high-resolution infrared sensor.

Difference between existing infrared sensors and high-resolution infrared sensors is that the latter can identify the approximate number of persons. Existing infrared sensors distinguished between the following two states: whether there are any human beings in its detection range or not. In contrast, using a high-resolution infrared sensor, which can grasp a certain number of heat sources, has enabled the simultaneous detection of multiple persons. As it detects heat sources in the detection range, however, it also detects heat emitted by PCs and OA devices. Therefore, if a high-resolution infrared sensor is actually introduced, as there are many heat sources such as PCs and OA devices, it is conceivable that such heat sources are mistakenly detected as human beings. It is thus necessary to distinguish human beings from PCs and OA devices based on temperature data obtained.

Temporal difference in temperature is used to distinguish human beings from other heat sources. By comparing the current temperature data with those before a certain period of time, human beings and other heat sources are distinguished from each other.

Since, however, a high-resolution infrared sensor is a temperature sensor, it cannot determine which worker was seated in a particular seat in a non-territorial office. Therefore, in this study, a smartphone carried by each worker is used to identify an individual.

3. Detection method of a seat occupancy/vacancy using a smartphone and high-resolution infrared sensors

3.1 Outline of the proposed method

This study proposes a method using a smartphone-embedded accelerometer to determine whether each worker is seated or not and smartphone-embedded accelerometer and geomagnetic sensor, together with a high-resolution infrared sensor, to identify the seat, if any, in which a worker is seated. If a worker takes a seat, the point of time at which his/her seated/unseated status changes and the point of time at which a high-resolution infrared sensor detects a human being are compared to determine which seat the worker is seated in. If a worker leaves a seat, it is detected by the smartphone's determination of whether the worker is seated or not.

In addition, if multiple workers take their seat at the same time, the position of the seat which a worker is seated cannot be identified by a smartphone's determination of whether the worker carrying the smartphone is seated or not together with the identification of seat locations by a high-resolution infrared sensor. Therefore, the position of the seat in which a worker is seated is identified by estimating which worker is located in an office by using an accelerometer and a geomagnetic sensor embedded in a smartphone to estimate the location of a worker who carries it.

3.2 Judging method of a seat occupancy/vacancy status

A method is thus proposed for detecting whether a worker is seated or not by using a smartphone. For the method in this study for detecting whether a worker is seated or not, it is assumed that a smartphone is put in a trousers pocket, where it is most frequently put.

Under the proposed method, the smartphone terminal's angle to the horizontal plane is measured to determine whether a worker is seated or not. The smartphone terminal's angle to the horizontal plane used to determine whether a worker is seated or not is calculated by using data measured by a 3-axis accelerometer embedded in the smartphone. Formula (1) gives the calculation formula.

$$\theta = arctan(Accy, Accz) \quad (1)$$

$Accy$: Acceleration of the smartphone terminal to the longitudinal direction of horizontal plane

$Accz$: Acceleration of the smartphone terminal to the vertical direction

(2)

The angle is calculated by averaging 25 immediate and consecutive points. If a worker is not seated, he/she is judged to be walking or standing and therefore his/her smartphone terminal's angle to the horizontal plane is close to 90 degrees. On the other hand, if he/she is seated, his/her smartphone terminal's angle to the horizontal plane is close to 0 degrees. A threshold is thus established, and the angle greater than the threshold is defined to correspond to a worker not seated, and the angle less than the threshold is defined to correspond to 0 degree, to determine whether a worker is seated or not.

3.3 Detection method of the position of a seat occupancy

PDR (pedestrian dead reckoning) is used to estimate the position of a worker in an office. In this study, the position of a worker is estimated by calculating three kinds of data — the number of steps, the direction a worker is headed to, and step length — through feature value calculation using data measured by an accelerometer and a geomagnetic sensor embedded in a smartphone. The distance moved is calculated by simply multiplying the number of steps detected and step length. Using data for the moving direction, the direction and distance of movement from the immediately preceding position are estimated.

In this paper, step length is calculated by the height of a worker. Formula (3) calculates step length. Step length calculated by this formula is said to be the average human step length.

$$Stride[m] = Height[m] - 1 \tag{3}$$

A method proposed by Muramatsu et al. is used to obtain data for the number of steps. The square root of the sum of the squares of the acceleration for each axis measured by the accelerometer is calculated to obtain a composite acceleration, the maximal value of whose waveform is detected as a single step. In this paper, the composite acceleration is calculated by taking the square root of the sum of the squares of the acceleration for each axis of 3-axis acceleration measured by a smartphone. The 25-point moving average of the composite acceleration is calculated, and the maximum value of the moving average waveform is detected as a single step. Formula (**??**) gives the calculation formula.

$$Acc = \sqrt{Accx^2 + Accy^2 + Accz^2} \tag{4}$$

Acc : Triaxial synthetic acceleration

$Accx$: Acceleration of the smartphone terminal to the lateral direction of horizontal plane

PDR, however, entails an error between an actual position after movement and an estimated position. The position of the seat in which a worker is seated is thus identified by combining PDR with person detection by a high-resolution infrared sensor. Figure 1 shows an example of detecting a worker seated by temperature change.

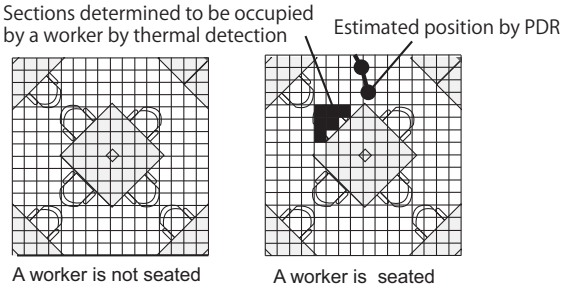


Figure 1: The detection of the position of the seat using comparison of temperature over time

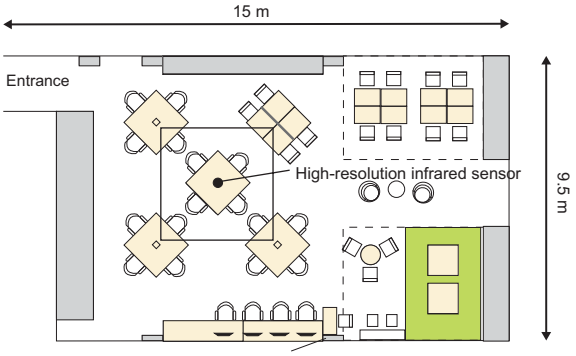


Figure 2: Experimental environment

The detection of the position of the seat in which a worker is seated by a high-resolution infrared sensor uses comparison of temperature over time. A high-resolution infrared sensor cannot distinguish between a human being and a PC. Therefore, temperature change is used to distinguish between them by taking a human being as a moving heat source and a PC as a non-moving heat source.

If the status of a worker changes from "not seated" to "seated" in the decision of whether the worker is seated or not by a smartphone, the position of the seat in which a worker is seated is identified by comparing the temperature data at the time measurement by the high-resolution infrared sensor is taken and the temperature data before a worker is seated.

4. Verification experiment

4.1 Experimental outline

This section verifies the effectiveness of the proposed method with regard to the following three points.

- Precision in determining whether a worker is seated or not by using a smartphone-embedded accelerometer
- Precision in estimating the position of a worker in office by using smartphone-embedded accelerometer and magnetic sensor
- Precision in identifying the position of the seat in which a worker is seated by using temporal difference in temperature

The verification experiment was conducted in a non-territorial office simulating environment in the Room KC104 in Kochikan, Doshisa University. The plan for the experimental environment is shown in Figure 2.

In this experiment, a high-resolution infrared sensor was installed in the center of the ceiling in the experimental environment, with a camera installed beside it. There were four seats for work at the desk directly under the detection range of the high-resolution infrared sensor.

A subject with a smartphone in his/her trousers pocket enters Kochikan, walks into Room KC104, and takes a target seat in the room. The Wi-Fi radio field intensity is measured by the smartphone. The trajectory of the movement of the subject is calculated by PDR with the smartphone from the time the Wi-Fi access point in Room KC104 was detected to estimate the subject's position. Position estimation ends at the time the subject is seated. The method for calculating the trajectory of movement is as previously described.

Since it takes several seconds to get connected to Wi-Fi, a large error in the starting point is considered to result. The trajectory of movement is thus compensated by comparing the trajectory calculated and map information including information on the passage and the desk in the experimental environment. A two-dimensional coordinate is superposed on the experimental environment, and the point where the moving direction changed is calculated for compensation on the basis of changes in x and y coordinates for each position where the number of steps is detected.

The subject works at a PC for an arbitrary time after taking the target seat and then leaves the seat. The experimented was conducted with four subjects, with 100 trials performed in total.

4.2 Precision in determining whether a worker is seated or not using a smartphone

A preliminary experiment was conducted to determine the threshold of a smartphone's angle to the horizontal plane for determining whether a worker is seated or not. A smartphone's angle to the horizontal plane was calculated by measuring 3-axis acceleration of the smartphone terminal when a person carrying it with him/her was walking and seated. Subjects were instructed to walk and take seat in their normal posture.

Figure 3 shows the result of measurement. Based on this result, the threshold for determining whether a worker is seated or not was set to 40.0 deg.

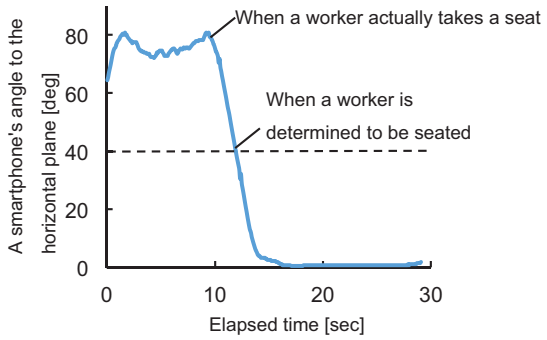


Figure 3: Angle measurement of the smartphone terminal to the horizontal plane

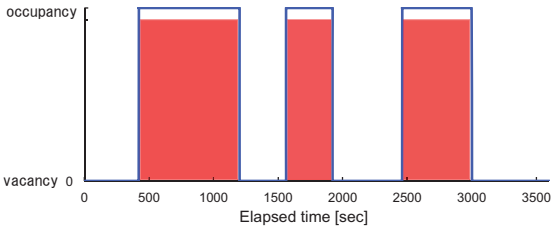


Figure 4: Result of the judge of a seat occupancy/vacancy status

The status of the subject as to whether he/she was seated or not was compared with photos taken by the camera installed on the ceiling of the experimental environment. Whether the subject was seated or not was determined at an interval of 0.1 second. Figure 4 shows the result of the experiment from 0 to 3,600 seconds after the start of the experiment.

As a result of comparison, it was confirmed that whether the subject is seated or not was determined correctly 100 % when a smartphone was put in his/her trousers pocket. There was a delay of about 1 second from the time a subject actually takes a seat to the time the smartphone determines that the subject is seated. This is considered to be attributed to the fact that the smartphone's angle to the horizontal plane is calculated as the average for 25 consecutive points. A delay of 1 seconds does not matter from a power consumption saving perspective in managing information on whether workers are seated or not. Therefore, if a smartphone is in a trouser pocket of a subject, it can be determined precisely whether he/she is seated or not. Hence the proposed method for determining whether a worker is seated or not is effective.

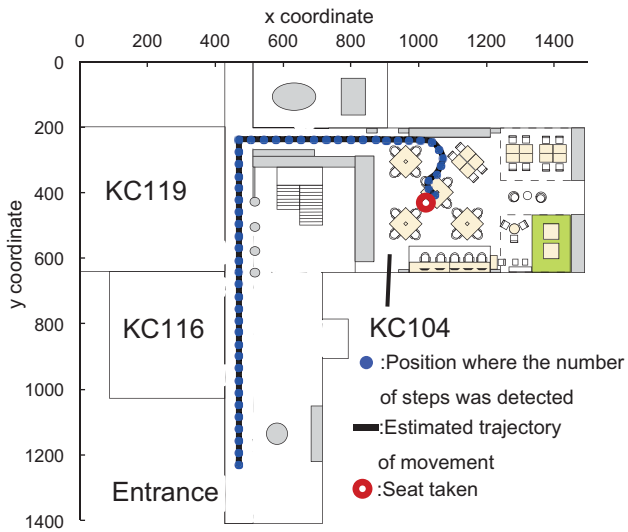


Figure 5: Result of estimating the position of a moving subject after correction

Table 2: The result of determining the position of the subject's seat

	error[m]
Max	1.13
Min	0.06
Ave	0.46

	Specific rate[%]
seat	68.0
area	98.0

4.3 Precision in estimating the position of a worker in office using a smartphone

The precision in estimating the position of a worker using a smartphone was verified, and an experiment was conducted to verify whether the seat taken by a worker can be identified or not. These are evaluated below. Figure 5 shows an example of estimating the position of a moving subject.

A seat taken by the subject is determined on the basis of the result of estimating the position of the subject. The nearest seat to the estimated position of the subject was determined to be taken by the subject in accordance with the proposed method. The distance between the seat which the subject actually took and the estimated position of the subject was evaluated as an error. The detectable range of the high-resolution infrared sensor was defined as the seatable area. Table 2 shows the result of determining the position of the subject's seat.

Table 2 shows that it is not easy to identify the position of the seat in which the subject is seated only by PDR with a smartphone. The rate of identifying an area in which the subject took a seat on the basis of the position estimation is 98 %. Therefore, the seatable area can be identified at high precision by position estimation using PDR. The seatable

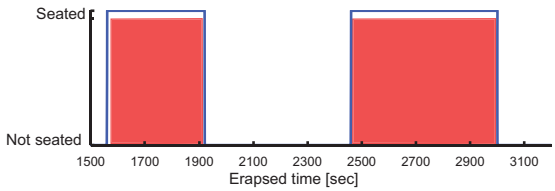


Figure 6: The status of seat occupancy detection using high-resolution infrared sensors

area cannot be identified at the rate of about 2 %. Since, however, the maximum error is 1.13 m, the estimated area is adjacent to the seat which the subject actually took. Therefore, it can be estimated on the basis of the time at which a smartphone determines that the subject has left the seat by synchronizing the smartphone with the high-resolution infrared sensor.

4.4 Precision in identifying the position of the seat using high-resolution infrared sensors

It is not easy to identify the seat which a worker takes by position estimation using a smartphone. Therefore, the seat position is identified by using a high-resolution infrared sensor installed in the identified seatable area. 256 sections of the detection range of the high-resolution infrared sensor were superposed on the positions of seats, and 4 sections were assigned in advance to one seat as seat sections.

Temporal difference in temperature was used to identify the position of a seat in which a subject is seated. Temporal difference in temperature was obtained by measuring the detectable range every 1 second, and data measured were compared with those 5 seconds before. The section for which temperature difference was 0.6 ℃ or greater was determined to be an occupied section. The method described above for determining whether a subject is seated or not using a smartphone embedded accelerometer was used to detect that a subject has left a seat. This is because, if the position of the seat in which a worker is seated has been identified, it is not required to identify the position by the high-resolution infrared sensor as a smartphone the worker carries with him/her is used. If 2 or more of seat sections assigned to one seat are determined as occupied, it is determined that the seat has been taken.

The experiment was shot by the camera installed beside the high-resolution infrared sensor and compared with positions detected as occupied by the high-resolution infrared sensor. Figure 6 shows the status of seat occupancy detection from 1,500 to 3,100 seconds after the start of the experiment.

The result of the experiment shows that the detection rate was 100 % although seat occupancy was detected several seconds earlier than the time the subject actually took a seat because the high-resolution infrared sensor detects heat before the subject is seated. This result confirmed that the

position of the seat in which a worker is seated can be identified by obtaining temporal difference in temperature using a high-resolution infrared sensor.

5. Conclusion

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 and energy saving deterioration the Intelligent Lighting System. On the other hand, in a non-territorial office, however, as any individual's seat is not fixed, it is not easy to manage information on which seat workers occupy. Then We proposed the method to detect a status of seat occupancy/vacancy automatically using a smartphone and high-resolution infrared sensors.

If the target illuminance for each worker is registered with a database, the worker does not have to set his target illuminance each time, and the Intelligent Lighting System will operate quickly to adjust illuminance to the worker's preference. Therefore the system detects the occupied/ unoccupied status automatically and improve the energy saving performance in the Intelligent Lighting System. In addition, managing information on which worker is seated in which seat in a non-territorial office is expected to increase opportunities for interactions.

It is necessary, however, to expand the scope of application by estimating how a smartphone is held when it is held in hand or placed in a bag.

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] Mitsunori Miki, "An Intelligent Lighting System and the Consortium for Smart Office Environment", The Japanese Society for Artificial Intelligence, vol.22, no.3, p399-410, 2007

[4] 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

[5] Keiko Ono, Mitsunoti Miki, Masato Yoshimi, Tatsuo Nishimoto, Tetsuya Omi, Hiroshi Adachi, Masatoshi Akita and Yoshihiro Kasahara, "Development of the Intelligent Lighting System using LED Ceiling Lights into an Actual Office", The transactions of the Institute of Electrical Engineers of Japan. A, A publication of Fundamentals and Materials Society, vol.131, no.5, pp.321-327, 2011

[6] Mariko Suzuki, Mitsunori Miki, Shingo Tanaka, Masato Yoshimi, Akihiko Nakagawa, Atsuko Saitou and Maiko Fukuda, "Construction of Intelligent Lighting System Using In-Office Frames", The Institute of Electronics, Information and Communication Engineers, vol.95, no.3, pp549-558, 2012

[7] Daichi Terai, Mitsunori Miki, Katsuya Ito, Kohei Yamaguchi, Hiroto Aida, "Verification of a Seat Occupancy/Vacancy Detection Method Using High-Resolution Infrared Sensors and the Application to the Intelligent Lighting System", ICAI2015, vol.2, pp522-528, 2015

Measuring Musical Rhythm Similarity: Edit Distance versus Minimum-Weight Many-to-Many Matchings

Godfried T. Toussaint[1], **and Seung Man Oh**[2]
[1]Computer Science Program, New York University Abu Dhabi, UAE
[2]Freelance IT Developer, Seoul, Korea

Abstract— *Musical rhythms are represented as binary symbol sequences of sounded and silent pulses of unit-duration. A measure of distance (dissimilarity) between a pair of rhythms commonly used in music information retrieval, music perception, and musicology is the edit (Levenshtein) distance, defined as the minimum number of symbol insertions, deletions, and substitutions needed to transform one rhythm into the other. A measure of distance often used in object recognition is the minimum-weight many-to-many matching distance between the object's features. These two approaches are compared empirically, in terms of how well they predict human judgments of musical rhythm similarity, using a real-world family of Middle-Eastern rhythms.*

Keywords: musical rhythm, perception, edit distance, many-to-many matchings, similarity measures, Hungarian algorithm

1. Introduction

For a variety of applications such as Music Information Retrieval (MIR) [22], automated music generation and composition [23], phylogenetic analysis of rhythms [21], [16], palylist generation [2], and music perception [4], it is desirable to have a mathematical measure of rhythm similarity (or distance) that predicts, or correlates well with, human judgments. A plethora of distance measures exists for computing the dissimilarity of two pieces of music that depend on how music is modeled. The present study is concerned with symbolically notated musical rhythms (rather than acoustic inputs). Furthermore, rhythms are represented in their most skeletal form as sequences of binary symbols denoting unit duration pulses that are either sounded (called *onsets*) or silent (called *rests*). For example, the rhythm used by Steve Reich in his piece titled *Clapping Music* [24] is represented by [x x x . x x . x . x x .], where 'x' and '.' denote, respectively, a sounded and a silent pulse.

1.1 The Edit Distance

A distance measure between a pair of rhythms commonly used in computational music is the *edit* (also Levenshtein [5]) distance, defined as the minimum number of symbol insertions, deletions, and substitutions needed to transform one rhythm into the other [18], [4]. An insertion of a sounded or silent pulse lengthens the duration of the rhythm, a deletion shortens its duration, and a substitution (reversal) of a sounded pulse by a silent pulse (and vice-versa) leaves the rhythm duration unaltered. Consider the two binary sequences A and B in Figure 1, expressed in box-notation, in which an empty box denotes a silent pulse and a filled box a sounded pulse. There are many possible series of edits that can transform A into B. One possibility is to delete silent pulses 8, 9, 10, and 11, followed by reversals of pulses 0, 1, 2, 5, 6, and 7, yielding a total of 10 edit operations. However, the minimum possible number of edits is 5, and may be obtained as illustrated in the figure. Sequence A* is obtained by deleting sounded pulse 0 in sequence A. Sequence A** is obtained by a reversal of pulse 7 in sequence A*, from a silent pulse to a sounded one. Finally, sequence B is obtained by deleting the last three silent pulses (8, 9, and 10) in sequence A**.

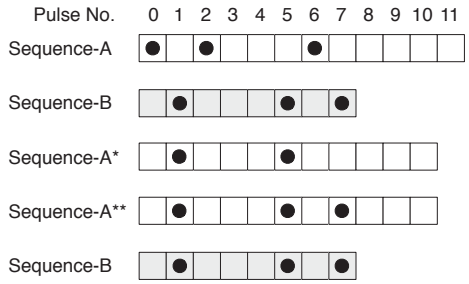


Fig. 1: The edit distance between sequences A and B is 5.

1.2 The Many-to-Many Matching Distance

A measure of distance frequently used in object recognition is the weight of the *minimum-weight many-to-many* matching between the features of one object and the features of the other [7]. Distances based on many-to-many matchings have also been applied to measure the similarity of two origami crease patterns corresponding to two different folded paper objects [10], to measure the difference between musical chords [11], and to compare relational data [6]. The many-to-many matching is a generalization of the simpler *one-to-one* and *many-to-one* matchings [12]. If two rhythms A and B have the same number of onsets, then the one-to-one matching assigns the i-th onset of A to the i-th onset of B, and the overall distance between $A = (a_1, a_2, ..., a_n)$ and $B = (b_1, b_2, ..., b_n)$, where the a_i and b_i are the x-coordinate

values of the onsets, is the sum over all i of the absolute values of the differences between the pulse locations of the corresponding i-th onsets, given by the equation:

$$Dist(A, B) = \sum_{i=1}^{n} |a_i - b_i| \quad (1)$$

In this case the one-to-one minimum-weight matching distance is equivalent to the *swap* distance [16]. A *swap* is an interchange of a sounded and silent pulse that are adjacent to each other. For instance, in Figure 2 in order to move the sounded pulse in location 2 to location 13, a total of 11 swaps are necessary, which is equivalent to $|13 - 2|$. When A and B have an unequal number of onsets then the one-to-one matching distance finds the minimum-weight perfect matching, thus ignoring the extra onsets of the denser rhythm [3]. If it is desired to assign all the onsets of the denser rhythm to the onsets of the sparser rhythm, with the constraint that every onset of the sparser rhythm must be matched to at least one onset of the denser rhythm, then the matching is a many-to-one matching, and the minimum-weight many-to-one matching is called the *directed swap* distance in computational music [8], and the *restriction scaffold assignment* distance in computational biology [9]. These two measures have achieved limited success in their applicability to accurately measure musical rhythm similarity because there are cases in which they give counter-intuitive and unsatisfactory results. One example that illustrates their weakness is shown at the top of Figure 2. Both the one-to-one and the many-to-one distance measures assign onset No. 2 of rhythm A to onset No. 13 of rhythm B, yielding a large distance of $1 + 11 + 1 = 13$, violating the Gestalt principle of proximity. Indeed, experimental results have provided evidence that the edit distance yields higher correlations with human judgments than these two matching distances [19]. In fact, the edit distance also has been shown to perform better than statistical features of the inter-onset-histograms of the rhythms [1], and better than global structural music-theoretical features of the rhythms [25]. Thus, the evidence garnered to date suggests the hypothesis that the edit distance is superior to all other measures of rhythm similarity with respect to correlation with human judgments.

In this paper the edit distance is compared empirically with the minimum-weight many-to-many matching distance, in order to test the above hypothesis, using a family of Middle-Eastern rhythms. A minimum-weight many-to-many matching is a matching of minimum cardinality that assigns each onset of one rhythm to at least one onset of the other rhythm, and vice-versa. The many-to-many matching between rhythms A and B seeks the minimum weight matching in a complete bipartite graph in which the nodes correspond to the onsets of the rhythms, and an edge is inserted between an onset of rhythm A and an onset of rhythm B such that its weight is equal to the absolute value

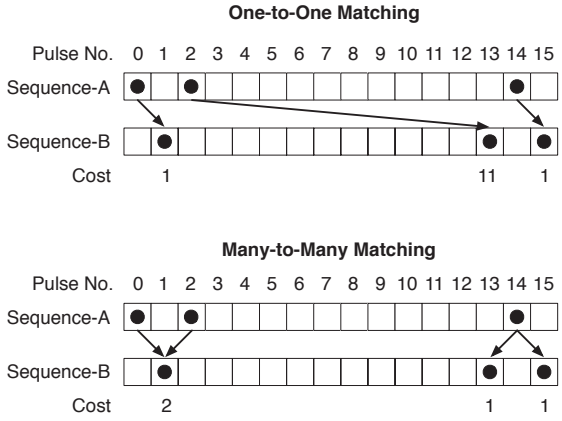


Fig. 2: The one-to-one (top) and many-to-many (bottom) minimum-weight matchings between sequences A and B.

of the distance (duration) between the two onsets. As the example at the bottom of Figure 2 illustrates, the many-to-many matching reflects more faithfully the Gestalt principle of temporal proximity, and makes musical sense to a greater extent, than the one-to-one or many-to-one matchings: pulse No. 2 in rhythm A is now matched to pulse No. 1 in rhythm B instead of pulse No. 13, and pulse No. 13 in rhythm B is matched with pulse No. 14 in rhythm A, yielding a distance of $2 + 1 + 1 = 4$.

2. Experiments and Results

The one-to-one and many-to-many minimum-weight matching distance measures were compared with the edit distance using a dataset consisting of nine Middle-Eastern musical rhythms, for which human similarity judgments were available from previous listening tests performed at the Radcliffe Institute for Advanced Study at Harvard University in the spring of 2010. For details about the human subjects, stimulus materials used, and the experimental procedures applied, the reader is referred to the paper by Toussaint, Campbell and Brown [19].

2.1 Computing the Edit Distance

The edit distance between two sequences A and B of lengths n and m may be computed in $O(nm)$ time using dynamic programming. Edit distance online calculators are readily available on the Internet, and open source code in several programming languages is also obtainable. In the present study the Wagner-Fisher algorithm implemented by Malcolm Campbell [19] was used, in which the cost (weight) of all three operations (deletions, insertions, and reversals) was set equal to 1. Note that a reversal may be considered to be a deletion followed by an insertion, and therefore if the cost of a reversal is set to 2, it becomes superfluous.

2.2 Computing the Many-to-Many Matching

The minimum-weight many-to-many matching distance for rhythms with unequal numbers of pulses and onsets was computed by combining two existing algorithms. The well known Hungarian algorithm, also known as the Kuhn-Munkres algorithm [13], [14], computes a minimum-weight one-to-one (perfect) matching in a bipartite graph, and code is available in the Python library [26]. However, this algorithm does not solve the many-to-many matching problem. Fortunately, Eiter and Mannila [12] provide an algorithm for transforming one bipartite graph G into another bipartite graph G^* that contains twice as many nodes, such that the minimum-weight one-to-one perfect matching in G^* is an optimal solution to the many-to-many minimum-weight matching in G. This algorithm was programmed in the Python language to complement the code for the Kuhn-Munkres algorithm [26]. This approach to computing the many-to-many matching has the advantage that it applies to graphs in general, but the drawback that the computational complexity is $O(n^3)$ for sequences of length n. This is not a problem for short rhythms such as those tested here. Furthermore, versions of the Hungarian algorithm with improved expected running times are also available [15]. For long sequences faster $O(n \log n)$ algorithms are known for the special case of one-dimensional sequences [20].

2.3 The Rhythm Dataset

The dataset consisting of the nine Middle-Eastern rhythms is shown in box-notation in Figure 3. These rhythms vary in their number of pulses and onsets, and thus allow for the differences between the distance measures to be fleshed out. The filled boxes come in two varieties: a solid black disk indicates a low-pitched "dum" sound, whereas a white-filled disk indicates a high-pitched "tak" sound. For this reason they are referred to as *dum-tak* rhythms [17]. Although the two types of onsets ("dum" and "tak") differed in the acoustic samples heard by the participants in the listening tests, the distance measures did not differentiate between them, and treated them both simply as sounded pulses.

2.4 Results and Discussion

For every pair of rhythms in the dataset the distance measure between the pair was computed, yielding a distance matrix. The various distance measures were then compared with the matrix of human similarity judgments, by calculating the correlations between the corresponding matrices, using the Mantel test [27] available in the *zt* software package [28]. The Mantel test is a statistical tool expressly designed to test the correlation of structures that contain dependencies between their elements (such as distance matrices), which invalidate the assumption of independence. The resulting Mantel correlations and their p-values are listed in Table 1. The edit and many-to-many matching distances correlated significantly and almost equally with human judgments.

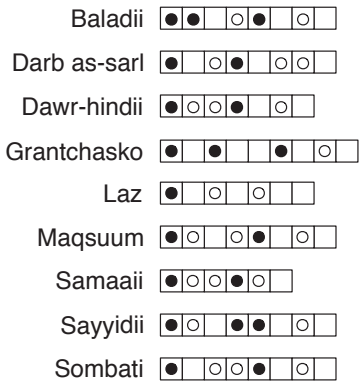


Fig. 3: The nine Middle Eastern rhythms in box notation.

By contrast, the one-to-one matching distance showed no significant correlation with human judgments. Although this distance fared well in previous studies that compared rhythms that had equal numbers of pulses and onsets [19], it is not surprising that for the *dum-tak* rhythms in Figure 3 the measure is inadequate, since in the comparison of rhythms such as the 3-onset *laz* with the remaining 5-onset rhythms the calculation ignores two onsets that may play a significant role in the human perception of these rhythms.

Table 1: Correlation with Human Judgments.

Distance Measures with Human Judgments		
Distance Measure	Corr	p-value
Edit Distance	0.677	0.001
Many-to-Many Matching	0.661	0.0006
One-to-One Matching	0.195	0.175

Correlations were also calculated between the edit distance and the many-to-many and one-to-one matching distance measures, and the results are shown in Table 2. In spite of the disparity between the correlation coefficients for the many-to-many and one-to-one distances when compared with human judgments in Table 1 and with the edit distance in Table 2, the many-to-many and one-to-one matching distances are significantly correlated with each other (correlation coefficient = 0.758 with $p = 0.0001$). This suggests that the seemingly small structural difference between the two matching distances is nevertheless an important factor for modeling the Gestalt principle of proximity in human rhythm perception.

Table 2: Correlation with Edit Distance.

Matching Distances with Edit Distance		
Matching Distance	Corr	p-value
Many-to-Many Matching	0.566	0.007
One-to-One Matching	0.205	0.168

3. Conclusion and Ongoing Research

The goal of this ongoing research project is to determine if the minimum-weight many-to-many matching distance as defined in the Introduction can yield better correlations with human judgements than the edit distance. The results obtained so far, with the dataset consisting of the *dum-tak* rhythms in Figure 3, provides no evidence contrary to the hypothesis that the edit distance is superior to all other distance measures. Similar studies are under way using several other datasets consisting of synthetic rhythms as well as rhythms from difference genres of music, to test this hypothesis further. In the present study no information was used that distinguishes between the two different sounding onsets ("dums" and "taks"), because previous studies that used a 3-symbol edit distance with different symbols for "dums," "taks," and silent pulses, revealed no differences in performance compared to the 2-symbol edit distance [19]. However, the many-to-many distance measure suggests an alternative method for incorporating information that distinguishes the "dums" from the "taks." The results reported here were obtained by computing the minimum-weight many-to-many matching on the complete bipartite graph in which all onsets of rhythm A are connected to all onsets of rhythm B. However, it may be that using incomplete bipartite graphs in which all "dums" in A are connected only to all the "dums" in B, and all "taks" in A are connected only to all the "taks" in B, may yield better results. There is also evidence that distance measures that take into account perceptually relevant musical factors can improve performance. The weights of the edges in the bipartite graph of the many-to-many matching distance, as well as the costs of the edit operations in the edit distance, may be chosen to reflect such perceptual information. These modifications of the distance measures will also be compared using all the datasets.

4. Acknowledgments

This research was supported by a research grant from the Provost Fund at New York University Abu Dhabi, and a Research Enhancement Fund grant from the NYUAD Institute, for the project titled: *Cross-Disciplinary and Multi-Cultural Perspectives on Musical Rhythm*. The research was done while the second author was a Global Academic Fellow at New York University Abu Dhabi, United Arab Emirates.

References

[1] J. F. Beltran, X. Liu, N. Mohanchandra, and G. T. Toussaint, "Measuring musical rhythm similarity: Statistical features versus transformation methods," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 29, no. 2, pp. 1–23, 2015.

[2] D. C. Correa, A. L. M. Levada, and L. da F. Costa, "A graph-based method for playlist generation," in *Proc. 9th International Symposium on Computer Music Modelling and Retrieval*, Queen Mary University of London, London,19-22 June 2012, pp. 466-473.

[3] R. M. Karp, and S. Y.-R. Li, "Two special cases of the assignment problem," *Discrete Mathematics*, vol. 13, pp. 129–142, 1975.

[4] E. Cao, M. Lotstein, and P. N. Johnson-Laird, "Similarity and families of musical rhythms," *Music Perception: An Interdisciplinary Journal*, vol. 31, no. 5, pp. 444–469, 2014.

[5] V. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Cybernetics and Control Theory*, vol. 10, pp. 707–710, 1966.

[6] T. Iwata, J. R. Lloyd, and Z. Ghahramani, "Unsupervised many-to-many object matching for relational data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 3, pp. 607–617, 2016.

[7] E. Olson, "M3RSM: Many-to-many multi-resolution scan matching," in *Proc. IEEE International Conference on Robotics and Automation*, Washington State Convention Center, Seattle, 2015, pp. 518-525.

[8] C. Guastavino, F. Gómez, G. Toussaint, F. Marandola, and E. Gómez, "Measuring similarity between flamenco rhythmic patterns," *Journal of New Music Research*, vol. 38, no. 2, pp. 175–184, 2009.

[9] J. Colannino, and G. Toussaint, "An algorithm for computing the restriction scaffold assignment problem in computational biology," *Information Processing Letters*, vol. 95, pp. 466–471, 2005.

[10] S. M. Oh, G. T. Toussaint, E. D. Demaine, and M. L. Demaine, "A dissimilarity measure for comparing origami crease patterns," in *Proc. International Conference on Pattern Recognition Applications and Methods*, 2015, pp. 386-393.

[11] D. Tymoczko, "The geometry of musical chords," *Science*, vol. 313, no. 5783, pp. 72–74, 2006.

[12] T. Eiter, and H. Mannila, "Distance measures for point sets and their computation," *Acta Informatica*, vol. 34, no. 2, pp. 109–133, 1997.

[13] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society of Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 109–133, 1957.

[14] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[15] R. Jonker, and T. Volgenant, "Improving the Hungarian assignment algorithm," *Operations Research Lett.*, vol. 5, no. 4, pp. 171-175, 1986.

[16] G. T. Toussaint, *The Geometry of Musical Rhythm: What makes a "Good" Rhythm Good?* Boca Raton: Chapman and Hall/CRC, 2013.

[17] H. H. Touma, *The Music of the Arabs*. Portland, Oregon: Amadeus Press, 1996.

[18] O. Post, and G. T. Toussaint, "The edit distance as a measure of perceived rhythmic similarity," *Empirical Musicology Review*, vol. 6, no. 3, pp. 164–179, 2011.

[19] G. T. Toussaint, M. Campbell, and N. Brown, "Computational models of symbolic rhythm similarity: Correlation with human judgments," *Analytical Approaches to World Music*, vol. 1, no. 2, pp. 380–430, 2011.

[20] M. Mohamad, D. Rappaport, and G. T. Toussaint, "Minimum many-to-many matchings for computing the distance between two sequences," *Graphs and Combinatorics*, vol. 31, pp. 1637–1648, 2015.

[21] G. T. Toussaint, "A comparison of rhythmic dissimilarity measures," *FORMA*, vol. 21, no. 2, pp. 129–149, 2006.

[22] R. L. Kline, and E. P. Glinert, "Approximate matching algorithms for music information retrieval using vocal input," in *Proc. ACM International Conference on Multimedia*, 2003, pp. 130-139.

[23] M. Panteli, B. Rocha, N. Bogaards, and A. Honingh, "Development of a rhythm similarity model for electronic dance music," in *Proc. International Conference of the Audio Engineering Society*, London, UK, January 27-29 2014, pp. 1-2.

[24] W. A. Sethares, and G. T. Toussaint, "Expressive timbre and timing in rhythmic performance: Analysis of Steve Reich's *Clapping Music*," *Journal of New Music Research*, vol. 44, no. 1, pp. 11–24, 2015.

[25] G. Toussaint, L. Matthews, M. Campbell, and N. Brown, "Measuring musical rhythm similarity: Transformation versus feature-based methods," *Journal of Interdisciplinary Music Studies*, vol. 6, no. 1, pp. 23–53, 2012.

[26] B. Clapper, "Munkres algorithm for the assignment problem: Version 1.0.6," (2008), [Online]. Available: http://software.clapper.org/munkres/

[27] N. Mantel, and R. S. Valand "A technique of nonparametric multivariate analysis," *Biometrics*, vol. 26, pp. 547–558, 1970.

[28] E. Bonnet, and Y. Van de Peer, "zt: A software tool for simple and partial Mantel tests," *Journal of Statistical Software*, vol. 7, no. 10, pp. 1–12, 2002.

The Intelligent Lighting System Realizing Individual Illuminance in the Office Not Influenced by Daylight Using Mathematical Programming

Katsuya ITO[2], Mitsunori MIKI[1], Daichi TERAI[2], Naoki KAWATA[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 authors have conducted a research of an Intelligent Lighting System, which provides individual illuminance environment to office workers. In the Intelligent Lighting System, the system maintains illuminance sensor which is set on the each office worker's desk surface to correspond the effect of daylight, the change in office layout, and the movement of office worker's desk, and the system uses generic algorithm to optimize lighting control. In the environment without the changes of lightings, the system doesn't need the conventional generic algorithm, and the illuminance sensor is no more needed. In this study, we propose the new lighting control method without using the illuminance sensor in the environment without the change of lighting environment.*

Keywords: Lighting, Intelligent Lighting System, illuminance sensor, Daylight

1. Introduction

In recent years, research has been actively conducted on the impact of an office environment on workers' productivity. It is expected that an improvement in an office environment will enhance workers' intellectual productivity, creativity, and comfortability[1], [2]. Above all, many studies have been made on office light environment. It has been revealed that providing optimal illuminance for work to each individual is effective in terms of improving an office environment[3].

On the other hand, lighting occupies a large portion of power consumption at an office, and methods have been proposed to reduce power consumption at an office by controlling lighting luminance[4], [5], [6]. These methods proposed reduce power consumption by controlling lighting in a particular area in accordance with information obtained from illuminance sensors to keep luminance from increasing more than required. These proposed methods, however, provide uniform illuminance to a particular area. It is not easy to provide illuminance catered to the preference, physical condition, and work content of each worker.

It is against such a backdrop that authors conduct research on a lighting system that provides an individualized illuminance environment to a worker in an office (hereinafter, Intelligent Lighting System)[7], [8]. As the Intelligent Lighting System enables workers to work in a light environment tailored to each of them and is expected to bring about such effects as improved comfort and reduced stress. In addition, as it provides necessary illuminance to a location requiring it, it can realize large energy saving by reducing the average illuminance of an entire room.

In introducing the Intelligent Lighting System to a real office, an illuminance sensor is placed on each worker's desktop in order to respond to various changes in light environment including the effect of daylight and workers' changing their seat. An individualized illuminance environment is realized by optimally controlling the luminance of lighting on the basis of illuminance information obtained from illuminance sensors. Illuminance sensors, however, are expensive, and wiring work must be performed in introducing the system. Consequently its introduction cost has become an issue in spreading the Intelligent Lighting System.

This study thus proposes a lighting control method that realizes individualized illuminance without using illuminance sensors and verifies its effectiveness. The cost for introducing the Intelligent Lighting System can be reduced as illuminance sensors are not used, which is expected to contribute to the diffusion of the system.

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

ISBN: 1-60132-438-3, CSREA Press ©

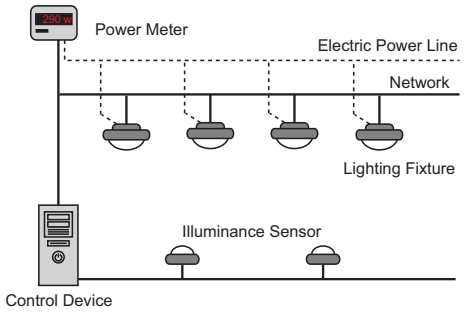


Fig. 1: Configuration of Intelligent Lighting System

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)[7], [9]. 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 varies the luminance of each lighting fixture in each search to an extent unnoticeable by workers to search an optimum lighting pattern.

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 Equation (1).

$$f_i = P + w \times \sum_{j=1}^{n} g_j \quad (1)$$

$$g_j = \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{cases}$$

i : lightng ID, j : illuminance sensor ID
P : power consumption [W], w : weight
n : number of illuminance sensors
Ic_j : current illuminance [lx], It_j : target illuminance [lx]

As indicate in the Equation (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 Introduction Issue of the Intelligent Lighting System

The Intelligent Lighting System responds flexibly to various changes in light environment by placing an illuminance sensor on each worker's desktop. Illuminance sensors, however, are expensive, and wiring work must be performed in introducing the system. Consequently its introduction cost has become an issue in spreading the Intelligent Lighting System.

On the other hand, there are many offices which do not have any window or hardly receive any daylight due to surrounding buildings even if they have a window. In such an environment, the effect of daylight need not be taken into account. Furthermore, as the service life of lighting fixtures have been prolonged in recent years owing to the spread of LED lamps, it is considered unnecessary to take the degradation of lighting fixtures into account in a short term. In this way, a change in light environment is not considered to occur frequently in an office uninfluenced by daylight. Since it is not required to respond to changes in light environment, we think that a more efficient algorithm which does not use an illuminance sensor can be taken into consideration.

This study thus examines a new lighting control method providing an individualized illuminance environment without using an illuminance sensor in an environment with infrequent changes in light environment.

3. The Intelligent Lighting System Realizing Individual Illuminance in the Office Not Influenced by Daylight

3.1 The Intelligent Lighting System Realizing Individualized Illuminance Without Using an Illuminance Sensor

A new control algorithm we propose uses a mathematical programming as a lighting control method realizing individualized illuminance without using an illuminance sensor in an office where the effect of daylight need not be considered. Mathematical programming is a mathematical method which formulates a real-world problem as a problem of mathematical programming and find the value of a variable that makes the value of the function maximum or minimum. The proposed method realizes the target illuminance for each worker and minimize power consumption by searching for the optimal luminance for each lamp satisfying the requirement of the Intelligent Lighting System by using mathematical programming and using the lighting patterns searched for lamps in the real environment. The proposed method can significantly reduce the cost of introducing the Intelligent Lighting System as it eliminates the need for illuminance sensors, as many of which as the number of workers were previously required.

3.2 Application of Mathematical Programming to the Requirements of the Intelligent Lighting s System

In applying mathematical programming to the requirements of the Intelligent Lighting System, Equation (1), which is the objective function, need to be calculable using design variables. In this case, a design variable is luminance of each lamp. If the objective function can be represented by using design variables, it becomes possible to calculate the optimal luminance for each lamp by mathematical programming only through a computer's internal processing.

In the objective function shown as Equation (1), power consumption P and illuminance information Ic are values obtained from a power meter and an illuminance sensor, respectively. That is, the objective function is represented by using measured values obtained from hardware instead of luminance of a lamp, which is a design variable. This section thus considers how to represent the objective function shown as Equation (1) by using design variables by describing power consumption and illuminance information in terms of luminance of a lamp.

First, let us consider how to represent power consumption in terms of luminance. It has been found by a preliminary experiment that luminance of a lamp in the vertically downward direction and its power consumption are in a proportional relation. It can thus be said that the luminance and power consumption of a lamp can be approximated by a linear equation. Therefore, power consumption can be represented as Equation (2).

$$P = \sum_{i=1}^{m} (\alpha L_i + \beta) \tag{2}$$

P : power consumption [W], i : lighting ID
m : number of lighting, α : coefficient [W/cd]
L_i : luminance of lighting$i$ [cd], β : coefficient [W]

Next, illuminance information is represented in terms of luminance of a lamp. There is a strong causal relationship between luminance and illuminance, which can be represented as Equation (3). The illuminance/luminance effect coefficient R in Equation (3) is a value that depends on light environment including the shape of a light source and the distance from a light source. Therefore, as long as there is no change in light environment, the illuminance/luminance effect coefficient R can be regarded as a constant.

$$Ic_j = \sum_{i=1}^{m} R_{i_j} L_i \tag{3}$$

Ic_j : illuminance of illuminance sensor$j$ [lx], i : lighting ID
j : illuminance sensor ID, m : number of lighting
R_{i_j} : illuminance/luminance effect coefficient [lx/cd]
L_i : luminance of lighting$i$ [cd]

This study assumes an office where a change in light environment does not occur frequently as an environment into which the proposed method is introduced. It is thus assumed that the illuminance/luminance effect coefficient R is measured prior to introducing the proposed method. It is described in Section 3.3 how to measure the illuminance/luminance effect coefficient R. If the illuminance/luminance effect coefficient R is already known, it is possible to represent information on the illuminance on the desktop of each worker in terms of luminance of each lamp by using Equation (3).

As described above, we managed to represent power consumption and illuminance information, which are measured values obtained from hardware, in terms of luminance of a lamp. Equation (1), the objective function, described in terms of luminance of lamps using Equations (2) and (3) is given as Equation (4). Now that the objective function is described in terms of luminance of lamps, lighting control by using mathematical programming is possible.

$$\begin{aligned} f &= \sum_{i=1}^{m} (\alpha L_i + \beta) + w \times \sum_{j=1}^{n} g_j \\ g_j &= \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ (\sum_{i=1}^{m} R_{i_j} L_i - It_j)^2 & (Ic_j - It_j) < 0 \end{cases} \end{aligned} \tag{4}$$

i : lighting ID, m : number of lighting
α : coefficient [W/cd], β : coefficient [W], w : weight
L_i : luminance of lighting$i$ [cd], j : illuminance sensor ID
n : number of illuminance sensor
It_j : target illuminance of illuminance sensor$j$ [lx]
Ic_j : current illuminance of illuminance sensor$j$ [lx]
R_{i_j} : illuminance/luminance effect coefficient [lx/cd]

3.3 Measuring Method of Illuminance / Luminance Effect Coefficient

It is assumed that a change in light environment does not occur frequently in an environment into which the proposed method is introduced.

This study thus uses an approach of actually measuring the illuminance/luminance effect coefficient R by placing an illuminance sensor on the desktop of each worker before the Intelligent Lighting System is introduced. Since the relationship between luminance and illuminance can be represented as in Equation (3), the illuminance/luminance effect coefficient for any illuminance to the illuminance sensor at any location can be represented as in Equation (5).

$$R_{ij} = \frac{I_j}{L_i} \tag{5}$$

i : lighting ID, j : illuminance sensor ID
R_{i_j} : illuminance/luminance effect coefficient [lx/cd]
I_j : illuminance of illuminance sensor$j$ [lx]

L_i : luminance of lighting$i$ [cd]

Therefore, by turning on and off each of lamps in the office in question, the illuminance/luminance effect coefficient R can be calculated by using Equation (5) on the basis of a change in illuminance of an illuminance sensor and a change in luminance of a lamp. By storing the illuminance/luminance effect coefficients R's measured in a database, it becomes possible to represent illuminance information in terms of luminance of lamps as described in Section 3.2.

4. Lighting Control Algorithm Using Mathematical Programming

4.1 Examination of Algorithm Used for Optimal Solution Search

This section examines the algorithm for mathematical programming used in searching the optimal lighting pattern for a lamp. Now, the objective function for the Intelligent Lighting System was managed to be represented by design variables by Equation (4) in Section 3.2. Since Equation (4) represents a non-linear function of design variables, the problem in question can be conceived as a non-linear programming problem.

While algorithms for searching the optimal solution to non-liner programming problems include the steepest descent method, Newton's method, and the quasi-Newton method, this study uses the steepest descent method, which is relatively easy to implement and has the least computational complexity per step. The steepest descent method, which is one of gradient methods, is an algorithm which search in the direction that maximizes the gradient of the objective function. A gradient vector obtained by differentiating the objective function with respect to each design variable is used as the search direction. If the objective function is unimodal, there is no possibility of ending up with a local optimal solution and global convergence is guaranteed. In this case, since the objective function of the Intelligent Lighting System shown as Equation (4) is a multivariate quadratic equation having as many design variables as the number of lamps and unimodal, global convergence can be said to be guaranteed.

4.2 Optimal Solution Search Using the Steepest Descent Method

This section describes the procedure for using the steepest descent method to search for the optimal luminance of each lamp that realizes the target illuminance for each worker. The luminance L of a lamp, a design variable in the objective function given as Equation (4), is represented by an m dimensional vector given in Equation (6) if the number of lamps is m. A gradient vector is a vector composed of partial derivatives with respect to each design variable of the objective function and given in Equation (7).

$$L = \{L_1, ..., L_i..., L_m\}^T \tag{6}$$

$$\nabla F(L) = \left(\frac{\partial F(L)}{\partial L_1}, \cdots, \frac{\partial F(L)}{\partial L_i}, \cdots, \frac{\partial F(L)}{\partial L_m} \right)^T \tag{7}$$

L_i : luminance of lighting$i$ [lx], m : number of lighting
F : objective function

The steepest descent method is an iterative solution and generates the sequence of points $\{L^k | k = 0, 1, \cdots\}$ from the initial point L^0. If the point L^k is searched for at the kth iteration, the gradient vector $\nabla F(L^k)$ is the increasing direction of the objective function F at this point. Therefore, the value of the objective function F can be decreased by taking steps in the opposite direction. Therefore, under the steepest descent method, the next search point is calculated in accordance with Equation (8) with step width α.

$$\begin{aligned} L^{k+1} &= L^k + \alpha s \\ s &= -\nabla F(L^k) \end{aligned} \tag{8}$$

k : iteration number, s : descending direction, L^k : luminance at the kth iteration [lx], F : objective function

Here the step width α can be calculated by solving the minimization problem given by Equation (9).

$$\min F(L^k + \alpha s) \tag{9}$$

k : iteration number, s : descending direction, L^k : luminance at the kth iteration [lx], F : objective function

As the minimization problem of Equation (9) is a minimization problem with one variable α, its solution can be found by linear search.

The procedure for searching the optimal luminance for each lamp using the steepest descent method is shown below. The golden section method is used in linear search in Step (4).

1) Choose the initial point L^0 by setting the initial value to the luminance of each lamp. Choose a sufficiently small positive number ε, and set the iterative number $k = 0$.
2) Calculate the gradient vector $\nabla F(L^k)$. If $||\nabla F(L^k)|| < \varepsilon$, terminate the search. L^k is the solution. Otherwise, proceed to Step (3).
3) Determine the descending direction $s = -\nabla F(L^k)$.
4) Solve the linear search problem given by Equation (9) to derive the step width α.
5) Update the luminance of each lamp in accordance with Equation (8) and return to Step (2).

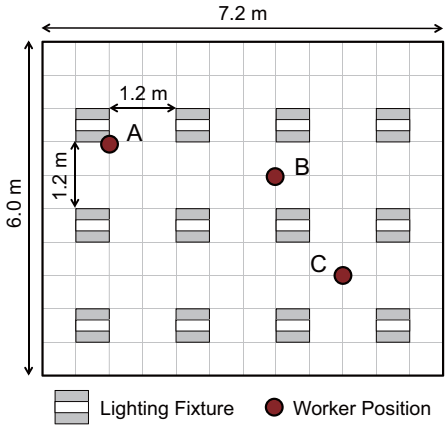


Fig. 2: Experiment environment

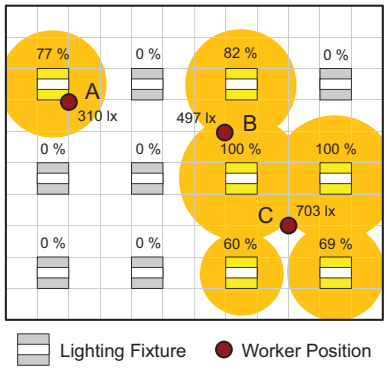


Fig. 3: Lighting Pattern After the Optimal Solution Search

4.3 The Flow of Lighting Control Under the Proposed Method

Following Section 4.2, we can now search the optimal luminance for a lamp that realizes the target illuminance for each worker using the steepest descent method. The flow of lighting control under the proposed method is shown below.

1) Turn on each lamp at the initial lighting luminance.
2) Set the target illuminance for each worker.
3) Search the optimal luminance of each lamp that realizes the target illuminance for each worker by the steepest descent method.
4) Make each lamp produce searched luminance.
5) If the target illuminance is changed, return to Step (2).

Controlling lighting as described above will realize the lamp lighting pattern that provides an individual illuminance environment required by each worker and minimizes power consumption.

5. Validation Experiments of the Proposed Method

5.1 Outline of Experiment and Experimental Environment

In order to verify the effectiveness of the proposed method, a verification experiment was conducted using the method. The experiment realizes the target illuminance set by each worker using the proposed method to verify illuminance and the lamp lighting pattern so realized. Fig.2 shows the experimental environment.

As shown in Fig.2, the experiment used a 7.2 m × 6.0 m room without window, in which 12 white dimmable LED lamps were placed. Although an illuminance sensor is not required for control by the proposed method, an illuminance sensor was placed on the desktop of each worker in order to confirm the extent to which the target illuminance for the worker is realized. As shown in Fig.2, 3 illuminance sensors were placed. The target illuminance for each worker was set to 300 lx, 500 lx, and 700 lx, respectively for seats A, B, and C in Fig.2. In the experimental environment shown in Fig.2, the vertical distance between the floor and lamps is 2.6 m, and the vertical distance between the floor and the desktops is 0.7 m, as recommended by JIS. The highest lighting luminance of LED lamps installed was 1144 cd, and the dimmable range of lamps was set from 5 % to 100 %. If the luminance of a lamp was below the dimmable range, the lamp was turned off.

5.2 Experimental Results and Discussion

Fig.3 shows lighting patterns of lamps which realized the target illuminance at a single trial of lighting control using the proposed method.

Fig.3 lets you confirm the lighting patterns of lamps searched by the proposed method and illuminance provided to workers. The Intelligent Lighting System is supposed to realize the target illuminance within the range of approximately ± 7 %. This result thus confirmed that the proposed method could provide illuminance individually required by each worker without using an illuminance sensor. Since lamps surrounding a worker were turned on and satisfied the target illuminance, the lighting pattern in Fig.3 can be said to be the optimal lighting pattern. In addition, as the proposed method realizes the optimal lighting pattern at a single trial of lighting control, given the result shown in Fig.3, it is considered to be a more energy-saving lighting control method.

The above showed the effectiveness of the proposed method for an environment with infrequent changes in light environment. Whereas the standard method has an advantage of being able to respond flexibly to various changes in light environment as it dynamically estimates the illuminance/luminance effect coefficient, as stated in Chapter 3, the proposed method has a problem of finding the illuminance/luminance effect coefficient in advance and updating it every time light environment changes. As an office

with infrequent changes in light environment, however, is assumed as an environment into which the proposed method is introduced, we do not think illuminance/luminance effect measurement is to be conducted frequently.

6. Conclusion

This study proposed a lighting control method using mathematical programming as a new algorithm for the Intelligent Lighting System for an office uninfluenced by daylight. The proposed method formulates the requirements of the Intelligent Lighting System as a problem of mathematical programming and searches the optimal lamp lighting pattern only through internal processing of a computer. While, in this case, it is required to find the illuminance/luminance effect coefficient in advance, an illuminance sensor need not be installed because lighting control does not have to use illuminance information obtained by an illuminance sensor. Since an illuminance sensor is not required, the method can significantly reduce the cost for introducing an Intelligent Lighting System, which has been its problem, and contribute to its diffusion.

As a result of the experiment verifying the effectiveness of the proposed method, it was found to realize illuminance individually required by each worker in a short time without using an illuminance sensor. It was also found to be a more energy-saving lighting control method. These results showed the effectiveness of the proposed method in an environment without the need to consider the influence of daylight.

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] M.J.Mendell and G.A.Heath, “Do indoor pollutants and thermal conditions in schools influence student performance? A critical review of the literature", Indoor Air, Vol.15, No.1, pp.27-52, 2005.
[3] Peter R.Boyce, Neil H.Eklund and S.Noel Simpson, “Individual Lighting Control: Task Performance, Mood, and Illuminance", Journal of the Illuminating Engineering Society, pp.131-142, 2000.
[4] Francis Rubinstein, Michael Siminovitch and Rudolph Verderber, “Fifty percent energy saving with automatic lighting controls", IEEE Industry Applications Society, Vol.29, pp.768-773, 1993.
[5] P.J.Littlefair, “Predicting lighting energy use under daylight linked lighting controls", Building Research and Information, Vol.26, No.4, pp.208-220, 1998.
[6] D.H.W.Li and J.C.Lam, “An investigation of daylighting performance and energy saving in a daylight corridor", Energy and Buildings, Vol.35, No.4, pp.365-373, 2003.
[7] M.Miki, T.Hiroyasu and K.Imazato, “Proposal for an intelligent lighting system, and verification of control method effectiveness", Proc. IEEE CIS, Vol.1, pp.520-525, 2004.
[8] M.Miki, K.Imazato and M.Yonezawa, “Intelligent lighting control using correlation coefficient between luminance anc illuminance", Proc.IASTED Intelligent Systems and Control, Vol.497, No.078, pp.31-36, 2005.
[9] 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.

Methodology for Disease Diagnosis based on Neural Network Using High Performance Computing

Sangman Kim[1], **Youngju Park**[1], **Jinhyeong Lee**[1] **and Jusung Park**[1]
[1]Department of Electronics and Electrical Engineering, Pusan National University, Busan, Rep. of Korea

Abstract - *In this paper, we suggest a methodology for disease diagnosis platform based on neural network. The methodology includes disease diagnosis algorithm, feature selection method and portable diagnosis platform. We also propose a method for high-performance computing to handle very large computations. For verification, we use clinical data which are obtained from 314 normal persons and 87 liver cancer patients. Each subject's clinical data consists of reacted values with 1,146 aptamers (features). Finally, we can diagnose liver cancer with 97.7% accuracy using 21 features and implement a portable disease diagnosis platform.*

Keywords: neural network, feature selection, diagnosis platform, high-performance computing, disease diagnosis

1 Introduction

Research for more accurate and easier way of disease diagnosis has been studied from past to the present [1]. Generally, disease diagnosis is carried out using limited number of markers (features) and it cannot guarantee the accuracy of diagnosis. In order to overcome this issue, disease diagnosis using neural network and microarray has been studied recently. To perform this method, we need solutions or resources to handle very large data and computations. Thanks to advanced IT, we can solve this issue. In this paper, we introduce a disease diagnosis algorithm, feature selection method based on neural network which has learning ability. The introduced feature selection method uses the same algorithm that is used for diagnosis so we can get effective features for diagnosis. But it is required a lot of time to compute and this might be a problem. However, the feature selection is performed one time after clinical data gathering step and the result of selection has great advantages. Firstly, if we can find effective features for diagnosis, we can diagnose a disease more accurate. It is the most important thing in diagnostic field. Secondly, if we can diagnose a disease with less features, it requires less resources and time for computations. These points have many advantages in hardware. Especially, these are meaningful to develop a portable disease diagnosis platform. In this paper, we also propose a method that to use high-performance computing for very large computations which are required to perform the feature selection algorithm. This method can dramatically reduce the computation time as available levels in real situations.

This paper is organized as follow. In section II, we will propose outline of methodology and introduce feature selection method based on neural network and cross validation. In section III, a high-performance computing method to perform algorithm which is introduced in pervious chapter will be proposed. In section IV, the result of proposed algorithm and implementation of disease diagnosis platform will be explained. Finally, conclusion will be made in section V.

2 Diagnosis and Feature Selection

The methodology is proposed in this paper consists of 3-steps: clinical data gathering, feature selection and generation of diagnosis algorithm and implementation of diagnosis platform as Fig. 1.

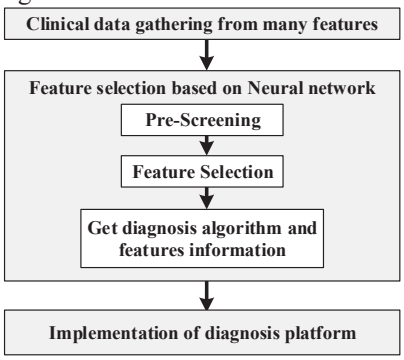


Figure 1. Outline of proposed methodology

In the 1st step, we gather clinical data using microarray which consists of many features. In the 2nd step, we select effective features for disease diagnosis from many features. In this step, we use the same neural network algorithm for feature selection and diagnosis so we can get more effective features for the diagnosis algorithm. In the 3rd step, we implement a disease diagnosis platform using the result of the 2nd step.

2.1 Data gathering

We use microarray to select effective features for detecting a specific disease. Generally, microarray has many features and it is the best structure for finding effective

features when we don't know about detecting materials. In this step, we gather clinical data for a specific disease using microarray.

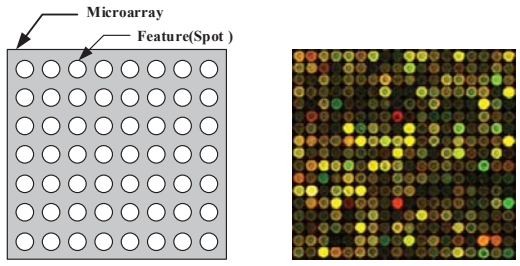


Figure 2. Structure of microarray

2.2 Neural Network

We design a basic neural network which has a 3-layers structure as Fig. 2. The network we used in this paper has many input nodes, 10 hidden nodes and 2 output nodes. We use reacting value of features as inputs and get normal and abnormal possibilities of used data as result. To adjust weight of the neural network, we use error-back propagation algorithm.

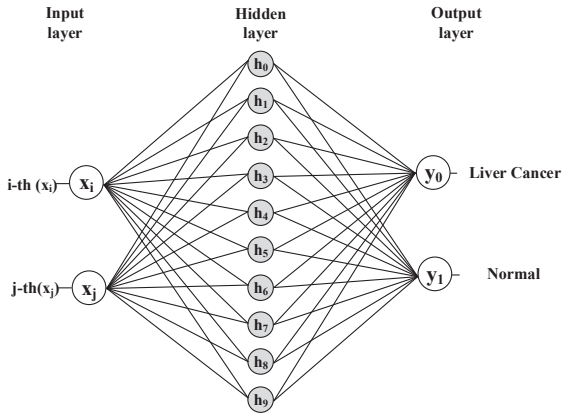


Figure 3. The structure of the three-layers neural network

2.3 Feature Selection

Microarray consists of many unspecified features so there is a possibility that some features contain un-useful information. Generally, a result of neural network algorithm is affected by combination of inputs. Therefore, it is important to select features which have useful information for diagnosis. For this, we use the algorithm which is proposed in [1]. Basic concept of this algorithm is to get diagnosis accuracy with adding up features one-by-one and select the feature if the accuracy is improved as Fig. 4. This feature selection algorithm consists of pre-screening, selection and determination steps.

Pre-screening step is performed for more effective selection. We use one-way ANOVA in this step. As the result of one-way ANOVA, we can obtain p-values of each feature and re-arrange features ascending order based on p-value.

In the selection step, we obtain a diagnosis accuracy to add up features one-by-one using the result of previous step and select the feature if the accuracy is improved or same compared with previous accuracy which is obtained before the addition. The accuracy which is used for selection is obtained using neural network and cross validation. In this step, the results of selections depend on grouping method for cross validation. In order to overcome this issue, we perform this selection step many times with different grouping method and gather information of the number of selected times of each feature. After gathering the information, we re-arrange the features in the descending order based on the number of selected times.

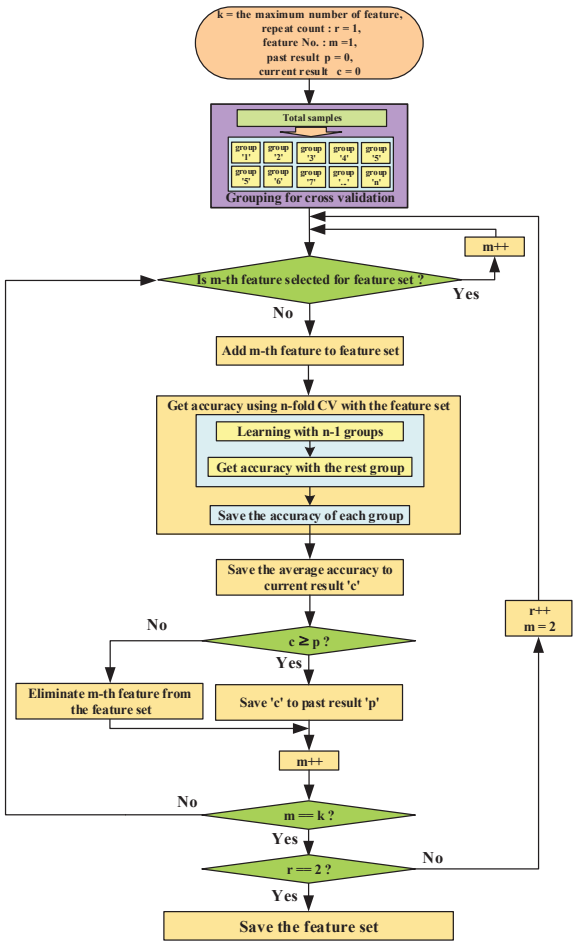


Figure 4. Feature selection method based on neural network

In the determination step, we get diagnosis accuracy using re-arranged result and save the feature information which have the best accuracy. Many cross validations are performed for reliable result in this step and it requires a lot of computations. Lastly, we save the neural network information which is used for determination. This will be used for implementation of diagnosis platform.

3 Using high-performance computing

The feature selection method we used in this paper uses cross validation. When using it, the reliability of the result is increased with more cross validation executions. Therefore, we need many cross validation executions and it also requires many computations. If we handle these computations with PC (Personal Computer), it will take a very long time and it will be difficult to use in real situation. To solve this problem, we use MPI (Message Passing Interface) which is one of the parallel programming. The basic concept of this method is to distribute the computations among the connected processors (threads) as Fig. 5.

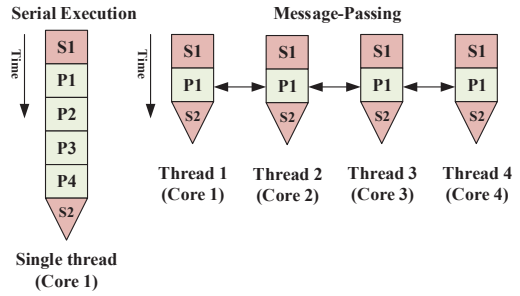


Figure 5. Basic concept of MPI

It is easy to use MPI for determination step of the feature selection method. As we mentioned it in section II, we use many cross validations to obtain diagnosis accuracy. The c-code for obtaining accuracy in determination step is shown in Fig. 6.

```
//Get diagnosis accuracy using N features

for(i=0; i<cross-validations; i++)
{
 Seed = Get seed value();
 Generate random number(seed);

 Make M groups using random number();

 for(j=0;j<M;j++)
 {
  Learning using M-1 Groups();

  Diagnose using j-th Groups();

  Sum += Save j-th accuracy;
 }
}
Diagnosis_accuracy = Sum / (M * cross-validations);
```

(For M-fold cross validation)

Figure 6. C-code for determination using single tread

In this step, the main issue is to generate many different cases for cross validation and we use random function. To parallelize this code, we add some API (Application Programming Interface) for MPI and modify the code in Fig. 5 with minor changes. The modified c-code using MPI is shown in Fig. 7. In case of using MPI, we use 'seed + nProcs (ID of thread)' as a seed value for random function. It prevents every thread from generating the same random number. As shown in Fig. 7, we can easily parallelize the determination step.

```
//Get diagnosis accuracy using N features with MPI

MPI_Init();                                   } MPI setting
MPI_Comm_size(..., nProcs);
MPI_Comm_rank(..., nRank);

nQuotient = cross-validations / nProcs;       } Divide iteration
nRemainder = cross-validations % nProcs;        cases for
nMyStart = nRank * nQuotient + 1;               each processor
nMyEnd = nMyStart + nQuotient - 1;

for(i=nMyStart; i<nMyEnd; i++)
{
 Seed = Get seed value();
 Generate random number(seed+nProcs);
  .                                           } For M-fold cross validation
  .                                             in Fig. 5
  .
}

if(nRank == Root)
{
   MPI_Recv(nProcs_Result,...);               } Root processor :
   Final_accuracy += nProcs_Result;             MPI result gathering
}
else
{
  MPI_Send(nProcs_Result, ...);               } Other processors :
}                                               MPI result sending
```

Figure 7. C-code for determination using MPI

4 Experiment and implementation

For verification of proposed methodology, we used high-performance computing method for feature selection. For this, we used a super computer at KIST (Korea Institute of Science and Technology). The super computer was equipped with Intel Xeon 2.6Ghz CPU-512 nodes (threads) and supported MPI. After we had obtained feature information and diagnosis algorithm, we implemented portable disease diagnosis platform using the information.

4.1 Experiment

We used microarray which consists of 1,146 aptamers (features) to obtain clinical data. The aptamer is oligo nucleic acid or peptide molecules that bind to a specific target molecule so we can measure a relative ratio of specific protein of serum. We obtained clinical data using microarray from 314 normal persons and 87 liver cancer patients. For feature selection method, we used 10-fold cross validation. In determination step, we carried out 100,000 cross validations for reliable result and the result is shown as Table I.

Table 1. Diagnosis accuracy depending on number of features

Number of aptamers	1	2	3	4	5	6	7	8
Accuracy	69.7%	78.3%	82.3%	88.0%	88.6%	89.6%	92.9%	93.3%
Number of aptamers	9	10	11	12	13	14	15	16
Accuracy	93.6%	93.5%	95.6%	96.2%	96.2%	95.8%	95.9%	95.7%
Number of aptamers	17	18	19	20	21	22	23	24
Accuracy	96.2%	96.7%	96.7%	96.9%	97.7%	97.7%	97.7%	97.5%
Number of aptamers	25	26	27	28	29	30	31	32
Accuracy	97.6%	97.5%	97.4%	97.3%	97.7%	97.7%	97.6%	97.5%

4.2 Implementation

We designed disease diagnosis platform based on 32 bits RISC (Reduced Instruction Set Computer) processor. The implemented platform consists of SRAM, Flash, UART, TFT-LCD and ADC sensor board as shown in Fig. 8.

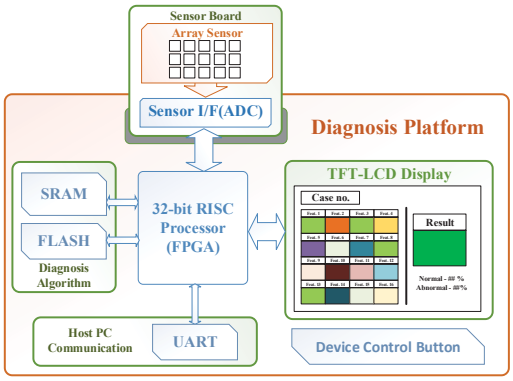


Figure 8. Architecture of the designed platform

The ADC sensor board has 16 2-channel ADC ICs so the platform can handle 32 bio signal inputs. The implemented platform and dimension information are shown in Fig. 9.

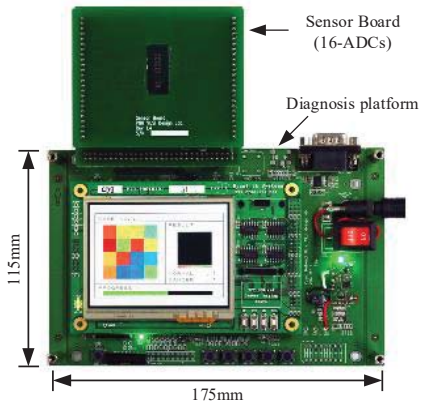


Figure 9. Picture of the designed platform

In case of using 32 inputs, the amount of required floating point computations for diagnosis are shown in Table II. It can be handled with low-power RISC type processor. Actually, we could get the diagnosis result using developed platform with 32 inputs in 1 minute.

Table 2. The amount of required computations using 32 features

	Input-Hidden	Hidden node	Hidden-Output	Output node	Total
MUL/DIV	640	20	20	4	684
ADD	0	640	20	2	662

5 Conclusion

In this paper, we proposed a methodology for implementation of diagnosis platform. The methodology includes clinical data gathering, disease diagnosis algorithm based on neural network, feature selection. We also proposed and verified the method to use high-performance computing for feature selection. For verification, we used clinical data which are obtained from 401 subjects (persons). As the result, we can diagnose liver cancer with 97.7% using 21 features. Finally, we calculated the amount of required computations and hardware resources in case of using 32 features for diagnosis and designed the portable disease diagnosis platform for verification. The proposed methodology is applicable to select features for other diseases and to develop a diagnosis platform.

6 Acknowledgment

This work was supported by PLSI supercomputing resources of Korea Institute of Science and Technology Information and BK21PLUS, Creative Human Resource Development Program for IT Convergence.

7 References

[1] Sangman Kim, et al. "Effective liver cancer diagnosis method based on machine learning algorithm", Biomedical Engineering and Informatics (BMEI), 2014 7th International Conference on. IEEE, pp.632-636, 2014

[2] Kun Yang, Haipeng Wang, Guojun Dai, Sanging Hu, Yanbin Zhang, Jing Xu, "Determining the repeat number of cross-validation," Biomedical Engneering and Informatics 4th international conference, Vol. 3, pp. 1957-1960, 2011

[3] Eom, J.-H, Kim, S.-C., andZhang, B.-T., "AptaCDSS : A classifier ensemble-based ovascular disease level prediction", Expert System with Application, Vol. 38, No. 16, pp.271-278, 2006

[4] Cho, S.-B., and Won, H.-H., "Machine Learning DNA Microarray Analysis for Cancer Classification", conferences in Research and Practice in Information Technology, Vol. 19, pp. 523-527, 2003

[5] Leung, Y.F., and Cavalieri D., "Fundamentals of cDNA microarray data analysis", Trends in Genetics, , 19(11), pp.649-659, 2003

[6] Churchill, G. A., "Fundamentals of experimental design for cDNA microarrays," Nature Genetics, Suppl. 32, pp.490-495, 2002

[7] Baxt, W. G., "Use of an artificial neural network for data analysis in clinical decision making : The diagnosis of acute coronary occlusion," Neural Computing, Vol. 2, No. 4, pp.480-489, 1990

[8] Geoffery J. McLachlan, Kim-Anh Do, Christophe Ambroise, "Analyzing microarray gene expression data", Wiley, 2004

[9] Carol Redmond, Theodore Colton, "Biostatistics in Clinical Trials", John Wiley & Sons, 2001

[10] E. Rich and K. Knight, "Artificial intelligence", McGraw-Hill, New York, 1991

Implementation to Provide Individual Illuminance and Color Temperature in an Intelligent Lighting System by Estimating the Color Temperature

Ryohei JONAN[2], Mitsunori MIKI[1], Shinya DAINAKA[2], Sota NAKAHARA[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 are conducting research and development of the Intelligent Lighting System providing illuminance and color temperature individually to workers at an office. Under the Intelligent Lighting System, a chroma meter is placed on the desktop of each worker in order to measure illuminance and color temperature. It is not easy to introduce the system to a real office, however, as a chroma meter is expensive. This study thus proposes a new method for providing illuminance and color temperature individually to each worker by estimating instead of measuring color temperature. It eliminates the need to use a chroma meter and contributes to enhancing the ease in introducing the Intelligent Lighting System. A verification experiment was conducted to show that illuminance and color temperature can be individually realized at the same precision as when a chroma meter is used.*

Keywords: Office, Lighting, Intelligent Lighting System, Color temperature, Chrome meter, Illuminance sensor

1. Introduction

In recent years, there has been increased attention on improvements in the intellectual productivity, creativity, and comfort of office workers within the office environment[1]. In the research of Boyce et al., it was reported that providing optimal illuminance separately for each worker is effective from the standpoint of improving the lighting environment[2]. Furthermore, when PHILIPS introduced lighting with a color temperature of 17000 K in offices and factories and conducted experiments in relation to this, powers of memory and concentration improved and work efficiency improved[3]. In regard to illuminance, it has been reported that preferences differed according to the individual[4]. Furthermore, in the same way, it is understood that preferences for color temperature also differ according to the individual[5].

It is against such a backdrop that we conduct research on the Intelligent Lighting System that provides illuminance and color temperature individually to each worker at an office[6]. The Intelligent Lighting System realizes the illuminance (target illuminance) and color temperature (target color temperature) required by a worker in a location in which a chroma meter is placed at minimal power consumption. This is expected to bring about such effects as improving workers' comfort and reducing their stress. The system also reduces power consumption by lighting to save more energy in lighting.

The effectiveness of the Intelligent Lighting System has already been recognized, and demonstration tests have been conducted in multiple offices in Tokyo[7]. As chroma meters are expensive, Intelligent Lighting Systems introduced into real offices only provided target illuminance by using inexpensive illuminance sensors. Workers manually set color temperature for each lighting fixture through a user interface, which requires their trouble.

We thus devised a method to provide illuminance and color temperature individually to workers without using a chroma meter (simplified method)[8]. The simplified method realizes color temperature required by a worker in a simplified manner by lighting lamps around an illuminance sensor at the target color temperature set for the illuminance sensor. In an environment in which illuminance sensors are placed in proximity with each other, however, color temperature required by a worker failed to be realized in some cases.

This study thus proposes a method that provides illuminance and color temperature individually to a worker without using a chroma meter even in an environment in which illuminance sensors are placed in proximity with each other. The proposed method estimates the color temperature of the desktop of each worker from the brightness (luminance) of the lamps and controls the lamps in accordance with the color temperature estimated. This eliminates the need for a chroma meter and improves the extent to which color temperature is realized in comparison with the simplified method. Furthermore, as the operation of setting the color temperature separately for each type of lighting is unnecessary, the burden on the office worker is reduced.

In this study, we construct a system integrating the proposed method, and demonstrate its effectiveness through verification experiments in an environment simulating an actual office. We conduct a verification experiment to show that target illuminance and color temperature can be realized at the same precision as they are realized by a method using a chroma meter. It is also to be shown that the precision for

ISBN: 1-60132-438-3, CSREA Press ©

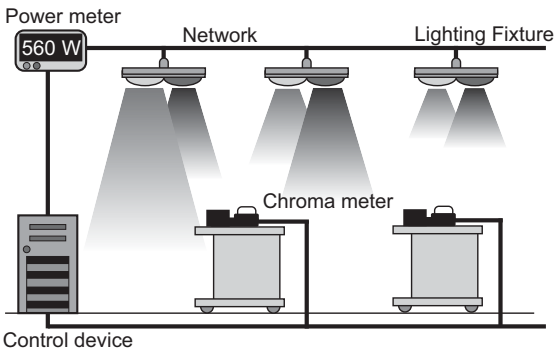


Fig. 1: Configuration of the Intelligent Lighting System

the target illuminance realized will be improved relative to the simplified method.

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, chroma meters, 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 and color temperature data from chroma meters 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 and color temperature 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)[9]. 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 varies the luminance of each lighting fixture in each search to an extent unnoticeable by workers to search an optimum lighting pattern.

Next, we will describe the objective function used in this algorithm. The objective of the Intelligent Lighting System is to achieve the illuminance and color temperatures required by each office worker and minimize power consumption. The objective functions are formularized as shown in Eq. 1.

$$f_i = P + \omega_l \times \sum_{j=1}^{n} g_{ij} + \omega_c \times \sum_{j=1}^{n} h_{ij} \quad (1)$$

$$g_{ij} = \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ R_{ij} \times (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{cases}$$

$$h_{ij} = \begin{cases} 0 & (Cc_j - Ct_j) \geq 0 \\ R_{ij} \times (Cc_j - Ct_j)^2 & (Cc_j - Ct_j) < 0 \end{cases}$$

$$R_{ij} = \begin{cases} r_{ij} & r_{ij} \geq T \\ 0 & r_{ij} < T \end{cases}$$

i:number of lightings, j:number of sensors
ω_l:wight [W/lx^2], ω_c:weight [W/K^2]
P:power consumption [W], Ic:current illuminance [lx]
It:target illuminance [lx], Cc:current color temperature [K]
Ct:target color temperature [K], L:luminace [cd]
T:threshold
r_{ij}:influence coefficient for illuminance sensor j for lighting i

As indicate in the Eq. 1, the ovjective function consists of power consumption P and illuminance constraint g_i and color temperature constraint h_i. The illuminance constraint g_i brings current illuminance to target illuminance or greater, as indicated by formula. And color temperature constraint h_i brings current temperature to target color temperature or greater, as indicated by formula. The influence coefficient r_{ij} of a Lighting i on each illuminance sensor j is a constant. By conducting a measurement experiment in introducing the system, the influence coefficient r of each Lighting on each chroma meter is measured.

2.3 Simplified method realizing the target color temperature without using a chroma meter

It is not easy to introduce the Intelligent Lighting System to a real office because chroma meters are expensive. Therefore, Intelligent Lighting Systems introduced into real offices have controlled lighting in accordance with ANA/RC so as to realize the target illuminance only by using only illuminance sensors, which are inexpensive relative to chroma meters. The target color temperature is realized by having workers set color temperature for each lighting fixture through a user interface, which requires their trouble.

We thus devised a method to realize target illuminance and color temperature without using a chroma meter (simplified method)[8]. To realize the target illuminance, the simplified method controls lighting in accordance with ANA/RC, just as Intelligent Lighting Systems introduced in real offices do. It realizes the target color temperature by turning multiple lamps around an illuminance sensor at the target color temperature set for the illuminance sensor. A lamp in a close distance to multiple illuminance sensors is turned on at the average of color temperature values set for those proximate illuminance sensors. Using the simplified method eliminated

the need for control through a user interface and led to reducing a worker's burden.

2.4 Problem with color temperature realization under the simplified method

If illuminance sensors are placed in proximity with each other, the simplified method failed to realize the target color temperature in some cases[8]. As workers' desks are often adjacent to each other in an ordinary office, it is concerned that the target color temperature may not be realized if the simplified method is used. This study thus propose a method that realizes target illuminance and color temperature set by a worker without using a chroma meter even in an environment in which illuminance sensors are in proximity with each other.

3. Method for realizing the target color temperature by estimating the color temperature of the desktop

3.1 Outline of the proposed method

We propose a method for realizing the target illuminance and color temperature for each worker without using a chroma meter. The proposed method estimates the color temperature of the desktop of each worker from information on lamps' luminance. In accordance with the estimated color temperature and illuminance measured by an illuminance sensor, the target illuminance and color temperature for each worker are realized by using ANA/RC.

3.2 Examination of the method for estimating the color temperature of a desktop

Lighting fixtures used in the Intelligent Lighting System are composed of two types of light sources: natural and warm white light sources. The color temperature of the desktop directly under a single lamp can be estimated from the ratio of illuminance from natural white light sources (natural white illuminance) to illuminance from warm white light sources (warm white illuminance)[8].

A verification experiment was conducted to investigate whether color temperature can be estimated from the ratio of natural white illuminance to warm white illuminance even in an environment in which multiple lamps exist. Fig. 2 shows the experimental environment. Nine LED Lightings made by SHARP were used which can range from 2,700 K to 5,400 K. Three CL-200A chroma meters made by Konica Minolta were used as chroma meters.

The experimental procedure is given below.

1) Randomly determine the luminance of natural and warm white light sources of each lamp.
2) Turn on the natural white light source only at the luminance determined in Step 1) and measure illuminance.

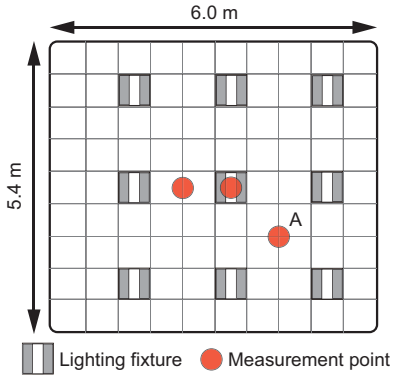


Fig. 2: Experiment Environment (ground plan)

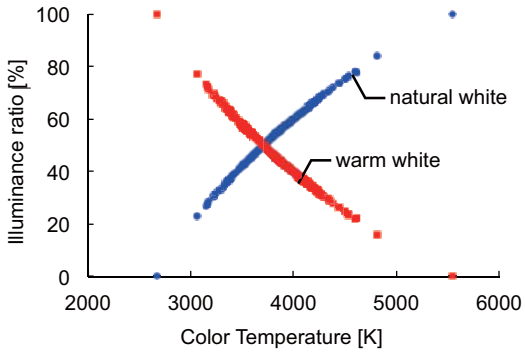


Fig. 3: The relashionship of neural and Incandescent lamp's illuminance ratio and color temoerature

3) Turn on the warm white light source only at the luminance determined in Step 1) and measure illuminance.
4) Turn on both natural and warm white light sources and measure color temperature.

Repeat a trial consisting of Steps 1) through 4). This is intended to verify the relationship between the ratio of natural white illuminance to warm white illuminance and color temperature at each measurement location.

Fig. 3 gives the result of the experiment at the measurement location A. Fig. 3 shows that there is a regular relationship between the ratio of natural white illuminance to warm white illuminance and color temperature. The same result was obtained at other measurement locations. These results confirmed that there is a regular relationship between the ratio of natural white illuminance to warm white illuminance and color temperature in an environment in which multiple lamps exist.

3.3 Method for estimating color temperature on the basis of the luminance of a lamp

Section 3.2 shows that color temperature at an illuminance sensor location can be estimated if natural white illuminance and warm white illuminance at the illuminance sensor location are known. There is a linear relationship between the

illuminance of an illuminance sensor and the luminance of a lamp. Eq. 2 gives the illuminance of an illuminance sensor in an environment in which multiple lamps exist.

$$I_j = \sum_{i=1}^{N} R_{ij} L_i \quad (2)$$

i : lamp ID, j : sensor ID, N : number of lightings
I_j : illuminance from sensor j [lx]
L_i : luminance of lamp i [cd]
R : influence coefficient for illuminance sensor j for lamp i [lx/cd]

Therefore, by using influence coefficient R, the natural white illuminance and the warm white illuminance at each illuminance sensor location can be calculated from the luminance of lamps.

Under the proposed method, the experiment in Section 3.2 is to be conducted in advance. A regression curve was derived from data obtained by using the least square method, and the polynomial in the function of the regression curve was defined as the model equation for the color temperature estimation at that location. Using the model equation, color temperature is estimated from the ratio of natural white illuminance to warm white illuminance calculated in accordance with Eq. 2. Color temperature is thus estimated from the luminance of lamps instead of being measured by using a chroma meter, and lighting is controlled using ANA/RC.

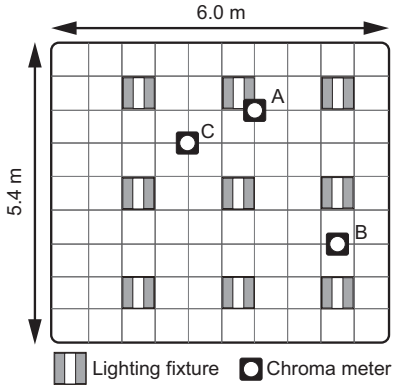


Fig. 4: Experiment Environment (ground plan)

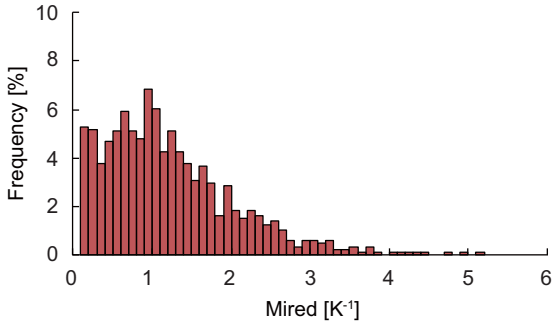


Fig. 5: Histogram of the color temperature error (sensor A)

4. Verification experiments

4.1 Outline of the verification experiments

Following two verification experiments were conducted to demonstrate the effectiveness of the proposed method.

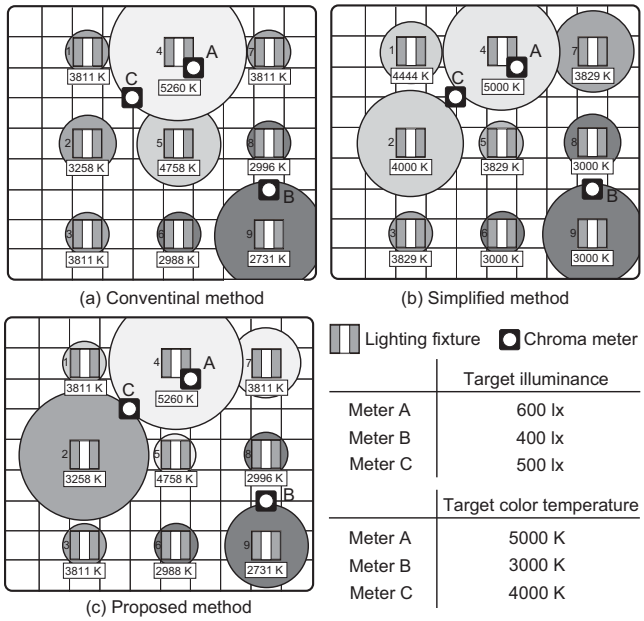
- Verification experiment for color temperature estimating precision
- Verification experiment for the extent of color temperature realization by the Intelligent Lighting System incorporating the proposed method

Fig. 4 shows the experimental environment. The laboratory, lamps, and chroma meter used in the experiment are the same as those for the experiment in Section 3.2. The influence coefficient of each lamp on each influence sensor was measured in advance. Since a difference in color temperatures is not in a proportional relation with that in colors recognized by human eyes, an inverse color temperature, which is the inverse of a color temperature, was used for verification. An inverse color temperature is the inverse of a color temperature multiplied by 10^6 and represented in the unit of mired (K^{-1}).

4.2 Verification of color temperature estimating precision

The luminance of each lamp was changed 500 times at random, and the color temperature at the location where each chroma meter was placed was estimated using the proposed method each time luminance was changed. The difference between an estimated color temperature and a corresponding measured color temperature was calculated to be used as an error to evaluate the proposed method. The model equation for estimating a color temperature by the proposed method at each measurement location was derived in advance by conducting the experiment described in Section 3.2.

Fig. 5 shows the histogram of color temperature error at the chroma meter A. In Fig. 5, the average inverse color temperature error was 1.11 K^{-1}, and the maximum inverse color temperature error was 5.04 K^{-1}. At the chroma meter B, the average inverse color error was 0.97 K^{-1}, and the maximum inverse color temperature error was 5.41 K^{-1}, exhibiting the largest color temperature errors among all chroma meters. At the chroma meter C, the average inverse color error was 1.03 K^{-1}, and the maximum inverse color temperature error was 4.52 K^{-1}, exhibiting the smallest color temperature errors among all chroma meters. Regarding color temperature, it is known that a difference in inverse color temperature of about 5.5 K^{-1} is not recognized by human eyes. Since the maximum inverse color temperature difference is 5.5 K^{-1} or less at any chroma meter, those errors are considered to be sufficiently small.



	Target illuminance
Meter A	600 lx
Meter B	400 lx
Meter C	500 lx

	Target color temperature
Meter A	5000 K
Meter B	3000 K
Meter C	4000 K

Fig. 6: The lighting distribution when the sensors are placed away

4.3 Verification of the extent of illuminance and color temperature realization

The extent of illuminance and color temperature realization was verified to show the effectiveness of the Intelligent Lighting System incorporating the proposed method. In the initial condition, all lamps were turned on at the maximum lighting luminance, and the Intelligent Lighting System was started under that condition. The target illuminance and color temperature at chroma meters A, B, and C were set to 600 lx and 5000 K, 400 lx and 3000 K, and 500 lx and 4000 K, respectively.

The Intelligent Lighting System was operated for 1,000 steps (about 2,000 seconds) using the conventional method using chroma meters, the simplified method, and the proposed method. Under the simplified and proposed methods, color temperatures obtained from chroma meters were used only to confirm the extent of realization and not to control lighting.

Fig. 6 shows lamp lighting patterns 2,000 seconds after the start of the experiment. It is found that lamps near a chroma meter were brightly turned on, realizing an energy saving lighting pattern. Fig. 6-(b) also shows that lamps in a close distance to a chroma meter were turned on at the target color temperature set for that chroma meter. Fig. 6-(a) and Fig. 6-(c) show that color temperatures of lamps were finely controlled in order to realize target color temperatures.

Histories of illuminance under the conventional, simplified, and proposed methods are given, respectively, in Fig. 7, Fig. 8, and Fig. 9. In each history of illuminance, let us focus

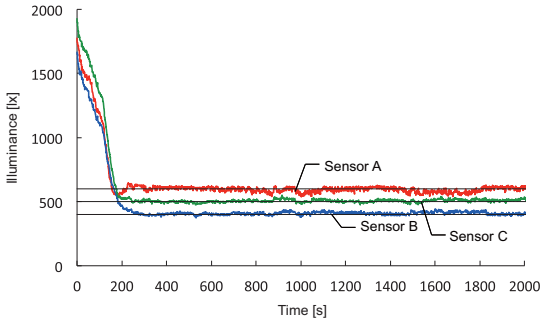


Fig. 7: History of Illuminance using conventional method

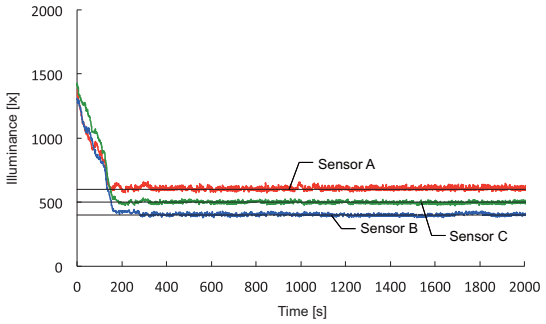


Fig. 8: History of Illuminance using simplified method

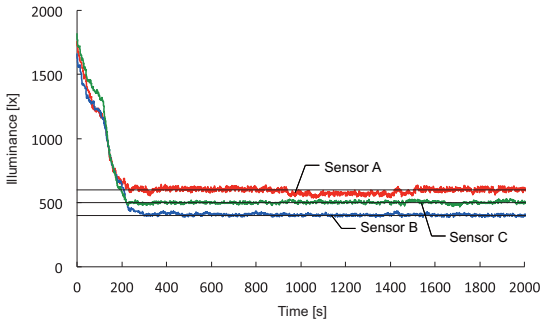


Fig. 9: History of Illuminance using proposed method

on the period from 1,600 seconds to 2,000 seconds after the start of the experiment, when lighting control stabilized. In Fig. 7, the average error from the target illuminance was 16 lx, 11 lx, and 13 lx, respectively, at the chroma meter A, B, and C. Next, in Fig. 8, the average error from the target illuminance was, respectively, 11 lx, 9 lx, and 10 lx. Next, in Fig. 9, the average error from the target illuminance was, respectively, 8 lx, 5 lx, and 7 lx. It is known that, when a worker is working at VDT, a difference in illuminance of about 12% is not recognized by human eyes. An error is considered to be sufficiently small at chroma meter A, B, and C if it is, respectively 72 lx or less, 48 lx or less, and 60 lx or less. Consequently, as a result, values of luminance measured by each chroma meter under each method satisfied this condition.

Next, histories of color temperature under the conven-

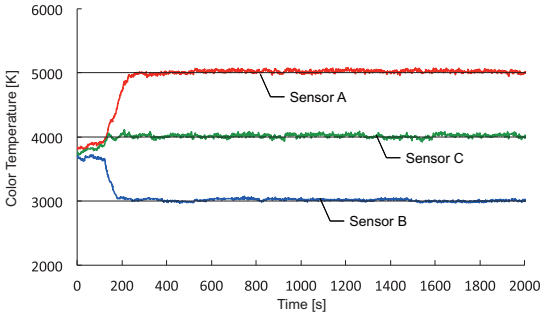


Fig. 10: History of color temperature using conventional method

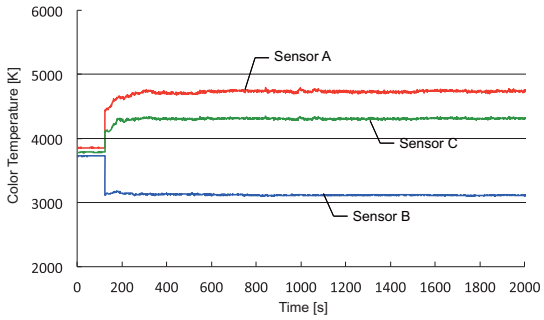


Fig. 11: History of color temperature using simplified method

tional, simplified, and proposed methods are given, respectively, in Fig. 10, Fig. 11, and Fig. 12. In each history of color temperature, let us focus on the period from 1,600 seconds to 2,000 seconds after the start of the experiment, when lighting control stabilized. In Fig. 10, the average error from the target color temperature was 22 K, 15 K, and 23 K, respectively, at the chroma meter A, B, and C. Next, in Fig. 11, the average error from the target color temperature was, respectively, 263 K, 113 K, and 312 K. On the other hand, in Fig. 12, the average error from the target color temperature was, respectively, 41 K, 20 K, and 15 K. It is known that a difference in color temperature of about 5.5 K^{-1} is not recognized by human eyes. An error is considered to be sufficiently small at chroma meter A, B, and C if it is, respectively 138 K or less, 50 K or less, and 88 K or less. As a result, values of luminance measured by every illuminance meter under the conventional and proposed methods satisfied this condition. On the other hand, under the simplified method, no illuminance sensor was able to satisfy this condition. Based on the above result, it can be said that, even in an environment in which chroma meters are in proximity with each other, the approach for realizing color temperature using the proposed method is effective compared with the simplified method.

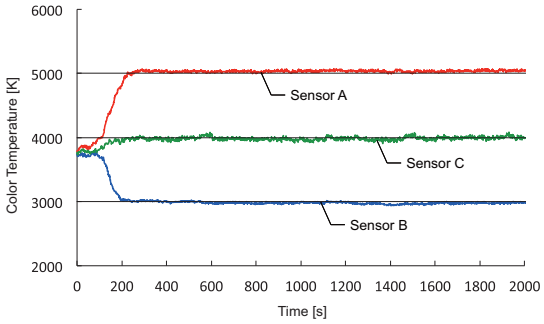


Fig. 12: History of color temperature using proposed method

5. Conclusion

In this study, we the proposed method to realize target illuminance and target color temperature required by each worker without using a chroma meter. We conducted verification experiment, and we showed that there is a regular relationship between the ratio of natural white illuminance to warm white illuminance and color temperature. Therefore, in the proposed method, color temperature of a desktop is estimated from the ratio of natural white illuminance to warm white illuminance. A verification experiment was conducted to show that illuminance and color temperature can be individually realized at the same precision as they are realized by a method using a chorma meter. Using this method, the chroma meter is unnecessary and the operation of setting the color temperature of each light separately is also not required. From these points, we can expect that by integrating the proposed method into the Intelligent Lighting System, individual illuminance and color temperature can be provided to each office worker.

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, JOURNAL of the Illuminating Engineering Society, pp.131-142, Winter 2000

[3] PHILIPS. ActiViva club. http://www.lighting.philips.co.jp/

[4] Hajimu Nakamura, Yoshinori Karasawa, Tadashi Kuwata, Hidemi Kashiwagi, Isao Maeda, Mitsuo Yokoyama, Mineo Suzuki, Mari Miyata, Psychological influence of illuminance and color temperature on atmosphere, Proceeding of Annual Conference of The Illuminating Engineering Institute of Japan, Vol.29, pp.265, 1996

[5] Y.Taniguchi, M.Miki, T.Hiroyasu, M.Yoshimi, Preferred illuminance and color temperature in creative works, 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp.3255-3260, 2011

[6] Maiko Ashibe, Mitsunori Miki, and Tomoyuki Hiroy- asu. Distributed optimization algorithm for lighting color control using chroma sensors. pp. 174-178, 2008.

[7] 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.

[8] Risa Kawashima, Mitsunori Miki, Hisanori Ikegami, Yohei Azuma and Hiroto Aida, Implementation to Provide Individual Illuminance and Color Temperature in an Intelligent Lighting System without Chroma Meter, The 2014 International Conference on Artificial Intelligence (ICAI2014)
[9] Shingo Tanaka, Mitsunori Miki, Tomoyuki Hiroyasu, and Mitsuharu Yoshikata. An evolutional optimization algorithm to provide individual illuminance in work- places. pp. 941-947, 2009.

Dynamic Decision-making with Reflecting and Learning for Self-adaptive Systems

Misoo Kim [1], **Hohyeon Jeong**[1], **Eunseok Lee**[1]
[1]Information and Communication Engineering, Sungkyunkwan University, Suwon, Republic of Korea

Abstract - *Self-adaptive systems can change their sub-goals and behaviors to achieve an ultimate goal in a changing environment. Existing approaches can be adapted in environments using pre-defined utility functions and human strategies; however, human designers cannot perfectly assume and predict all possible system environments during design time. We propose a new method of dynamic decision-making for real-time adaptive strategies through the following steps. First, we design a dynamic decision network (DDN) with environmental factors and goal model;. Second, we evaluate and predict goal satisfaction using the DDN. We further propose a dynamic reflection method that changes the model using real-time data. We applied the proposed method in Robocode and verified its effectiveness by comparing it with static decision-making.*

Keywords: self-adaptive system, dynamic decision-making, dynamic decision network, reflection model

1 Introduction

Self-adaptive systems (SASs) can change their own sub-goals and behavior without a human operator to achieve an ultimate goal in a variety of environments [1]. To change sub-goals and behavior, researchers have proposed goal-model-based SASs because they provide information for the evaluation of goal satisfaction and the formulation of rational decisions [1–2].

Existing decision-making techniques for goal-model-based SASs have pre-defined utility functions and the strategies depend on assumptions at the design time [3–9]. However, during the design time, system designers cannot perfectly assume and predict all possible system environments that the system will be deployed in; as a result, neither goal achievement nor proper adaptability can be guaranteed. To cope with this limitation, SASs require dynamic-decision-making that can consider the variety of runtime environments that can be known after deployment.

Among Artificial Intelligence (AI) techniques, the dynamic decision network (DDN) [10], which is used in our research, is a proper model for the making of dynamic decisions. It can design alternatives and outcome-utility values for all the alternatives using environmental information. Consequently, it can make the best possible decisions within changing environments.

In this paper, we propose a method of dynamic decision-making for the deployment of SASs in runtime environments that are not known prior to deployment. To represent the system goals and environment, we designed a DDN using the goal model and environmental information so that decision-making can occur during runtime. After system deployment, our DDN can evaluate and predict each goal satisfaction during runtime using real-time environmental information. Furthermore, the DDN is dynamically reflective, so it can make suitable decisions continuously in a variety of runtime environments. SASs can therefore adapt appropriately to their various runtime environments through the use of the proposed method, which considers both the real system environment and goal satisfaction, even when the environmental knowledge of the system designers is imperfect. We applied our proposed method in the Robocode context [11] and verified its effectiveness through a comparison with the conventional static-decision-making method.

The rest of the paper is organized as follows: Section 2 describes related work; Section 3 presents the proposed approach; and in Section 4, we describe how we applied the proposed method in Robocode and evaluated it using experiments, thereby allowing for the presentation of our conclusions in Section 5.

2 Related Works

2.1 Decision-making in Goal-model-based SASs

The goal model represents system goals according to the stakeholders' objectives captured by Requirement Engineering. Figure 1 depicts a goal model for the Robocode robot, which is utilized in our research. It is composed of the *goals* (a circle) that should be achieved for the system, *tasks* (a hexagon) that form the operations of the system for the achievement of the goals, and *links* (a solid-arrow) that indicate the goal refinement achieved by facilitating connections via AND/OR decompositions. To satisfy a decomposed goal, all/at least one of the sub-goals should be satisfied through an AND/OR link; therefore, alternatives can be captured in accordance with an OR decomposition. Goals can be divided into *hard goals*, functional requirements with *conditions* (a dotted-rectangle)

[3] that represent satisfactory goal conditions, and *soft-goals,* non-functional requirements.

For example, to satisfy the goal "Move for Avoidance" at the bottom-left side of Figure 1, the condition "Event!=hitBy" must be satisfied, which prevents the robot from being hit by an enemy missile. Furthermore, to satisfy that condition, the robot chooses a task from among "Random Movement," "Normal Movement," and "No Movement" based on its decision-making. The goal model provides information for the evaluation of goal satisfaction, and rational decisions are consequently made regarding the OR decomposition [1-2]. Therefore, we use it in this study to create adaptation strategies.

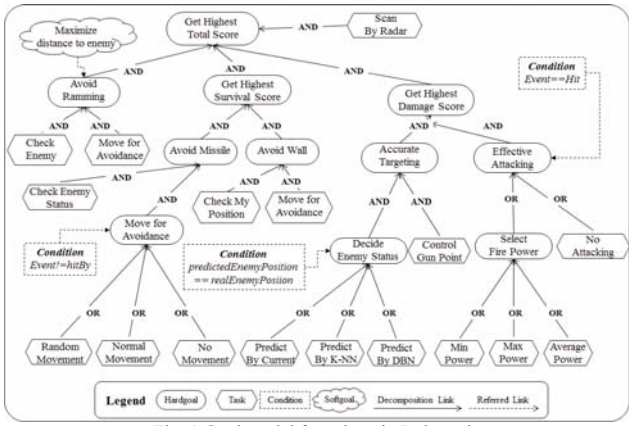


Fig. 1 Goal model for robots in Robocode

The decision-making techniques of existing goal-model-based SASs evaluate or predict goal satisfaction using pre-defined utility functions, which allows them to determine adaptation strategies regarding their runtime environments [3–8]. Morandini et al. [3] introduced the concept of *Condition* for the representation of satisfactory goal conditions on the environmental level; because it can be mapped with environmental factors, we adopted this method. Baresi et al. [4] and Nunes et al. [5] used pre-defined utility functions to make decisions in handling the uncertainties of soft-goal satisfaction. Menasce et al. [6] used a linear utility function with a static threshold that can be defined according to goal satisfaction. Welsh et al. [7] proposed a *Claim* for the expression of an environment that is assumed on a goal model during the design time. The previously mentioned research studies, however, would lead to the making of unsuitable decisions in real time because none of them can predict the adaptation result after executing a strategy. Chen et al. [8] predicted goal satisfaction based on the probability of success or failure. To adapt to the runtime environment, they proposed an updating probability method regarding goal satisfaction and an evaluation method regarding overall system utility values using [12]. With the exception of the real environmental information that can affect goal satisfaction, their approach considers only an abstract goal model. In our previous research [13] on adapting to the real system environment, we used a DDN in the design for system-environment information to illustrate both a deployed environment and goals.

2.2 DDN-based Decision-making

The DDN [10] is a proper AI technique for making dynamic decisions. Figure 2 shows a DDN with its corresponding components and time slices. The DDN uses a time slice to extend decision networks (DNs), using *temporal dependencies* (a dotted arrow) to express changes within the time domain. The DNs are composed of *decision nodes* (a rectangle) for a set of alternatives, *utility nodes* (a diamond) that compose a utility table (UT) to estimate the outcome-utilities of the alternatives, *chance nodes* (a white circle) such as random variables with a conditional probability table (CPT) for the occurrence probabilities of events, and *causal dependencies* (a solid arrow) that exist between the nodes; here, chance nodes are inferred for the next time slice through observed *evidence nodes* (a gray circle). The system environmental information (EI) can be designed using the set of chance nodes about environmental factors (EFs) and their dependencies. A DDN supports decision-making as well as evaluation of goal satisfaction for SASs because it can predict EI and the outcome-utility values of decisions at varying times.

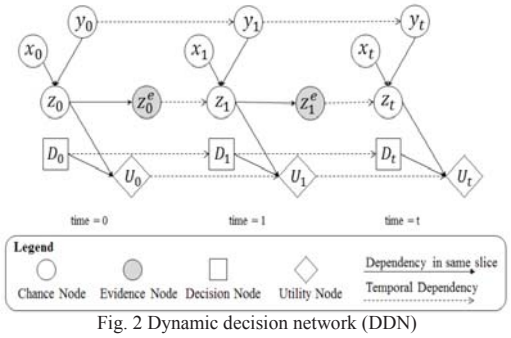


Fig. 2 Dynamic decision network (DDN)

The DDN has already been verified as part of a dynamic-decision-making method for a variety of domains: the auto-defense system for an aircraft [14], a groundwater management system with a climate-prediction capability [15], and an assisted-living system that assigns caregivers to an elderly man [16]. In the corresponding research, a DDN used to predict EI makes decisions for a specific domain and not for a variety of domains; however, we propose a general purpose DDN-modeling process derived from the goal model and EI.

For SASs, Bencomo et al. [9] proposed a modeling framework that transforms the goal model into a DDN for decision-making by SASs. The framework can predict soft-goal satisfaction based on a pre-defined probability in the abstract goal model: the pre-defined probability should lead to the making of suitable decisions in real time. The application of such a DDN concept is limited to the use of runtime EFs for monitoring design-assumption deviation, which is to say that it considers only a partial goal model that contains soft goals and tasks. The research focus of the previous studies and our focus differ in how they handle design assumptions. That is, our current work considers both the runtime-EI and the overall goal model in the design of the DDN, extending our previous research [13], which was limited to the design of soft-goals.

3 Proposed Approach

3.1 Goal-model-based DDN-modeling

In this section, we describe a modeling process that uses a DDN to fill the gap between the goal model and the system by considering the deployed environments, as shown in Figure 3. This method extends our previous paper [13] and integrates its mapping for soft goals.

A DDN structure can abstractly design decision making model using a goal model. First, we narrow down the overall goal model into a compacted goal model consisting of leaf goals that comprise only an OR decomposition. Second, we map the compacted goal model onto the DDN. The top of Figure 3 explains how the goal model can be abstractly mapped onto the DDN. The tasks, conditions, soft goals, and hard goals in the goal model are mapped with the decision, evidence-based chance, static chance, and utility nodes in the DDN, respectively. Third, to describe the EI, we extend the abstract DDN to add the EFs that have causal relationships with EFs expressed within conditions and to add temporal dependencies with pre-defined *t* time slices between the same EFs. Thereby, a DDN has the EI, which is composed of EFs and the causality among them, which is helpful in representing the real system environment. Fourth, to build a CPT of a DDN, the system designers define the discretization criteria to represent the states of the chance nodes and subsequently build the CPT with those discrete values using the knowledge of domain experts or learning algorithms. In the design of a complex system, however, the number of EFs would be too great, making the inference time too long. To deal with that problem, designers can use a feature-selection algorithm, such as that in [17], to select important EFs. We suggest a mutation information–based feature selection algorithm such as that in equation 1, where CEF is an EF for a specific condition, and EF is all EFs except CEFs. A pre-defined threshold can be used to decide which EFs to use as CEFs.

$$I(CEF;EF)=\sum\sum p(CEF,EF)\log(p(CEF,EF)/p(CEF)p(EF)) \quad (1)$$

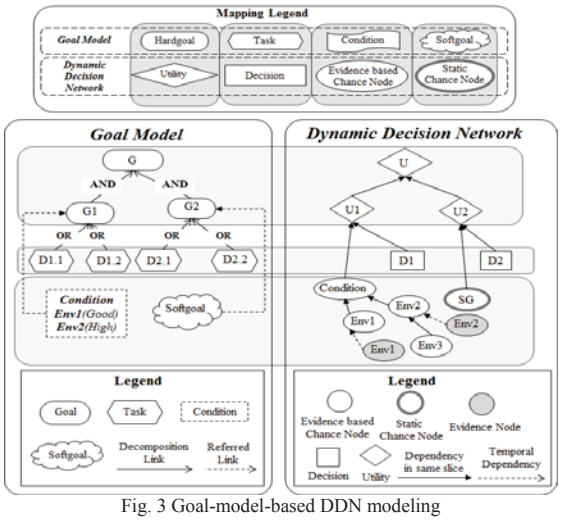


Fig. 3 Goal-model-based DDN modeling

Fifth, to build a UT of for a DDN, according to the experts or the historical data, we design the UTs of the leaf-utility nodes are designed according to the outcome-utility values, depending on the goal conditions and each alternative. The parent-utility nodes comprise UTs with sub-goal weights for the satisfaction of their own goals.

3.2 Dynamic Decision-making Techniques

In terms of dynamic decision-making, our approach maintains and updates a DDN that is stored in the Knowledge base. Figure 4 presents an overview of the proposed framework. Algorithm 1 programmatically describes the proposed dynamic decision-making techniques in our framework.

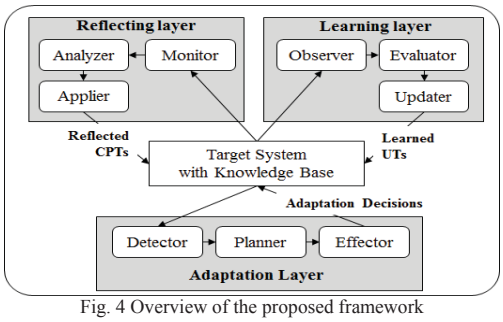


Fig. 4 Overview of the proposed framework

Algorithm 1 Dynamic Decision Making with Reflection Model

Require: DDN, MonitorableData, HistoricalData, PredictedData, a, b

// 1. Reflecting Layer
$currentState \leftarrow discretize(MonitorableData)$
while $i < DDN.chanceNode.length()$ **do**
  $stateNumber \leftarrow DDN.chanceNode.get(i).stateNumber()$
  while $j < stateNumber$ **do**
    $state \leftarrow chanceNode.get(i).getState(j)$
    if $currentState.contain(state)$ **then**
      $state \leftarrow state.getProbability() + 1/10^a$
    else
      $state \leftarrow state.getProbability() - \frac{1}{10^a} \times \frac{1}{stateNumber-1}$
    end if
  end while
  DDN.applying($state$)
  $i \leftarrow i+1$
end while

// 2. Learning Layer
$currentUtility \leftarrow evaluator(MonitorableData)$
if $currentUtility! = predictedData.getUtility()$ **then**
  $previousDecision \leftarrow HistoricalData.getPrevDecisions()$
  while $i < previousDecision.getNumber()$ **do**
    $table \leftarrow DDN.utilityNode.getTable(i)$
    $historicalUtility \leftarrow \frac{HistoricalData.getUtilities(b)+currentUtility}{b+1}$
    $table \leftarrow table + historicalUtility$
    DDN.updating($table$)
    $i \leftarrow i+1$
  end while
end if

// 3. Adaptatiob Layer
$utilityList \leftarrow calculateUtility(DDN)$
ordering(utilityList)

return utiiltyList.getTheHighestUtility()

3.2.1 Reflecting Layer

The *Reflecting Layer* comprises the *Monitor*, *Analyzer*, and *Applier* to reflect the system environment in the CPT.

The *Monitor* regularly collects the raw data that represents the EI of the target system. According to the need, it also transforms the raw data into discrete values using discretization criteria.

The *Analyzer* checks whether the predicted EI matches the real EI. If a difference is found, it calculates and updates the new conditional probability in the CPTs using equations 2 and 3 for the CPTs of child chance nodes, and equations 4 and 5 for the CPTs of root chance nodes based on the Dirichlet Distribution [18] and our previous research [13]. We use the CPT to predict the EFs in the runtime environment and represent the EI in the *Planner* in the *Adaptation Layer*.

$$P(EF_i=x|EF_j,..,EF_n) = previousP(EF_i=x|EF_j,..,EF_n)+1/10^{\alpha} \quad (2)$$

$$P(EF_i=x|EF_j,..,EF_n) = previousP(EF_i=x|EF_j,..,EF_n)+1/10^{\alpha}\times 1/(m-1) \quad (3)$$

$$P(EF_k=x) = previousP(EF_i=x)+1/10^{\alpha} \quad (4)$$

$$P(EF_k=x) = previousP(EF_i=x)+1/10^{\alpha} \times 1/(m-1) \quad (5)$$

According to α, which indicates how much of the current state needs to be reflected, equations 2 and 3 show a way to update the conditional probability of the ith EF (EF_i) using m discrete values given from the jth (ENV_j) to the nth (ENV_n), which are the parent nodes of the ith EF. Using the previous CPT, equation 2 is used to increase the probability of the currently evaluated discrete value x, and equation 3 is used to decrease the probability of the remaining $m-1$ discrete values, with the exception of the current discrete value. Furthermore, the CPT regarding the temporal dependency can be updated using both equations 2 and 3. Given that the current time slice is i, information for time slices from j (= $i-1$) to n (= $i-t$ and $t<i$) were available for this study. The root node indexed by k, which does not have parent nodes, would be updated using equations 4 and 5; accordingly, CPTs would be maintained on a variety of runtime environments. In addition, previous CPTs are shifted to the $t-1$ time slice. After reflecting the real environmental information in CPTs, the *Applier* applies the new CPTs to the DDN of the target systems.

3.2.2 Learning Layer

The *Learning Layer* comprises the *Observer, Evaluator,* and *Updater* to learn the utility values for the executed decisions and real environmental information.

The *Observer* step regularly observes the completion of executed decisions and collects the raw data. The *Evaluator* calculates the real utility values and checks whether the predicted utility values and real utility values are the same after executing decisions.

The *Updater* calculates and updates the new utility values for the UTs using equations 6 and 7, based on our previous research [13]. In our approach, the UT is used to predict the utility value through each alternative and to calculate the expected utility in the *Planner* in the *Adaptation Layer*. It is difficult to know to what degree and by which action the system acquires the overall utility value (*OV*); to estimate this, we used the weights of the parent-utility nodes, which explain the priorities of the lower goals to the upper goals. Equation 6 illustrates how a residual value is divided between the predicted utility value at a previous time (*predicted(OV)*) and the real-time utility value (*real(OV)*) into a partial utility value according to the weight of the rth parent-utility node with respect to the qth leaf-utility node ($U(U_q|U_r)$). Equation 7 represents the predictive utility value of the ith utility node through a previous decision (D). The UT expresses the utility value according to the amount the system can obtain for each alternative when the EFs encounter a specific discrete value. We use the k-nearest neighbor (k-NN) algorithm [19] to predict the utility value for the next time the same decision is made in the same environment. Using the k-NN as a basis, k indicates how many previous time slices should be considered. The Utility Updater calculates an average value among the previously estimated utility values ($V(U_q)$) for the k time slices and the current utility value for the ith utility node. Then, adding the average value to the previously predicted utility value produces a predictive utility value for the ith leaf-utility node with a given alternative (D) and the EI (EI).

$$V(U_q)=(real(OV)-predicted(OV)) \times U(U_q|U_r) \quad (6)$$

$$U_q(D\backslash EI)=previousU_q(D|EI)+(V(U_q)+ \sum_{t=0}^{k} V_t\,(U_q))/(k+1) \quad (7)$$

3.2.3 Adaptation Layer

The *Adaptation Layer* comprises the *Detector, Planner,* and *Effector* to make adaptive decisions using reflected and learned DDN.

The *Detector* step periodically senses the set of raw data and calculates the real utility values.

The *Planner* infers the subsequent EFs and obtains probabilities regarding the discrete values of the goal-satisfaction conditions. At the current time t, equation 8 is used to predict the EFs (X) for the next time (X^{t+1}), which composes a parent EF (Y^t) and the current ith EF (X^t) when each alternative (D_{t+1}) is performed, based on the Markov property [10]. The *Planner* then calculates the expected utility values for each strategy according to equation 9, followed by the formulation of a decision that includes the highest expected utility, which is sent to the *Effector*.

$$P(X^{t+1}|X^t,Y^{t+1},D_{t+1}) = P(X^{t+1}|X^t,Y^{t+1},D_t)P(X^t|X^{1:t},Y^t,D_{1:t-1}) \quad (8)$$

$$ExpectedUtility = \sum_{j=1}^{m} U(U_i|U_j)\times \sum_{i=1}^{n} U_i\,(X^{t+1}|D_{t+1}) \times P(X^{t+1}|X^t,Y^{t+1},D_{t+1})) \quad (9)$$

4 Case Study

4.1 ROBOCODE

To verify the effectiveness of the proposed method, we applied it in the context of Robocode [11], a Java-based programming game wherein virtual robots battle against teams or individuals. When a battle is started, it can be watched, along with the result and score after its end, on a screen. The game is thus advantageous for quantifying adaptation results.

A robot consists of a *Vehicle Heading* for movement; *Missile Heading* for a turret that fires a missile to decrease the energy of oneself or an enemy; and a *Radar Heading* to detect enemies and collect their information. These components are controlled independently, so they form the basis of a strategy.

During a battle, a robot can obtain two types of information, as shown in Table 1. If the radar sensors an enemy robot, it collects *enemy information*. The robot can use that information to make decision. After firing a missile, which disappears from the screen after it hits or misses an enemy, the robot receives a *Hit Event* or *Miss Event*; alternatively, if a robot is hit by a missile, it receives a *HitBy Event*.

A robot might not use an appropriate strategy to avoid an enemy missile or accurately attack an enemy because it cannot collect perfectly accurate information about the time that an enemy fires a missile and the missile's trajectory. The robot therefore needs to be able to predict the enemy state in a variety of systemic runtime environments given unknown robots that can be known only after a battle.

Table 1: Obtainable Information

Type	Name	Description	Data Range
Enemy Infor-mation	Distance	Distance from my robot to scanned robot	0 to diagonal length of field
	Heading	Vehicle heading	0° to 360°
	Bearing	Direction relative to my current heading	-180° to 180°
	Energy	Remaining energy	0 to 100, 100 to ∞
	Velocity	Velocity (negative is backward)	-8 to 8 (pixel/sec)
Event	Hit	My bullet hit the target	
	HitBy	Hit by an enemy missile	
	Miss	My fired bullet missed	

4.2 Experimental Setup

4.2.1 Goal-model-based DDN Modeling

Each goal condition is designed according to Table 2.

Table 2: Goal Satisfaction Criteria

Goal	Condition
Avoidance	Event != hitBy
Attacking	Event == Hit || Event != Miss
Targeting (control)	enemyVectorX == realVectorX enemyVectorY == realVectorY

We chose the robot "EpeestMicro," which was recently published by the community [20], and used it as a base-bot. We modeled the DDN using OpenMarkov [21] to express a design assumption, which we collected until we had 10 rounds of base-bot information. The battle type is a man-to-man fight. We built the EI of a DDN using the hill climbing algorithm and Bayesian scoring, with the historical base-bot information serving as the basis. We discretized each EF in reference to [22]. Each factor affects itself on the next occasion, when it is used as an evidence node.

According to the mapping method shown in Figure 3, we designed each UT of for leaf-utility nodes using the historical data scores. We designed the decision nodes in the DDN using the tasks in Table 3, which shows the robot's goals and tasks from Figure 1 and whether the base-bot considers the goals and tasks. We designed the tasks of our robot from the base-bot, but the execution points of our tasks are different from those of the base-bot because we used the dynamic EI—that is, the dynamic-utility function—to evaluate goal-satisfaction.

Table 3: Basic Strategies for Goals

Goal	Tasks	Base-bot
Move to Avoid Missile	Random	O
	Normal	O
Accurate Targeting (Control)	Current	X
	K-NN	O
	DBN	X
Effective Attacking	Min. power	X
	Average power	O
	Max. power	X

The number of time slice t was set to 3, and we used three pieces of historical evidence to infer the factors on the next occasion.

4.2.2 Dynamic Decision-making

We designed the parameters shown in Table 4.

Table 4: Parameter Settings

Layer	Module	Parameter
Reflecting	Analyzer	$\alpha = 4$
Learning	Evaluator	Overall utility = real-time score
	Updater	$k = 4$

4.3 Evaluation Method

Our method can predict goal satisfaction, whereby we use events to determine an avoidance and firing strategy and "enemyVectorX/Y" to control the missile and radar heading in Table 2, according to whether each condition will be satisfied. The prediction accuracy is 100 % when a prediction result is correct; otherwise, the prediction accuracy is 0 %. We used the following three steps to evaluate the proposed method.

First, to evaluate the effectiveness of the DDN-based prediction, we compared the average prediction accuracies of the base-bot and the DDN-based bot, excluding the reflecting and learning layers. Second, to identify whether the reflecting layer provides adaptability in a variety of runtime environments that weren't considered during design, we compared the

average prediction accuracies of the DDN-bot and the R-DDN-bot based on the reflecting layer using a battle with sample robots not considered during the design. Last, to verify the effectiveness of our overall framework for dynamic decision-making, we compared the score results of the base-bot with those of the Adaptive-DDN-bot based on our framework (A-DDN-bot).

4.4 Evaluation Results

4.4.1 Effectiveness of DDN-based prediction

This section illustrates that the DDN-bot designed using base-bot information can accurately predict goal satisfaction. Table 5 shows a comparison of the average prediction accuracies with the base-bot: the prediction accuracy of the DDN-bot is higher than that of the base-bot.

Table 5 Accuracy Comparison of Goal Satisfaction Predictions

Accuracy	Base-bot	DDN-bot	Effect
Event	53.9 %	72.2 %	**+ 34.0 %**
Control	70.2 %	74.0 %	**+ 5.2 %**

When it scans an enemy, the base-bot predicts that enemy's next position using a linear prediction based on a pre-defined utility function. Alternatively, the DDN-bot uses an EI-based prediction. The base-bot assumes that enemies move linearly, but the DDN-bot assumes that enemies select a movement strategy according to their own current information.

For example, the base-bot fires a missile of average power to the predicted position of the enemy whenever it scans an enemy, meaning that the base-bot uses a pre-defined strategy. The DDN-bot predicts events such as the "Hit" or "Miss" of a missile; furthermore, it fires a missile with low or high power when the probability of a "Miss" or "Hit" is high, respectively. In addition, the base-bot moves to avoid a missile fired by an enemy when the enemy's energy decreases, but an enemy's energy can decrease for any reason, including ramming into a wall. The DDN-bot predicts an event, such as "HitBy," with respect to an enemy's missile based on the current enemy's information. As a result, the DDN-bot considers much more information, resulting in higher accuracies than the base-bot regarding the goal "control" and events. The effect on "control" is lower than that on "event" because the base-bot uses k-NN prediction that is effective on continuous movement.

4.4.2 Effectiveness of Reflecting Layer

First, we compared the average prediction accuracies of the DDN-bot and the R-DDN-bot, as shown in Table 6.

Table 6: Accuracy Comparison Between DDN and R-DDN

Accuracy	DDN	R-DDN
Event	72.2 %	**76.8 %**
Effect		**+6.7%**
Control	74.0 %	**77.5 %**
Effect		**+4.7%**

Next, to identify the adaptability according to a variety of runtime environments, the DDN-bot and R-DDN-bot fought against other bots that we didn't consider during the design time. Figure 5 illustrates the trend regarding accuracy during the battles with bots not considered in the DDN. Table 7 shows the results of a battle between the R-DDN and the static DDN, revealing that the R-DDN-bot is much more effective than the DDN-bot across a variety of runtime environments.

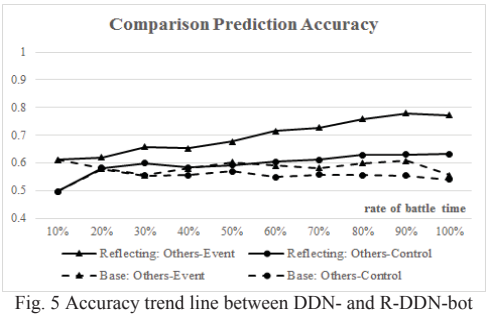


Fig. 5 Accuracy trend line between DDN- and R-DDN-bot in battle with other robots

Table 7: Accuracy Comparison in Battle Environment with Other Robots

	DDN		R-DDN	
	Event	*Control*	*Event*	*Control*
SpinBot	54.61 %	72.8 %	59.6 %	76.8 %
Corner	67.47 %	63.9 %	62.5 %	63.4 %
FireBot	69.76 %	40.5 %	75.4 %	51.7 %
Crazy	43.74 %	43.7 %	81.3 %	46.5 %
Average	**58.6 %**	**55.2 %**	**69.7 %**	**59.6 %**
Effect			**+18.9%**	**+8.0%**

The results show that although the DDN was modeled based on a specific robot (base-bot), it successfully used real-time information.

4.4.3 Effectiveness of Dynamic Decision-making

Having verified the reflecting layer, we show the higher scores of the A-DDN. Table 8 explains the battle result for fights between the base-bot and the A-DDN-bot. After 10 rounds in 10 battles, both the total score and the bonus score of the A-DDN-bot are higher than those for the base-bot.

Table 8: Battle Result with Base-bot

	Total	Survival	Survival Bonus	Damage	Damage Bonus
Base-Bot	3451	1200	245	1801	205
A-DDN	**3937**	**1455**	**298**	**1939**	**245**
Effect	**+486**	**+255**	**+53**	**+138**	**+40**

Table 9 compares the results from the battles with the other bots with those from the battle with the A-DDN-bot. For the proposed method with dynamic decision-making, the enemy robot score decreased by approximately 6.7 %, while our robot's score increased by approximately 13.0 %. From Table 8 and Table 9, it is evident that the results for the robot with dynamic decision-making and the proposed method are more favorable than those for the static robot.

Table 9: Score Result Comparison with Other Bots

Enemy Bot	vs. Base-Bot		vs. A-DDN-Bot	
	Enemy	*Mine*	*Enemy*	*Mine*
SpinBot	32	1387	27	1488
Corner	36	1345	34	1693
FireBot	23	1471	18	1721
Crazy	104	1614	103	1674
Average	48.75	1454.25	45.5	1644
Effect (%)			**-6.7%**	**+13.0%**

5 Conclusion and Future Work

A limitation of the existing decision-making protocol for SASs has been that neither goal achievement nor proper adaptability could be guaranteed because the design assumptions inadequately consider the execution environment. To solve this problem, we have here proposed a dynamic-decision-making technique in the design method using a goal-model-based DDN and a new framework.

We validated the effectiveness of our proposed method using Robocode. According to our experimental results, the prediction accuracy of the base-bot with a static utility is lower than that of the environmental-information-based DDN-bot. Using the proposed method, the accuracy of the model with an inadequate design assumption increases, and in other deployment environments, the changing accuracy trend increases. Also, in comparison with the static DDN, the prediction accuracy for goal satisfaction increased by an approximate average of 13.45 %. Last, the A-DDN-bot obtained a higher score than the static-utility-based bot, and the enemy robot score decreased by approximately 6.7%, while our robot's score increased by 13.0%.

In the proposed method, the environmental factors are determined according to all the information that can be obtained in the environment; however, the monitoring of the overhead must also be shown. Our future work will therefore study preprocessing and a design process to decrease the overhead. Moreover, our approach should be validated in various case studies.

6 Acknowledgements

This research was supported by the Next Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) and is funded by the Ministry of Education, Science and Technology (No. 2015045358) and the MISP (Ministry of Science, ICT & Future Planning), Korea, under the National Program for Excellence in Software, which is supervised by the IITP (Institute for Information & Communications Technology Promotion).

7 References

[1] R. D. Lemos et al., "Software engineering for self-adaptive systems: A second research roadmap." Software Engineering for Self-Adaptive Systems II, pp. 1-32, 2013.

[2] C. Krupitzer et al. "A survey on engineering approaches for self-adaptive systems." Pervasive and Mobile Computing Vol. 17, pp. 184-206, 2015

[3] M. Morandini et al., "Towards goal-oriented development of self-adaptive systems." Proceedings of the international workshop on Software engineering for adaptive and self-managing systems (SEAMS 2008). pp. 9-16. 2008.

[4] L. Baresi et al., "Fuzzy goals for requirements-driven adaptation." Proceedings of the 18th International IEEE Requirements Engineering Conference (RE 2010), pp. 125-134. 2010.

[5] I. Nunes et al. "Softgoal-based plan selection in model-driven bdi agents." Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems (AAMAS 2014), pp. 749-756, 2014.

[6] D. Menascé et al. "Sassy: A framework for self-architecting service-oriented systems." IEEE Software, Vol. 28, No.6, pp. 78-85, 2011

[7] K. Welsh et al. "Self-explanation in adaptive systems based on runtime goal-based models." Transactions on Computational Collective Intelligence XVI, pp. 122-145, 2014.

[8] B. Chen et al., "Requirements-driven self-optimization of composite services using feedback control." IEEE Transactions on Services Computing, Vol. 8, No. 1, pp. 107-120, 2015.

[9] N. Bencomo et al., "Supporting decision-making for self-adaptive systems: from goal models to dynamic decision networks." Requirements Engineering: Foundation for Software Quality, pp. 221-236, 2013.

[10] S. J. Russell et al., Artificial intelligence: A modern approach, 2nd ed., Prentice Hall series in artificial intelligence. Prentice Hall. 2003.

[11] ROBOCODE, http://robocode.sourceforge.net/

[12] B. W. Boehm, "Value-based software engineering: Overview and agenda." Value-based software engineering, pp. 3-14, 2006.

[13] M. S. Kim et al., "Dynamic Decision Making for Self-Adaptive Systems Considering Environment Information.", Korea Institute of Information Scientists and Engineers, in press

[14] P. C. G. Costa, "The fighter aircraft's auto-defense management problem: a dynamic decision network approach." Master's thesis, George Mason University, 1999.

[15] J. L. Molina et al., "Dynamic Bayesian networks as a decision support tool for assessing climate change impacts on highly stressed groundwater systems." Journal of Hydrology, Vol. 479, pp. 113-129. 2013.

[16] N. Z. Naqvi et al., "The right thing to do: Automating support for assisted living with dynamic decision networks." Proceeding of the 10th International Conference on Ubiquitous Intelligence and Computing, and 10th International Conference on Autonomic and Trusted Computing (UIC/ATC), pp. 262-269, 2013.

[17] N. Mogharreban et al., "Self-Adaptive Feature Selection Strategy for Multi-type Object Classification Problems." Proceedings on the International Conference on Artificial Intelligence (ICAI), pp. 884-889, 2013.

[18] T. T. Wong, "Generalized Dirichlet distribution in Bayesian analysis." Applied Mathematics and Computation, Vol. 97, No. 2, pp. 165-181, 1998

[19] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression", The American Statistician, Vol. 46, No.3, pp. 175–185, 1992.

[20] RobotSearch, http://robocoderepository.com/, 2015.6.

[21] OpenMarkov, http://www.openmarkov.org/

[22] J. Frokjaer et al, "Robocode: Development of a Robocode team", The university of Aalborg, pp. 72-78, 2004.

Proposal for a Beacon-type Intelligent Lighting System Automating the Toggling of the Occupancy Status Using a BLE Beacon

Sota NAKAHARA[2], Mitsunori MIKI[1], Kohei YAMAGUCHI[2], Shinya DAINAKA[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.Under the current Intelligent Lighting System, each worker manually toggles the occupancy status of their seat, but as a result of demonstration test, it was found that there were many workers who did not toggle it appropriately. Consequently, unnecessary lamps remained on even though there was no worker, reducing energy efficiency. This study thus proposes a Beacon-type Intelligent Lighting System. As a result of verifying the effectiveness of the Beacon-type Intelligent Lighting System proposed, it was found to be able to detect correctly the occupancy status of a seat and the location of a worker and satisfy the illuminance requirement by the worker.*

Keywords: Lighting, Intelligent Lighting System, BLE Beacon, Attendance Management

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 luminous 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. 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. 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].

The Intelligent Lighting System judges that it is not required to provide illuminance to a seat unoccupied by any worker. By dimming or turning off lighting for an area around the seat in such a manner as not to affect working space of other workers, the system realizes even higher energy saving. As a result of demonstration tests at real offices, however, it was found that workers did not appropriately toggle the occupancy status of their seat. It was thus found that there were lighting fixtures which provided brightness more than required even though there was no worker.

We thus propose a Beacon-type Intelligent Lighting System, which identifies the location of a worker by using a BLE beacon (hereinafter "beacon") and a smartphone to automate the toggling of the occupancy status in the Intelligent Lighting System. The Beacon-type Intelligent Lighting System determines the degree of proximity between a beacon and a smartphone by using the signal strength of radio wave transmitted by the beacon and received by the smartphone and then determines whether a seat is occupied or not by using the degree of proximity. It is intended to improve the convenience of the Intelligent Lighting System for workers by having the system toggle the occupancy status of a seat, which was previously toggled manually by a worker, by using a beacon and a smartphone.

2. Intelligent Lighting System

2.1 Overview

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.

A lighting control device evaluates the effectiveness of the current lamp lighting pattern on the basis of illuminance information obtained from an illuminance sensor and power information obtained from a power meter. It is intended to minimize power consumption while satisfying the illuminance constraint required by each worker by repeatedly

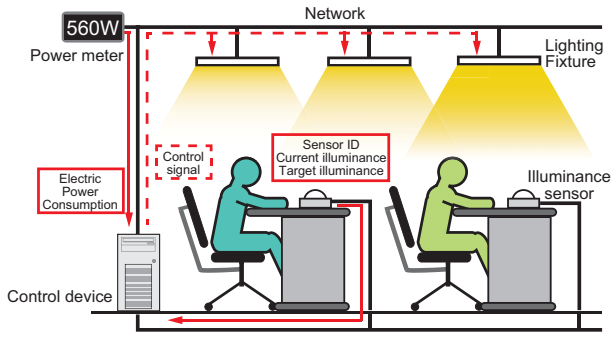


Fig. 1: Configuration of Intelligent Lighting System

changing a lamp lighting pattern slightly and evaluating its effectiveness.

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 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 flow of control by the Intelligent Lighting System using 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 evaluation value in accordance with the objective function described below.
(5) Generate the next luminance in accordance with the illuminance/luminance influence factor and light the lighting fixtures at the next luminance.
(6) Obtain values measured by illuminance sensors and a power meter.
(7) Calculate the evaluation value under the lighting condition in (5).
(8) If the evaluation value of the objective function is improved, accept the next luminance; otherwise revert to the original luminance.
(9) Return to step (3).

An illuminance/luminance influence factor is set for each lighting fixture in accordance with its positional relationship with each illuminance sensor, and directionality is given to a random change in luminance in accordance with the degree of the effect.

The objective function of each function is given by Equation (1).

$$f_i = P + w\sum_{i=1}^{n} g_i \quad (1)$$

$$g_i = \begin{cases} 0 & (Lc_i - Lt_i) \geq 0 \\ R_i(Lc_i - Lt_i)^2 & (Lc_i - Lt_i) < 0 \end{cases}$$

$$R_i = \begin{cases} r_i & r_i \geq T \\ 0 & r_i < T \end{cases}$$

n:number of illuminance sensors, w:weight
P:power consumption[W], L_c:current illuminance[lx],
L_t:target illuminance[lx]
r_i:illuminance/luminance influence factor[lx/cd]
T:threshold

The objective function given by Equation (1) is composed of power consumption P and illuminance constraint g_i and calculated for each lighting fixture. Penalty g_i 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_i, lighting fixtures affecting a given illuminance sensor can be narrowed down to those in its proximity. This enables controlling lighting fixtures far from the illuminance sensor so as to minimize power consumption.

The illuminance/luminance influence factor is a value indicating the relationship between an illuminance value obtained from an illuminance sensor and the luminance of a lighting fixture. The illuminance/luminance influence factor is given by Equation (2).

$$I = RL \quad (2)$$

I : illuminance [lx], L : luminance [cd]
R : illuminance/luminance influence factor [lx/cd]

Illuminance/luminance influence factor R is a value that depends on an illuminance environment and can be regarded as a constant so long as there is no change in an illuminance environment. Since desks are infrequently moved in an ordinary office, it is possible to measure an illuminance/luminance influence factor in advance in an environment into which the Intelligent Lighting System is to be introduced by turning on and off its lighting fixtures one by one.

2.2 Simulation based control

In the Intelligent Lighting System described in the previous section, necessary illuminance is provided to necessary locations by repeatedly changing luminance by using an optimization method based on illuminance information obtained from each illuminance sensor and power consumption

information obtained from a power meter. This method enables flexible responses to various changes in a lighting environment including changes in the layout in an office, the displacement of illuminance sensors, the degradation of lighting fixtures, and the effect of external light.

There are, however, many offices which do not have any window or hardly receive any daylight due to surrounding buildings even if they have a window. In such an environment, the effect of daylight need not be taken into account. In addition, in real offices in which demonstration tests for the Intelligent Lighting System were conducted, seats were often fixed, and the locations of illuminance sensors on desktops were determined in advance, resulting in no displacement of those sensors. Furthermore, as the service life of lighting fixtures have been prolonged in recent years owing to the spread of LED lighting fixtures, it is considered unnecessary to take the degradation of lighting fixtures into account in a short term. In such an environment, there is a method to control the Intelligent Lighting System which reduces the number of lighting control steps required to reach illuminance convergence[7]. This method searches for the optimal lamp lighting pattern without obtaining information from a power meter and illuminance sensors by estimating power information and illuminance information and repeating optimization on a computer using estimated values. By applying the lamp lighting pattern searched on a computer to lighting fixtures in a real environment, the target illuminance for each worker is realized by a single step of lighting control. The method also enables controlling the Intelligent Lighting System without an illuminance sensor because it estimates power and illuminance information on a computer. In what follows, an Intelligent Lighting System control method that reduces the number of lighting control steps required to reach illuminance convergence is called a simulation based control.

Assuming an office environment without a change in an illuminance environment and an effect of external light, the proposed system used a simulation based control that does not use an Intelligent Lighting System as an Intelligent Lighting System control method.

3. Beacon

A beacon is a device based on location technology using a low-power-consuming short-range radio technology named Bluetooth Low Energy (BLE). A beacon transmits its own identification information in a short range at a regular interval. The following information is transmitted by a beacon as identification information.

proximityUUID
: a unique ID assigned for identifying the group to which it belong

major
: the value for identifying beacons with the same UUID

minor
: the value for identifying beacons with the same UUID and major

Measured Power
: the measured value of the strength of signal received at a point 1 m from the beacon

A smartphone estimates the approximate distance from a beacon by using the beacon's identification information it receives and received signal strength indicator (RSSI).

4. Beacon-type Intelligent Lighting System

4.1 Overview

A Beacon-type Intelligent Lighting System is an Intelligent Lighting System that automates the toggling of the occupancy status of a seat by using a beacon and a smartphone. Fig. 2 gives a configuration diagram for a Beacon-type Intelligent Lighting System.

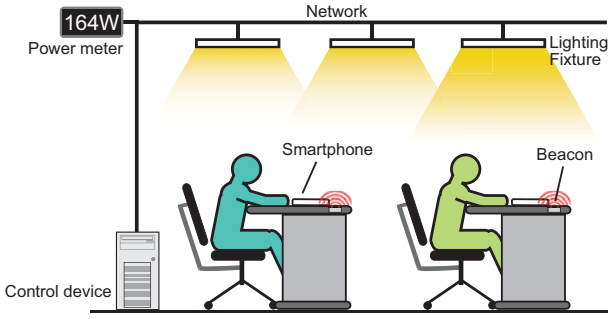


Fig. 2: Configuration Diagram for a Beacon-type Intelligent Lighting System

When a worker enters an office room and takes a seat, he/she puts a smartphone on the specified spot on the desk. Then his/her smartphone detects a beacon signal from a beacon placed on the desk and transmits an instruction to set the seat's status to "occupied" to the control PC. When the worker leaves the room, the smartphone detects that it is no longer able to receive a beacon signal and transmits an instruction to set the seat's status to "vacant" to the control PC. In this way, the occupancy status of a seat, which was toggled manually in the current Intelligent Lighting System, is toggled automatically.

4.2 System Configuration

MyBeacon Pro MB004 Ac by Aplix was used as a beacon. This beacon was placed on each of workers' desks. Consequently, in order to minimize interference with other beacons, the radio wave output from each beacon was set to the minimum at -20 dBm. The Intelligent Lighting System is controlled assuming that RSSI on a worker's smartphone is greater than the threshold for determining that the worker is

in proximity with the seat and that beacon signal is strongest at the position where the worker is seated. Shown below is the procedure starting with a worker's entrance into a room and ending with the system's setting the status of his/her seat to "occupied".

(1) A worker enters a room
(2) The worker's smartphone receives identification information from a beacon.
(3) The smartphone transmits the received identification information of the beacon and the target illuminance set by the worker to the system.
(4) The system identifies the location of the smartphone on the basis of identification information received from the smartphone and sets the occupancy status of the worker's seat to "occupied" and turns on lamps.

The system sets the status to "vacant" when the conditions for the "occupied" status are no longer met.

5. RSSI Threshold Setting for Determining Seat Occupancy Status and RSSI Measurement Experiment

5.1 Overview

This system uses RSSI on a worker's smartphone from a beacon placed on each desk to determine whether their seat is occupied or not. It was thus verified how RSSI changed in accordance with the distance between a beacon and a smartphone.

5.2 Experiment Environment

RSSI was measured to see how it changes in accordance with the distance between a beacon and a smartphone. This measurement experiment was conducted in the Smart Office Laboratory Kochikan, Doshisha University. Table 1 shows equipment used, and Fig. 3 shows the experimental environment.

Table 1: Equipment Used

Beacon	MyBeacon Pro MB004 Ac
Smartphone	iPhone 4S

A smartphone was placed at an interval of 0.1 m from a beacon, and RSSI on the smartphone was measured up to the maximum distance of 2.0 m. iPhone 4S was used, and a beacon was placed on a desk in such a manner that its indicator lamp faces up. Measurement was made five times each in four directions.

5.3 Result of Measurement

Fig. 4 shows the result of RSSI measurement experiment.

Based on Fig. 4, it was found that the average RSSI decreases gradually up to the distance of 0.7m but does not change so much at a farther distance. This system assumes that a beacon is placed on a desktop, with a smartphone to be placed on a specified spot on the desk. Therefore, the distance between a beacon and a smartphone is within about 0.2 m. The RSSI threshold in this system is thus set to -80 dBm.

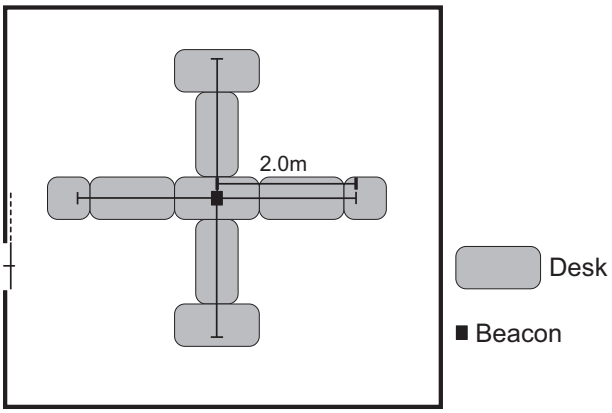


Fig. 3: RSSI Measurement Experiment Environment

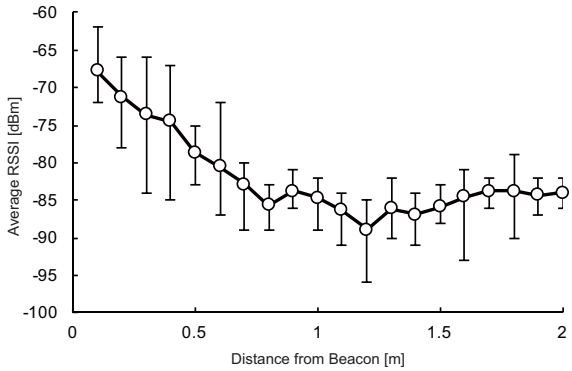


Fig. 4: Result of RSSI Measurement

6. System Operation Experiment

6.1 Overview

Using the RSSI threshold for determining whether a seat is vacant or not obtained in the previous chapter, we developed a Beacon-type Intelligent Lighting System and conducted its operation experiment. The Intelligent Lighting System was controlled based on simulation and without using an illuminance sensor in conducting the operation experiment. The equipment used was the same as that for the RSSI measurement experiment: "MyBeacon Pro MB004 Ac" beacon by Aplix and iPhone 4S smartphone. An illuminance sensor was placed on the desk of a subject worker in order to confirm that the Intelligent Lighting System works normally and satisfies the target illuminance. As with the RSSI experiment, this measurement experiment was conducted in the Smart Office Laboratory, Kochikan, Doshisha University.

Fig. 5 shows an environment in which the system operation experiment was conducted.

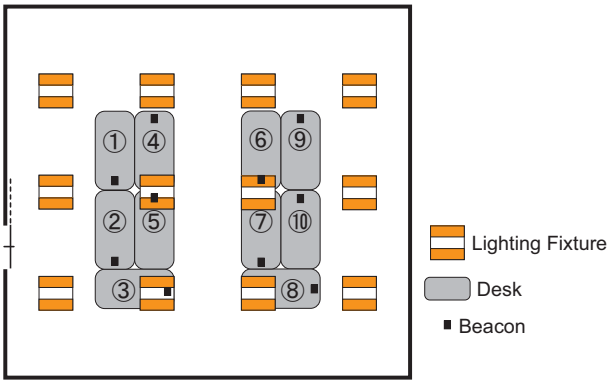


Fig. 5: System Operation Experiment Environment

The flow of the movement of workers in this experiment is shown below.

(1) Worker A (target illuminance: 300 lx) takes Seat 1.
(2) Worker B (target illuminance: 500 lx) takes Seat 5.
(3) Worker B moves to Seat 6.
(4) Worker C (target illuminance: 700 lx) takes Seat 8.
(5) Worker A leaves his/her seat.
(6) Worker B leaves his/her seat.
(7) Worker C leaves his/her seat.

6.2 Experiment Result

Fig. 6 shows the history of illuminance at each seat under the proposed Beacon-type Intelligent Lighting System.

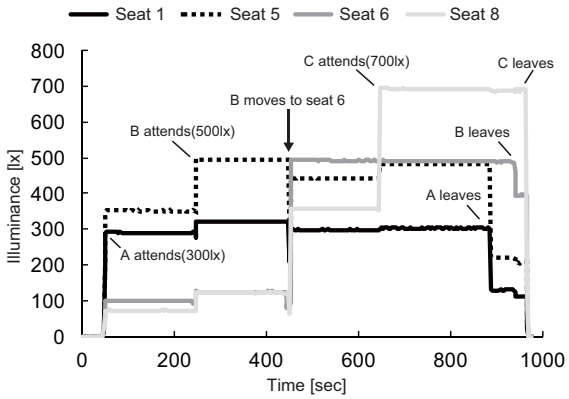


Fig. 6: History of Illuminance at Each Seat

Based on Fig. 6, it is confirmed that, by using the proposed method, the system detected whether each worker is seated or not and changed the luminance of lighting fixtures to satisfy the target illuminance. For each worker, illuminance converged to the target illuminance as soon as it is detected whether the worker is seated or not, and the system controlled luminance to maintain the target illuminance thereafter. Then 250 seconds after the start of the experiment, Worker B, in addition to Worker A, took a seat. It was confirmed that the system realized the target illuminance for Worker B while satisfying the target illuminance required by Worker A being satisfied; 200 seconds afterward, Worker B moved to Seat 6, and the system provided the target illuminance required by Worker B at Seat 6 while satisfying the target illuminance required by Worker A. It was confirmed that the system worked to satisfy the target illuminance for each worker when Worker C took a seat subsequently. The system also correctly detected that each of Workers A, B, and C left their seat and precisely dimmed or turned off lighting fixture. The graph in Fig. 6 shows that illuminance at a seat decreased when it was vacated.

Based on the result of this experiment, it can be said that the system can correctly manage the occupancy status of each worker's seat by detecting whether they are seated or not by using a beacon and smartphone and hence that a Beacon-type Intelligent Lighting System is useful.

7. Conclusion

This paper described a Beacon-type Intelligent Lighting System, which automates the toggling of the occupancy status of each worker's seat by incorporating a seat occupancy detection system using a beacon into the current Intelligent Lighting System. In order to enable the system to detect whether a worker is seated or not correctly using a beacon and a smartphone, RSSI was measured as the distance between them was changed by 0.1 m to set the threshold for determining whether a worker is seated or not. Then the operation of the system was verified by simulating an actual office. As a result of the verification, it was confirmed that the Beacon-type Intelligent Lighting System operates properly. Although a beacon is now placed on a desk in an experiment, it is considered important to examine and experiment a variety of ways to place a beacon and consider introduction into a real office.

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] 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, **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)

3D SOM neighborhood algorithm

Hongsong Li, Fulin Cheng, Yanhua Wang,Xinyu Ai
School of information and communication, Gui university of electronic technology
Guilin,Guangxi,China

Abstract: Neighborhood algorithm is an important part of 3D SOM algorithm. We proposed three kinds of neighborhood shape and two kinds of neighborhood decay function for 3D SOM algorithm and applied them in three-dimensional image compression coding. Experimental results show that the 3D orthogonal cross neighborhood shape and exponential function algorithm has better peak signal to noise ratio (PSNR) and subject quality.

Key words: self-organizing maps；three-dimensional image coding; pattern recognition; neighborhood algorithm

1. INTRODUCTION

The self-organizing map (SOM) proposed by Kohonen is a highly intelligent information processing method that forms a topologically ordered mapping from the high-dimensional data space to a low-dimensional representation space[1]. It has already found wide applications in such areas as pattern recognition[2-4], data mining[5-8], automatic control[9-10] and failure detection[11-12] and so on.

Traditional SOM network can effectively deal with one-dimensional and two-dimensional signal, but can not deal with three-dimensional signal directly. To solve this problem, three-dimensional self-organizing feature maps (3D SOM) algorithm has been proposed in recent years. In 3D SOM theory, the neighborhood is one of the core concepts of self-organization mechanism, which reflects the interaction and mutual influence between the winning neuron and the adjacent neurons, and has important affected on the quality of learning and convergence performance of SOM algorithm. For this reason, in order to improve the performance of 3D SOM algorithm, we need to find new three-dimensional neighborhood shape and neighborhood decay function. In this paper, we propose three kinds of neighborhood shapes and two kinds of neighborhood decay functions, and compare the performance difference.

2. 3D SOM neighborhood algorithm

2.1 3D Neighborhood Shape

The neighborhood shape of 3D SOM is a very important factor for the performance of the pattern library. We propose three kinds of neighborhood shapes, cube neighborhood, sphere neighborhood, 3D orthogonal cross neighborhood. Fig.1 (a) show the cube neighborhood, the length of cube side is twice the radius of neighborhood. Fig.1 (b) show the sphere neighborhood, the radius of sphere is the radius of neighborhood. Fig.1 (c) show the 3D orthogonal cross neighborhood, the length in each direction is the radius of neighborhood. The performance of 3D SOM algorithm is different when choosing different neighborhood shape.

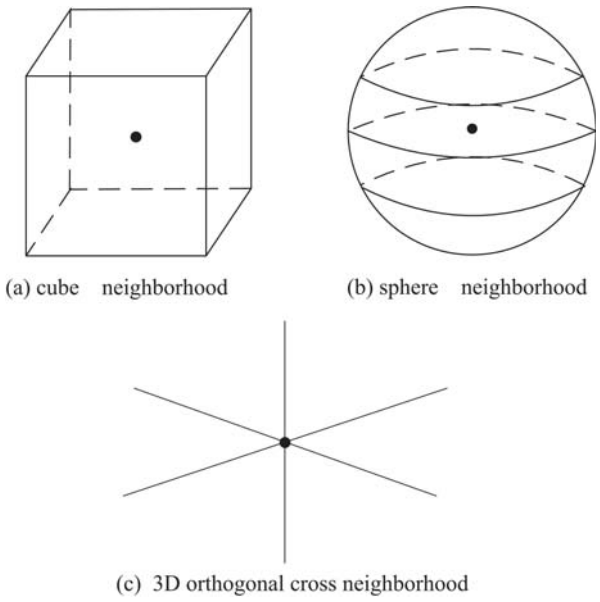


Fig.1 Three kinds of 3D neighborhood shape

2.2 3D Neighborhood Decay Function

Neighborhood function $NE(t)$ decreases with time t. When t is sufficiently large, $NE(t)$ tends to a constant, the SOM network tends to stability. This paper proposes the linear and exponential functions as neighborhood decay function.

(1) Linear function

$$NE(\mathrm{t}) = NE_{\min} + (NE_{\max} - NE_{\min}) * (T - t) / T \qquad (9)$$

(2) Exponential function

$$NE(t) = NE_{\min} + NE_{\max} \mathrm{e}^{-\mathrm{t}/T} \qquad (10)$$

$NE_{\max}$ and $NE_{\min}$ are constants determining the range of the neighborhood, and T is is a constant determining the decreasing rate. t is the number of iterations. When training begins, $t = 0$, $NE = NE_{\max}$, as the training progresses, t increases constantly, NE tend to $NE_{\min}$.

3. EXPERIMENTAL RESULTS

We use two testing grayscale stereo images to test our 3D SOM neighborhood algorithm, Tree and Plant. 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

$C_{\mathrm{R}} = \frac{M \times B_{\mathrm{O}}}{B_{\mathrm{C}}}$, where M is the dimensions of the pattern vector, B_{O} is bits of the original image pixels, B_{C} is bits of pattern vector class indexes.

In the experiment, the pattern library size is 2048 (compression ratio $C_{\mathrm{R}} = \frac{64 \times 8}{11} = 46.5$). we use 3D SOM algorithm with different neighborhood shape or neighborhood decay function to compress the disparity of the stereo image, and compare the prediction performance using PSNR between the original image and predicted image. Tab.1 shows PSNR comparison of three neighborhood shapes. It is clear that 3D orthogonal cross neighborhood achieve the best performance. Tab.2 shows PSNR comparison of two neighborhood decay functions. It is clear that exponential function has better than linear function.

Tab.1 PSNR comparison of three neighborhood shapes (dB)

Neighborhood shape	Cube	Spherical	3D orthogonal cross
Tree	31.61	30.43	33.39
Plant	30.43	28.93	32.69

Tab.2 PSNR comparison of two neighborhood decay functions (dB)

Neighborhood decay function	Linear function	Exponential function
Tree	29.55	33.39
Plant	27.63	32.69

REFERENCES:

[1]Kohonen T.Essentials of the self-organizing map[J].Neural Networks,2013,37:52-65.

[2]Hikawa H, Kaida K. Novel FPGA Implementation of Hand Sign Recognition System With SOM–Hebb Classifier[J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2015, 25(1): 153-166.

[3]Ben Haj Ayech M, Amiri H. Content-based image retrieval in the topic space using SOM and LDA[C]//Control, Engineering & Information Technology (CEIT), 2015 3rd International Conference on. IEEE, 2015: 1-6.

[4]Kuremoto T, Kobayashi K, Otani T, et al. One dimensional ring type growing SOM with asymmetric neighborhood function and its application to a hand shape instruction learning system[C]//Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2014 15th IEEE/ACIS International Conference on. IEEE, 2014: 1-6.

[5]Ma Z, Yang Y, Wang F, et al. The SOM Based Improved K-Means Clustering Collaborative Filtering Algorithm in TV Recommendation System[C]//Advanced Cloud and Big Data (CBD), 2014 Second International Conference on. IEEE, 2014: 288-295.

[6]Sanghvi Y, Gupta H, Doshi H, et al. Comparison of Self organizing maps and Sammon's

mapping on agricultural datasets for precision agriculture[C]//Innovations in Information, Embedded and Communication Systems (ICIIECS), 2015 International Conference on. IEEE, 2015: 1-5.

[7]Sapkota N, Karwowski W, Ahram T. Application of Evolving Self-Organizing Maps for Analysis of Human Adverse Events in the Context of Complex Socioeconomic Infrastructure Interactions[J]. Human-Machine Systems, IEEE Transactions on, 2015, 45:500-509.

[8]George A J, Gopakumar G, Pradhan M, et al. A self-organizing map-harmony search hybrid algorithm for clustering biological data[C]// Signal Processing, Informatics, Communication and Energy Systems (SPICES), 2015 IEEE International Conference onIEEE, 2015.

[9]Wang C H. Toward a New Task Assignment and Path Evolution (TAPE) for Missile Defense System (MDS) Using Intelligent Adaptive SOM with Recurrent Neural Networks (RNNs)[J]. IEEE Transactions on Cybernetics, 2015, 45:1134-1145.

[10]Honda S, Narita Y, Kida N. Adaptive control for vibration suppression by using self-organization map[C]//Control Conference (ASCC), 2015 10th Asian. IEEE, 2015: 1-5.

[11]Zhai X, Appiah K, Ehsan S, et al. A Method for Detecting Abnormal Program Behaviour on Embedded Devices[J]. IEEE Transactions on Information Forensics & Security, 2015:1-1.

[12]Minhoe Kim, Souhwan Jung, Minho Park. A Distributed Self-Organizing Map for DoS attack detection[C]// Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference onIEEE, 2015.

Proposal of load reduction method of the control computer in the cloud-type Intelligent Lighting System

Shinya DAINAKA[2], Mitsunori MIKI[1], Sota NAKAHARA[2], Katsuya ITO[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—*Authors are engaged in research and development of the Intelligent Lighting System, which makes improving the comfort of workers at offices compatible with reducing power consumption. As the Intelligent Lighting System was found to be effective as a result of demonstration tests, it is expected to be introduced into large-scale offices. We are thus examining a cloud-type Intelligent Lighting System which controls multiple areas by a single control computer. This study thus verifies the impact of an increase in controlled areas under a cloud-type Intelligent Lighting System on the control computer. It also proposes a method to reduce a load on the control computer.*

Keywords: Office, Intelligent Lighting System, Cloud, 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].

Since the control method used in the current demonstration tests requires one control computer per area, as the number of areas increases, the number of control computers increases. Therefore, it had an issue that the initial cost and the cost for maintenance and management increase accordingly. As a solution to this issue, we are examining a cloud-type Intelligent Lighting System which controls multiple areas by a single control computer. No verification has been made, however, on a load on the control computer when it controls multiple areas by using the control algorithm of the current Intelligent Lighting System. This study thus verifies the impact of an increase in controlled areas under a cloud-type Intelligent Lighting System on the control computer. It then proposes a method to reduce a load on the control computer of a cloud-type Intelligent Lighting System in a large-scale environment on the basis of the verification result.

2. 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.

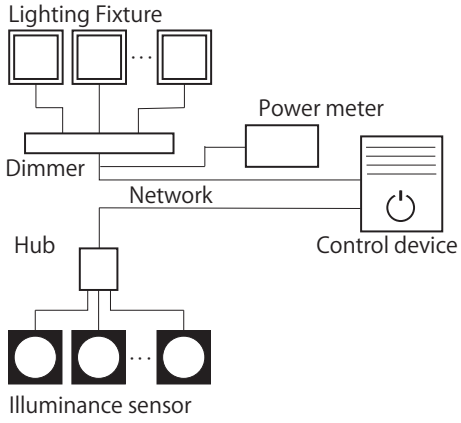


Fig. 1: Configuration of Intelligent Lighting System

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.

Authors have conducted demonstration tests using various control algorithms of the Intelligent Lighting System. Control algorithms of the System can be roughly classified into two types.

The first type of algorithms are a ANA/RC (Adaptive Neighborhood Algorithm using Regression Coefficient) 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 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 second type of algorithms are lighting control algorithms that use a method which can quickly realize the target illuminance for each worker (hereinafter referred to as the "simulation- based method") [7]. Since desks are infrequently moved in an ordinary office, it is possible to measure an illuminance/luminance effect in advance in an environment into which the Intelligent Lighting System is to be introduced by turning on and off its lighting fixtures one by one. By using that illuminance/luminance influence factor, a precise simulation environment can be constructed on the control computer. As a lighting pattern that satisfies illuminance required by each worker is rapidly derived in a simulation environment and applied to a real environment, the target illuminance for each worker is realized quickly.

We have measured the illuminance/luminance influence factor before introducing the Intelligent Lighting System. For this reason, we use the simulation-based method, which can quickly realize the target illuminance for each worker, as a control algorithm in this study.

As the largest-scale office building environment, authors assume one composed of several floors, each of which has 1,000 lamps and 500 illuminance sensors that are separated into multiple areas. An area here refers to a partitioned space or room.

Under the Intelligent Lighting System currently used in demonstration tests, only one area is controlled by a single control computer. For this reason, introducing the Intelligent Lighting System in the present configuration into a large-scale office results in an increase in the initial introduction cost and the cost for operation and maintenance. It is thus intended to solve these problems by using a cloud-type Intelligent Lighting System which controls multiple areas by a single control computer.

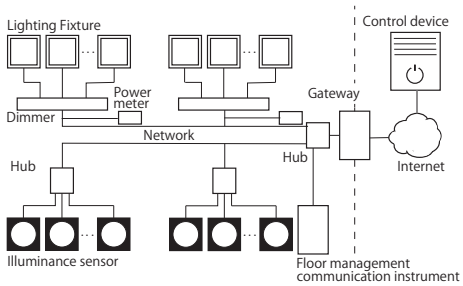


Fig. 2: Figure of constitution of the Cloud-Type Intelligent Lighting System

3. Cloud-Type Intelligent Lighting System

3.1 Overview of the Cloud-Type Intelligent Lighting System

A cloud-type Intelligent Lighting System is a system that controls multiple areas by one control computer. Fig.2 shows the configuration of the cloud type intellectual lighting system.

Floor management communication instrument in Fig.2 is a device that obtains illuminance values from illuminance sensors and performs the light control of lamps upon request by the control computer. This floor management communication instrument is placed on each floor. The control computer in Fig.2 obtains illuminance values from illuminance sensors installed in each area from the floor management communication instrument and execute processing for determining luminance for each lamp to satisfy the target illuminance set by each worker. This processing is called the next luminance determination processing and starts up a process that determines the next luminance for each area. After the next luminance is determined, the next luminance for each lamp is transmitted to a floor management communication instrument. A floor management communication instrument receiving the next luminance dims lamps. The interval between the reception of illuminance values from a floor management communication instrument by the control computer and the transmission of the next luminance to the instrument is set to 60 seconds as many areas are controlled simultaneously.

3.2 Control of Cloud-Type Intelligent Lighting System

It show a flow of the control in the control computer of the cloud type intellectual lighting system.

1) Set initialization parameters (initial luminance, target illuminance, illuminance/luminance influence factor, etc.).

2) Transmit luminance data for each lamp to floor management communication instruments to turn on each lamp.
3) Request a floor management communication instrument to obtain illuminance values to obtain an illuminance value from each illuminance sensor.
4) Calculate the value of the objective function at the current luminance.
5) Determine an appropriate range (neighborhood) within which the next luminance is generated in accordance with the regression coefficient of each lamp.
6) Randomly generate the next luminance within the neighborhood determined in (5) for each lamp to determine the next luminance.
7) Estimate the illuminance value of each illuminance sensor by using the illuminance/luminance influence factor.
8) Calculate the value of the objective function at the luminance determined in (6).
9) If the value of the objective function calculated in (8) is worse than that calculated in (4), reject the next luminance and revert to the previous luminance.
10) Repeat steps (4) through (9) for the specified times and send the next luminance at the end of the repetition to the floor management communication instrument and turn on lamps.
11) Wait for a specified period of time and return to (3).

In order to curb the maximum CPU utilization rate, steps (3) through (10) are executed for each floor instead of controlling all floors simultaneously.

3.3 Study parameter of Control of Cloud-Type Intelligent Lighting System

The current cloud-type Intelligent Lighting System starts a process for determining the next luminance severally for each area. Therefore, if the number of areas subject to control increases, the number of processes increases, which is considered to increase the load on the control computer. We thus verify the impact of an increase in the number of areas on the control computer.

4. Verification of a Load on the Control Computer Caused by an Increase in the Number of Areas

4.1 Verification environment

We verify a load on the control computer brought about by an increase in the number of areas in a cloud-type Intelligent Lighting System. As an area in the environment used for verification, we assumed the environment of Ecozzeria in Shin-Marunouchi Building, where a demonstration test using the simulation-based method was conducted. Fig.3 shows a figure of placement of the illumination sensor and lighting fixture of Ecozzeria.

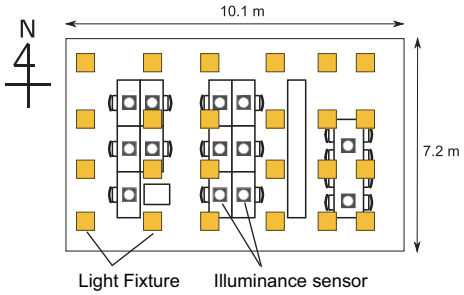


Fig. 3: Figure of placement of the illumination sensor and lighting fixture of Ecozzeria

Authors assumed the introduction of the system into an office building having several floors each of which has 40 areas identical to Ecozzeria shown in Fig. 3 (960 lamps and 520 illuminance sensors). Verification was carried out by increasing the number of floors until the maximum CPU utilization rate reached 100 % to make stable control no longer possible. Fig. 4 shows the assumed verification environment. Table.1 shows the details of control computer and floor management communication which I used by this inspection

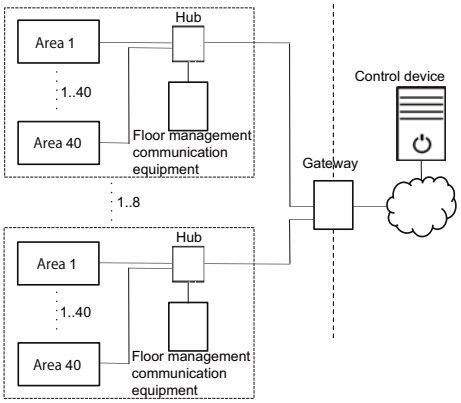


Fig. 4: Inspection environment

Table 1: Detail of the control computer

	Control computer	Floor management communication instrument
CPU	Intel Core 2 Duo (2.80 Ghz)	ARM1176JZF-S (770 Mhz)
Memory	2GB	512MB

4.2 Load inspection of the control computer and consideration

We verified an additional load on the control computer when the number of areas controlled is increased. Assuming a work day at a real office, control was performed for

12 hours in this verification. In addition, as the result of demonstration tests confirmed that the frequency of changes in target illuminance is small, it was assumed that the target illuminance is changed twice in one area. It is also assumed that workers come to the office in the first two hours of the simulation period. Table.2 shows an inspection result

Table 2: Load of the control computer

	one floor	two floor
maximum CPU utilization rate	85.9 %	100 %
maximum memory usage rate	66.8 %	92.1 %
maximum swap area usage rate	0 %	29.1 %

Table 2 shows the maximum CPU utilization rate, the maximum memory usage rate, and the maximum swap area usage rate. Table 2 shows that the maximum CPU utilization rate, the maximum memory usage rate, and the maximum swap area usage rate increase with an increase in the number of floors. When two floors were simultaneously controlled, the maximum CPU utilization rate reached 100 % and made stable control difficult. As the next luminance determination is processed severally for each floor, it does not raise the CPU utilization rate significantly. It is considered as the reason for a large increase in the CPU utilization rate that the maximum memory usage rate reached the limit to cause the use of the swap area, resulting in the use of CPU resources for swap processing (slashing). As a result of this verification, the number of stably controllable floor is up to one. This study thus proposes a method to reduce the memory usage rate and the CPU utilization rate of the control computer in order to increase the number of simultaneously stably controllable areas. For comparison with the proposed method, the control method used in this verification is called the standard method.

5. Composed of Control of Cloud-Type Intelligent Lighting System using the process suspension method and inspection

A cloud-type Intelligent Lighting System using the standard method intelligently controls lighting in each floor severally. In processing the next luminance determination for each floor, the process goes into sleep after completing the next luminance determination so that the light control interval may be 60 seconds. This chapter proposes a process suspension method which suspends the process while it is in a sleep mode to reduce a load on the control computer. This method can increase the number of simultaneously controllable areas by releasing memory allocated even during a sleep period after the completion of the next luminance determination processing under the standard method to reduce a memory load.

Assuming the environment same as the one assumed in Chapter 4, load verification was performed using the process suspension method. Fig. 5 shows the maximum CPU utilization rate, the maximum memory usage rate, and the maximum swap area usage rate when the method was used.

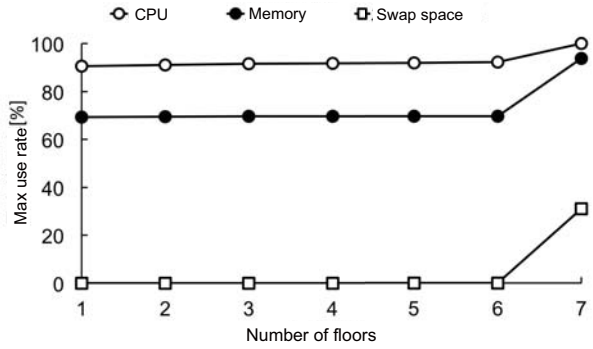


Fig. 5: Load of the control computer when using the process suspension method

Fig. 5 confirms that up to 6 floors can be stably controlled by using the process suspension method. Based on this result, it was confirmed that the use of the process suspension method is effective for reducing the memory usage rate. It was also confirmed that there was no large change in the maximum memory usage rate when the number of floor is from 1 to 6. This is because the next luminance determination is processed at staggered timing for each floor, as a result of which processes for floors which are not light controlled are suspended.

On the other hand, it was confirmed that the maximum memory usage rate increased when 7 floors are controlled. In this verification, the light controlling interval is set to 60 seconds. While it was possible to control 6 floors in 60 seconds, when 7 floors were controlled, the next luminance determination was simultaneously processed for 2 floors. For light control was performed severally for each floor in 60 seconds, which are not sufficient to cover 7 floors. This is considered to be the reason for the said increase in the maximum memory usage rate. Consequently, when 7 floors were controlled, the maximum CPU utilization rate reached 100 % to make stable control difficult. This is considered to be because the memory usage rate reached the limit to cause swap processing. It takes about 10 seconds after the control of 1 floor starts until the control of the next floor starts. Since the light controlling interval is 60 seconds in this verification, if 7 floors are controlled simultaneously, the control computer must start processes for 2 floors simultaneously. Therefore, the maximum memory usage rate increased significantly when 7 floors were controlled.

6. Composed of Control of Cloud-Type Intelligent Lighting System using the communication blocking method and inspection

A cloud-type Intelligent Lighting System using the standard method continues to control lighting intelligently after the illuminance value of every illuminance sensor in the area converges to the target illuminance in order to respond to external light from a window or a change in the target illuminance by a worker. External light, however, does not cause a sharp change in illuminance in a short time, and it has been found by multiple demonstration tests that workers seldom change the target illuminance. Therefore, it is considered to pose no problem if the intelligent lighting control is suspended during the time when there is neither any large change in illuminance values by external light nor a change in the occupancy status of a seat or the target illuminance by any worker.

Therefore, we propose a communication blocking method which suspend intelligent lighting control and block communication with the control computer after the illuminance of every illuminance sensor in the area converges. Under the communication blocking method, if a sharp change in illuminance values is indicated by an illuminance sensor in an area or if a worker changes the occupancy status of any seat or the target illuminance, a request to resume control is sent to the control computer to resume intelligent lighting control.

This methed reduces the memory load of the control computer. The thresholds for determining whether illuminance has converged or not are set at 3 % above and below the current illuminance taken as the standard. If illuminance values shown in the past 10 illuminance logs retrieved from each illuminance sensor are between those thresholds, illuminance is determined to have converged and communication is blocked.

Assuming the environment same as the one assumed in Chapter 4, load verification was performed using the process suspension method. Fig. 6 shows the maximum CPU utilization rate, the maximum memory usage rate, and the maximum swap area usage rate when the method was used.

Fig. 6 confirms that up to 5 floors can be stably controlled by using the communication blocking method. Stable control became difficult when 6 floors were controlled, which is considered to be caused in the following manner. As there were many areas simultaneously communicating with the control computer, the memory usage rate reached the limit to cause swap processing, which in turn caused the CPU utilization rate to reach the limit. Based on this result, it was found that, whereas only up to one floor can be stably controlled under the standard method, up to 5 floors can be stably controlled by using the communication blocking method, increasing the number of stably controllable floors by 4.

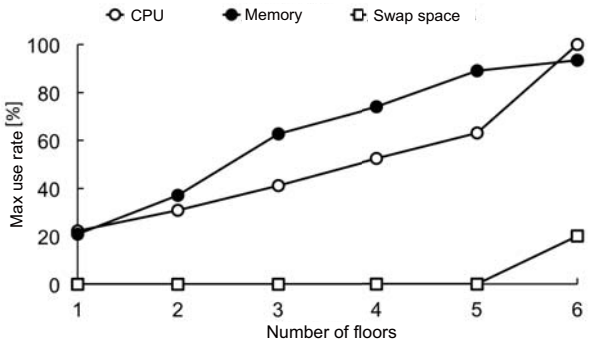


Fig. 6: Load of the control computer when using the communication blocking method

These results confirmed that the use of the communication blocking method is useful for reducing the load on the control computer.

7. Composed of Control of Cloud-Type Intelligent Lighting System using combined method and inspection

Methods proposed in Chapters 5 and 6 reduced a load on the control computer and increased the number of areas controllable with limited computational resources. It is considered possible to control more areas with limited computational resources by combining these two methods.

Assuming the environment same as the one assumed in Chapter 4, load verification was performed using the process suspension method. Fig. 7 shows the maximum CPU utilization rate, the maximum memory usage rate, and the maximum swap area usage rate when the method was used.

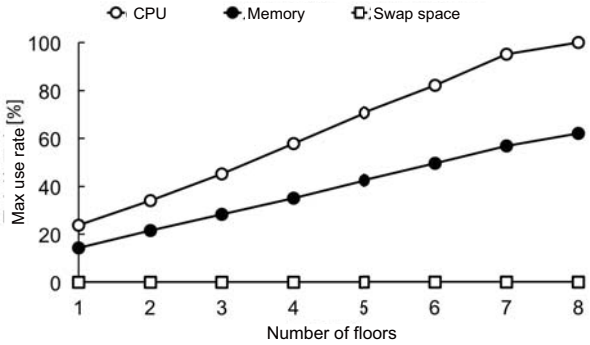


Fig. 7: Load of the control computer when using combined method

Fig. 7 confirms that up to 7 floors can be stably controlled by using the combined method. When 8 floors were

controlled, the CPU utilization rate reached the limit and made stable control difficult. The CPU utilization rate rose due to starting up processes and blocking or resuming communication. This is considered to be because, as the number of areas controlled increases, a CPU load becomes greater because of an increase in processing of the next luminance determination and process start-up.

These results confirmed that the use of the combined method is useful for reducing the load on the control computer.

8. Conclusion

This study thus verifies the impact of an increase in controlled areas under a cloud-type Intelligent Lighting System on the control computer. It also proposes a method to reduce a load on the control computer.

The process suspension method which suspends the process while it is in a sleep mode to reduce a load on the control computer. It was found that, whereas only up to one floor can be stably controlled under the standard method, up to 6 floors can be stably controlled by using the process suspension method, increasing the number of stably controllable floors by 5. The communication blocking method which suspend intelligent lighting control and block communication with the control computer after the illuminance of every illuminance sensor in the area converges. It was found that, whereas only up to one floor can be stably controlled under the standard method, up to 5 floors can be stably controlled by using the process suspension method, increasing the number of stably controllable floors by 4. Up to 7 floors can be stably controlled by using the combined method.

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 Wireless Networks(ICWN2014)

Intelligent Software Environment for Integrated Expert Systems Development

Galina V. Rybina[1], **Yuri M. Blokhin**[1]
[1]Department of Cybernetics, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow, Russian Federation

Abstract - *Development of the integrated expert systems with the task-oriented methodology and the problems of the intellectualization of AT-TECHNOLOGY workbench are reviewed. Intelligent planning methods applied for integrated expert systems architecture model generating are described with usage of the intelligent planner, reusable components, typical design procedures, and other intelligent program environment components.*

Keywords: integrated expert systems, problem-oriented methodology, AT-TECHNOLOGY workbench, intelligent software environment model, automated planning, automated planner.

1 Introduction

Trends towards integration of research in different fields of artificial intelligence had most clearly manifested at the turn of the XX and the XXI centuries and made it necessary to combine semantically different objects, models, methodologies, concepts and technologies. It created new classes of problems and new architectures of intelligent systems. The systems based on knowledge or expert systems (ES) (they were originally intended to support the decision formalized problems (UF-problems)), are an essential part of a significant number of static and dynamic intelligent systems, as the analysis of development experience [1-6] has shown. At the same time, in conjunction with ES, methods of applied mathematics, soft computing and versatile means of control data are used for solving tasks with practical complexity and importance. It leads to the necessity of combining ES into a single architecture with such diverse components as: a database comprising engineering, production and management information; software packages with the developed calculating, modeling and graphics tools; educating systems, including built on technology training and learning systems; simulation modeling systems for dynamic applications, etc.

A new class of intelligent systems emerged - integrated expert systems (ES). Their scalable architecture is represented in the form of ES + K. K - some software that provides decision support for the formalized part of the problem, and function of finding solutions to the UF-tasks relevant to the components of the poorly structured problem falls on the ES.

To address IES as an independent object of research, the IES terminological basis was formed and the basic definition of the IES as a systematically organized set of components ES and K was introduced. Their structure and nature of the interaction determine the type of the IES relevant and solvable problems - its functionality. The concept of "static IES " and "dynamic IES" was also introduced [1]. Automated task-oriented methodology [1] was created for the construction of IES, with powerful functionality and scalable architecture. It is actively used and constantly develops. The core idea is based on the conceptual modeling of the IES architecture at all levels of the integration processes in the IES and focusing on the modeling specific types of UF-tasks that are relevant to the technology of traditional ES. In the laboratory "Intelligent Systems and Technologies" Department of Cybernetics National Research Nuclear University MEPhI was created several generations of software tools such as Workbench (AT-TECHNOLOGY workbench) for the automated support for task-oriented methodology [1-4,7].

A large part of the problems is linked to the high complexity phases of design and implementation of the IES, as showed by practical experience of creating a series of static, dynamic, and educating IES through the use of task-oriented methodology and AT-TECHNOLOGY workbench [1-4,8]. Specifics of a particular area of concern and the human factor provide a significant impact. Therefore, the need to develop intelligent software environment and its basic component - intelligent planner [1,2,9] for the further development of task-oriented methodology and AT-TECHNOLOGY workbench with the aim of creating intelligent technology to build specific classes of IES has become urgent.

To date, multiple versions of intelligent planners for the AT-TECHNOLOGY workbench were created. They were developed by combining predictive models and methods of planning methods used in IES [1,2,9-14].

This paper point of issue is a new research phase and obtained results relating to the further development of intelligent planner and other components of the intellectual environment. The purpose for the new research is to increase the degree of automatic (intellectualization) planning and project management for the creation of a broad class of IES.

2 Some aspects of current research in the field of intelligent planning

Today automatic generation of plans by a software and hardware system is often meant as the intelligent planning, but the term “intelligent planning”» has no clear definition. For example, the term “intelligent planning” is more commonly used in Russian literature [15], while “automated planning” is used in English [16]. However, in both cases the plan generation process done by the computer is meant. Accordingly, plan is a glimpse of future behavior in the context of intelligent planning. In particular, the plan is usually a set of steps with some restrictions (e.g., temporal) for the execution of some agent or agents. [16].

Software system that generates plans based on a formal description of the environment, the initial state of the environment and assigned to the planner purpose is meant under planner in practical application. In some works, planner is called the agent [15]; in other cases, the planner also contains the designer carrying out the allocation of resources for the implementation of building plans.

A significant number of methods, approaches, formalisms, etc. was developed by now in the field of intelligent planning. Among them should be highlighted: planning with propositional logic; planning in a space plans; planning in space of conditions; planning as constraint satisfaction problems; planning on the basis of precedents; broadcast to the other problem; temporal planning; planning in non-deterministic and probabilistic areas; hierarchical planning (HTN-formalism), and others. Detailed reviews can be found in [15-17], and others, as well as in the works of authors [11-14].

Methods of the intelligent planning are widely used in a number of applications. The most popular fields of intelligent planning applications are: management of autonomous robots [16,18]; semantic web and web services composition [19]; computer-aided learning (in particular for the construction of individual education plans [10,20]); calibration of equipment [21]; control of conveyor machines [22]; resource-scheduling [23]; resource allocation in computing systems [24] logistics [25], the automation of software development [26] and others.

PDDL (Planning Domain Definition Language) developed under the McDemort’s leadership is the current norm in the field of the intelligent planning with the three main versions. We can detect the following trends in the context of development and use of various intelligent planning formalisms and related languages: the unification and standardization of planning languages on the basis of PDDL; the emergence of task-specific planning languages; a shift in emphasis towards nondeterministic research planning (RDDL languages and PPDDL); the emergence of complex mechanisms to meet the preferences and limitations, and others.

However, in general, application of intelligent planning for the automated support processes of building intelligent systems is a poorly investigated area, and it is possible here to refer mainly to the experience gained in the creation of applied IES based on the task-oriented methodology and AT-TECHNOLOGY workbench, in particular , the development and use of educational and dynamic IES [10,28,29]. Let us consider the basic concepts of intellectual AT-TECHNOLOGY workbench software in more detail.

3 Model of intellectual software environment and its components

Significant place in the framework of the task-oriented IES constructing methodology (basic points are reflected in [1]) is given to the methods and means of intelligent software support for the development processes. It is general concept of "intellectual environment". Complete formal description of the intellectual environment model and methods of the individual components implementation is presented in [1], so here only a brief description of the model in the form of quaternion is presented.

$$M_{AT}=<KB, K, P, TI> \quad (1)$$

KB is a technological knowledge base (KB) on the composition of the project, and typical design solutions used in development of IES. $K=\{K_i\}$, $i=1..m$ - set of current contexts K_i, consisting of a set of objects from the *KB*, editing or implementing on the current control step. *P* – a special program - an intelligent planner that manages the development and IES testing process. $TI = \{TI_i\}$, $i = 1..n$ - many tools TI_i, applied at various stages of IES development.

A component of the KB is a declarative basis of intellectual support for the development of IES, acting as data storage in a given environment and defined as

$$KB=<WKB, CKB, PKB>. \quad (2)$$

WKB is a KB containing knowledge of the standard design procedures (SDP), describing the sequences and methods of using various tools to create applied IES and a sequence of steps for creating IES. *CKB* - is KB comprising knowledge about the use of SDP and re-used components (RUC), including fragments of previously created IES prototypes. *PKB* (optional) - is a KB containing specific knowledge used at various stages of creating IES prototype for solving problems that require innovative approaches.

The current context K_i is represented as set of $K_i = <KD, KP>$. *KD* here is a declarative context for storing static declarative information about the structure of the project, the knowledge engineer and the current user. *KP* is a procedural

context, which includes objects clearly affecting the further planner steps (LC system phase, currently edited or executable object, the current target, the current executor, the global development plan, etc.).

The main procedural (operational) component is intelligent planner. This model generally describes it.

$$P = <SK, AF, Pa, Pb, I, GP> \quad (3)$$

SK here is the state of the current context, in which the scheduler was activated. $AF = \{AF_i\}$, $i = 1..k$ is a set of functional modules AF_i, a part of planner. *Pa* is a selection procedure for the current target based on the global development plan. *Pb* is a selection procedure for the best executive function module from the list of possible candidates. *I* - procedures to ensure the interface with the corresponding components of the AT-TECHNOLOGY workbench; *GP* - operating procedures for the IES global development plan.

Any SDP can be represented as triples

$$SDP_i = <C, L, T> \quad (4)$$

, where *C* - is the set of conditions under which the SDP can be implemented; *L* - script implementation described in the describing internal language actions of the SDP; *T* - set of parameters initialized by intelligent planner at SDP inclusion in the development plan of a IES prototype. Each RUC, involved in the development of an IES prototype, is represented by the tuple:

$$RUC = <N, Arg, F, PINT, FN> \quad (5)$$

N in this model is the name of the component, by which it is registered in the complex. $Arg = \{Arg_i\}$, $i = 1..l$ - set of arguments containing current project database subtree serving the input parameters for the functions from the set. $F = \{F_i\}$, $i = 1..s$ - a variety of methods (RUC interfaces) for this component at the implementation level. *PINT* - a set of other kinds of RUC interfaces, used by the methods of the RUC. $FN = \{FN_i\}$, $i = 1..v$ - set of functions names performed by this RUC.

Prototypes IES model of the functioning process is represented as triples:

$$M_A = <Sc, C, Cl>. \quad (6)$$

Sc is a scenario of IES prototype work. *C* - set of IES prototype subsystems that can be divided into 2 categories (standard subsystem (RUC) from the RUC repository and prototype's IES subsystem implemented by developers). *Cl* is a ratio of "control transfer ", describing the procedure for interaction between the IES prototype's subsystems.

Let us briefly examine the methods and approaches used in the implementation of the intellectual supportive environment for the development of IES model. The main components of this IES are the technological KB on the composition of IES project, SDP and RUC, and the intelligent planner managing the process of plans construction and implementation for the development of IES prototypes. These are the main purposes why it is necessary to use different types of knowledge in the process of developing a IES prototype: checking referential integrity of the project on the development of IES; automated construction of components diagrams; layout synthesis of IES prototype architecture; planning a series of steps to create a prototype of IES-specific features and tasks; determining a set of the most relevant sub-tasks for each of the stages (steps) in the development of IES prototype and others.

4 Development plan construction for the applied IES prototype

The task of creating a plan for developing an applied IES prototype is a complete task from the field of artificial intelligence, because it requires involvement of a variety of knowledge about models and methods for solving typical problems [1], a technology on design and development of IES, on how to integrate with external databases, packaged applications and other programs. Therefore, a project to develop IES-based task-oriented methodology and data on the problem being solved is stored in some format on physical carrier body of knowledge. Based on this data the intelligent planner is running the process of prototyping for applied IES.

It should be noted that the implementation of the current intelligent planner version [14] is a hybridization of approaches based on the use of HTN-formalism and flexible mechanisms to find solutions used in the IES. It allows the use of a declarative way of describing knowledge of the development, in this case the production-type knowledge representation language [1].

The implementation of managing the IES prototype development is a basic process, and requires the use of certain types of knowledge and use of intelligent planner. A brief description of some of the basic models for this process in accordance with [1] is given.

The model of the IES prototype is presented as a seven:

$$PRJ = <PN, KB, Solver, PD, PDFD, PPIK, PCOMP> \quad (7)$$

PN - project name, *KB* - IES prototypes KB, *Solver* - output machine (agents) for the IES prototype. *PD* - project data, i.e. the different types of information (knowledge, data, individual parameters, text, etc.), used by the intelligent planner in the process of developing a IES prototype and to generate the finished prototype, wherein the main project data include records of interviewing experts, lexical dictionary, fragments of the knowledge field, KB in the different

knowledge languages representation, the type of problem being solved, as well as various service information (the profile of the current user, the name of knowledge engineer who created the project, the date of commencement and the anticipated completion, etc.). *PDFD* - enhanced information and logical model of the IES prototype architecture presented as a hierarchy of advanced data flow diagram (ADFD), which is one of the most important components of the project because its structure largely determines the composition of the prototype and its functionality (hierarchy ADFD built using just a single relationship - decomposition, i.e. the upper levels operation is detailed using ADFD of the lower level hierarchy, and it's possible that only one ADFD that is not part of the operation exists, such as "contextual data flow diagram" - a set of PIC). $PCOMP = \{PCOMP_i\}, i = 1..l$ is the collection of various IES subsystems developed by means of AT-TECHNOLOGY workbench and external applications.

IES prototypes model development plan is presented by the cinque:

$$PL = <S, PP, A, R, P, D> \quad (8)$$

S is the set of IES prototype development stage, usually meaning life cycle (LC) stages of the IES prototyping. *PP* – a set of SDP. *A* - number of tasks whose implementation by the knowledge engineer is necessary for the successful development of a IES prototype; *R* - relation "part of the" between the elements of the A_i plurality from set *A*. *P* - consequence relation between the objectives. *D* - ratio that determines the feasibility of the task A_i from set *A* using a specific SDP from a *PP* variety. Hierarchical tasks network in the form of $TN = <A, R>$ can be defined for the *PL* plan using the HTN-formalism.

Plan of the IES prototyping can be decomposed into global and detailed. The global plan is a

$$PLG = \{AG, PG, CA\} \quad (9)$$

, where $AG \subset A$; $PG \subset P$, defined on the set of *AG*; *CA* - relationship "fit architecture models component", between the elements of the *AG* components plurality and the architecture of the IES prototype PDFD model, described as ADFD hierarchy. It should be noted that each component of the IES architecture model prototypes development is a complex process, and can be detailed as a complete specific tasks network. Detailed plan *PLD* is a global *PLG* plan detailed using HTN-based formalism, and is defined as $PLG = \{A, P, TN\}$.

IES prototype architecture model refers to the set of interconnected RUC and other subsystems that solve the problems described at the stage of user's system requirements analysis with the help of information and logical model.

Here is a general description of the IES prototype development plan synthesis problem. Assume *KU* - the set of all possible K contexts. Then the task of planning the IES prototype development can be represented as

$$PlanningTask = <K, AU, KG> \quad (10)$$

, *AU* is the set of all available actions of knowledge engineer on the PRJ project, *K* - is a *KU* element with full description, *KG* - is an element of *KU* and represents (perhaps not complete) description of the target context of the project. Ordered set $DevPlan = ap_1 ... ap_n$, where each $ap_i \in AU$, is the planning problem's solution, if superposition of operations $ap_n (... (ap_2 (ap_1 (K))) ...)$ eventually leads to a *KG* goal state. Previously described model of the *PL* plan is the *DevPlan* models development based on HTN-formalism and taking into account additional information.

Thus, the main task of intelligent planner is a dynamic support knowledge engineer operations at all life cycle stages of building prototypes as shown in Fig. 1. Dynamic support is done by generating IES development plans for the current IES prototypes and allowing the specific plans execution (made either automatically or interactively). It should be noted that detailed plans and global IES prototyping generation and architecture model synthesis is based on the integration of IES with planning methods.

Since the task of planning the IES prototype development usually implies an unambiguous outcome of the planned tasks implementation, determinate planning has been chosen as the main approach [15,16]. Of course, this approach does not explicitly take into account developments related to the risks and failures of the project due to the human factor, but this is offset by the restructuring plan strategies in the event of deviations from the plan. The environment for the scheduler is a description of the IES prototype project and entirely foreseeable.

Planning is performed in a state space of the project [16] (in the early studies the concept of "design space" was used). State space is formed by a plurality of project parameter's possible values. This approach was chosen because of its popularity and the large number of tools that can be effectively integrated into AT-TECHNOLOGY workbench intelligent planner.

Plan to build an IES prototype is a complete project development task, and it is a reason why elements of the plan have to be tied to the time. According to this [30] temporal approach with an explicit modeling of time was chosen, as it is planned to adapt the intelligent planner under the team development management in the further research. It is also another argument in favor of temporal planning. Described above formalism and language PDDL 2.1 is used as a basis for combining these approaches. It allows taking the time into account and using real variables.

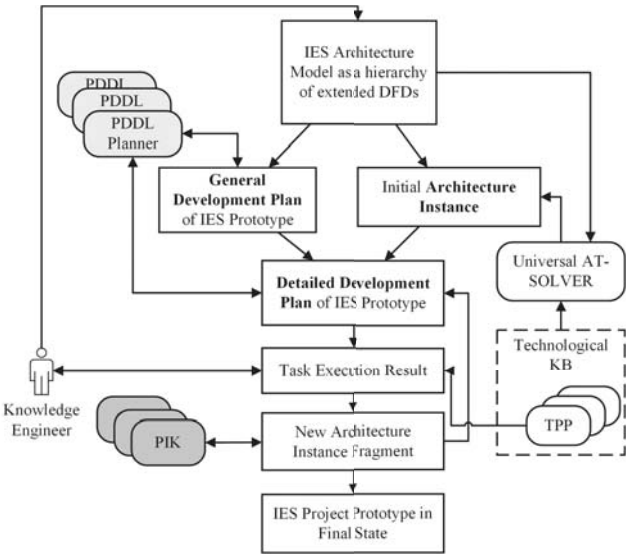


Fig. 1. The process of building IES prototypes based on the integration of planning with the IES methods

5 Features of model components realization for the intelligent software environment

Third generation of the AT-TECHNOLOGY workbench [1-4,7,14] is fairly complex modern software designed to solve problems associated with the design, development and maintenance of static and dynamic IES. The basis of the implementation of the workbench current version is approach based on technology using dynamic extensions (plugins), providing a modular and easy interchangeability of additional developer tools (knowledge engineer), implemented in the form of RUC complex.

Intelligent environment is implemented within the framework of the AT-TECHNOLOGY workbench architecture to support the development of IES in the form of several software modules (intelligent planner project manager, etc.) and data storage (project repository, the technological KB, etc.).

Software implementation of the intelligent planner current version is in the form of a few blocks. The main ones are: the architecture model synthesis component, plan visualizer, interaction with the knowledge engineer component, a component generation development plan for a prototype IES and others. These are implementation features of the most important components [12-14].

Plan visualizer. It is designed to visualize current plans and to initiate a specific task of a plan for the knowledge engineer.

Component generation development plan for a prototype IES. Provides a generation process for the generalized plans (based on the architecture IES model) and detailed plans (using the current IES architecture model and generalized plan). These processes are implemented using interaction with external PDDL-planners through component integration.

The PDDL-planners integration component. To solve the planning problem, intelligent planner uses external PDDL-planners, including LAMA, COLIN [31].

Methods of intellectual planning are used to solve described above tasks with intelligent scheduler planning. These methods are: deterministic classical planning in state-space model with the trivial time (to generate a global plan) and temporal planning with the action cost (to generate a detailed plan). These methods are implemented with the help of external programs - PDDL-planners, and the integration is based on file sharing. For PDDL-planners the task is created in the PDDL language version 2.1 [30] with additional enhancements that are further addressed in a separate process.

Separation to the global and detailed plan allows effective optimization to the intelligent planner operations, retaining its response time, as the number of states even for a small scheduling problem can be quite high [32]. The task of planning for PDDL-planner must be submitted in the form of quaternary.

$$P = <AP, IP, OP, GP> \quad (11)$$

In this model *AP* is a set of variables determining the state; *IP* - the initial state; *OP* - the set of operators; *GP* - target condition. Then converting of the planning tasks PlanningTask with intelligent planner is made as follows: a set of *AP* PDDL- predicates is formed by *PRJ*, *KU* and the ratio *R* (to determine the hierarchical relationship between tasks); set of true predicate and metric value *IP* is formed from *PRJ* and the current context *K*; from *AU* and *PRJ* a lot *OP* PDDL-operators (action) are generated; from the target context *KG* and *PRJ* a lot of truth for the target state PDDL-predicates are generated. It should be noted that the set of *OP* operators in the PDDL language is given implicitly through the use of typing and additional predicates.

The architecture model generation component. Produced generation of initial architecture model based on IES architecture model prototypes (represented as ADFD hierarchy), the generation of architectural model fragments by performing the objectives of the plan, as well as the merger of the new piece of architecture with the current layout. Generation of the initial model is done taking into account the specifics of the problem area that the IES prototype is developed for. That's why generic AT-SOLVER [1] is used to build the initial model architecture. It allows generating fragments of architecture model based on the subject area knowledge.

The configuration component. Configuration management is carried out, and it determines the choice of PDDL-planner, a directory for intermediate files, etc. Configurations are stored as XML-files, and component includes a graphic interface for editing configurations.

Interactive tasks in the development are transmitted to the means of AT-TECHNOLOGY interface, and the results are obtained from the interaction of a knowledge engineer. It is done with the help of the integration component and a variety of RUC. In addition, cooperation with RUC AT-TECHNOLOGY workbench by the transmit / receive messages is realized, and in order to highlight certain characteristics of the project and monitor the implementation of interactive objectives of the plan (by sending RUC messages) perform a number of internal project processing functions is performed to develop a prototype of the IES.

In the process of architecture model generation AT-SOLVER transmits instructions for specific kinds of tasks to the integration component with a variety of RUC. Specially developed language is used for a description of instructions. It is a subset of XML, because as a reporting format used by the components of the AT-TECHNOLOGY workbench, XML was selected.

Thus, the use of intelligent planning in managing the process of building IES prototypes based on ask-oriented methodology and AT-TECHNOLOGY workbench allows for the following functionality: the construction and implementation of the IES prototype's development plan at all stages of life cycle with intelligent planner; dynamic assistance to the knowledge engineer in the construction of current IES prototype based on the SDP and RUC knowledge; synthesis of IES prototype architecture and its components on the basis of enhanced information and logical IES architecture model; IES prototype analysis by using knowledge of the models and methods for solving typical problems; issuing recommendations and explanations to the knowledge engineer.

The amount of technological KB of the AT-TECHNOLOGY workbench currently stands at about 120 rules that are used to implement SDP at all life cycle stages of the IES development for problem areas associated with the diagnosis and design. In addition the development of IES prototypes uses about 70 RUC, of which 20 realize the capability of information RUC based on a single repository, and other 50 implement possible procedural RUC.

6 Conclusions

Currently, an experimental software study of the current version of the intelligent planner during educational IES prototyping on various courses is carried out. This study is carried out in particular, for the collective development of IES prototypes with limited resources. Research and development related to the use of intelligent software environment for building software applications for two dynamic IES prototypes are presented ("Management of medical forces and resources for major traffic accidents" and "Resource management for satellite communications system between regional centers").

The work was done with the Russian Foundation for Basic Research support (project № 15-01-04696).

7 References

[1] G.V. Rybina. Theory and technology of integrated expert systems construction. Monography. Moscow: Nauchtehlitizdat, 2008. – 482p.

[2] G.V. Rybina. Intelligent systems: from A to Z. Monography series in 3 books. Vol. 1. Knowledge-based systems. Integrated expert systems. Moscow: Nauchtehlitizdat. 2014. - 224 p.

[3] G.V. Rybina. Intelligent systems: from A to Z. Monography series in 3 books. Vol. 2. Intelligent dialogue systems. Dynamic intelligent systems. Moscow: Nauchtehlitizdat. 2015. - 160 p.

[4] G.V. Rybina. Intelligent systems: from A to Z. Monography series in 3 books. Vol. 3. Problem-oriented intelligent systems. Tools for intelligent system developing. Dynamic intelligent systems. Moscow: Nauchtehlitizdat. 2015. - 180 p.

[5] Grosan C., Abraham A. Intelligent Systems - A Modern Approach. Intelligent Systems Reference Library. Vol. 17, Springer 2011.- 450p.

[6] Expert systems research trends / A.R. Tyler (editor).- New York: Nova Science Publishers, 2007. - 238p.

[7] G.V. Rybina. AT-TECHNOLOGY workbench for integrated expert systems construction support: common description and developing prospects // Devices and Systems. Control, monitoring, diagnostics. 2011. №11, pp.17-40.

[8] G.V. Rybina. Practical use of problem-oriented methodology of integrated expert system construction (a review of applications in static and dynamic problem fields) // Devices and Systems. Control, monitoring, diagnostics. 2011. №12, pp.10-28.

[9] G.V. Rybina. Intelligent technology of integrated expert system construction implementation models and methods // Devices and Systems. Control, monitoring, diagnostics, 2011. №10: pp.27-37.

[10] G.V. Rybina, Y.M. Blokhin, Ivashenko M.G. Some aspects of intelligent technology for integrated expert system construction // Devices and Systems. Control, monitoring, diagnostics. 2013. №4. pp.27-36

[11] G.V. Rybina, Y.M. Blokhin, I.D. Danyakin. Models, Methods and Tools for Intelligent Program Environment for Integrated Expert Systems Construction // Information-measuring and Control Systems, 2014. №8, pp.34-40.

[12] G.V. Rybina, Y.M. Blokhin. Use of intelligent program environment for construction of integrated expert systems // Life Science Journal Volume 11, Issue SPEC. ISSUE 8, 2014, P.287-291.

[13] G.V. Rybina, Y.M. Blokhin, I.D. Danyakin. Intelligent tichnology for integrated expert system construction // Information-measuring and Control Systems, 2015. №1, pp.3-19.

[14] G.V. Rybina, Y.M. Blokhin. Modern automated planning methods and tools and their use for control of process of integrated expert systems construction // Artificial intelligence and decision making. 2015. №1. pp.75-93.

[15] G.S. Osipov. Artificial intelligence methods, Moscow.: FIZMATLIT, 2011. – 296 p.

[16] D.S. Nau. Current trends in automated planning // AI Magazine. 2007. Vol. 28. No. 4. P. 43–58.

[17] J. Rintanen. Introduction to Automated Planning (draft). – Germany, 2005.

[18] K.S. Yakovlev, E.S. Baskin. Graph models for solving 2D path finding problems // Artificial intelligence and decision making. 2013. №1. pp. 5–12.

[19] G. Zou, Y. Chen, Y. Xu et al. Towards automated choreographing of web services using planning // Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence. 2012. P. 178–184.

[20] A. Garrido, L. Morales, I. Serina. Using AI planning to enhance E-Learning processes // Pro- ceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012. 2012. P. 47–55.

[21] S. Parkinson, A. Longstaff, A. Crampton, P. Gregory. The application of automated planning to machine tool calibration // Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012. 2012. P. 216–224.

[22] M.B. Do, L. Lee, R. Zhou et al. Online planning to control a packaging infeed system // Proceedings of the Twenty-Third Conference on Innovative Applications of Artificial Intelligence. 2011. P. 1636–1641.

[23] V.N. Burkov, N.A. Korgin and D.A. Novikov, 2009. Theory of organizational system management introduction, Mosow. : Librokom, 2009. — 264 p.

[24] N.V. Kolesov, M.V. Tolmacheva and P.V. Yuhta.. Computation planning timecost and complexity assesment in distributed systems // Proceedings of 4th Russian multiconference, Taganrok: Taganrok: TTI YFU, 2011. pp. 312-314.

[25] A. Gerevini, P. Haslum, D. Long et al. Deterministic planning in the fifth international planning competition: PDDL3 and experimental evaluation of the planners // Artificial Intelligence 2009. Vol. 173, №5-6. P. 619–668.

[26] V. Novoseltsev, A. Pinzjin. Пинжин А. Implementation of efficient algorithm of linear functional programm synthesis // Bulletin of the Tomsk Polytechnic University. 2008. Vol. 312, №5. pp. 32–35.

[27] D. McDermott, M. Ghallab, A. Howe et al. PDDL - the planning domain definition language : Tech. Rep. TR-98-003 / Yale Center for Computational Vision and Control. 1998.

[28] G.V. Rybina, Y.M. Blokhin. Distributed knowledge acquisition control with use of the intelligent program environment of the AT-TECHNOLOGY workbench // Communications in Computer and Information Science, 2014, P. 150-159.

[29] G.V. Rybina, A.V. Mozgachev. The Use of Temporal Inferences in Dynamic Integrated Expert Systems // Scientific and Technical Information Processing, 2014, Vol. 41, No. 6, pp. 390–399.

[30] J. Benton, A.J. Coles, A. Coles. Temporal planning with preferences and time-dependent continuous costs // Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling (ICAPS 2012). 2012.

[31] S. Richter, Westphal, Helmert. LAMA 2008 and 2011. // Short paper for the International Planning Competition 2011. 2011.

[32] S. Russell, P. Norvig. Artificial Intelligence: A Modern Approach, Prentice Hall, 3rd Edition, 2009 — 1152 p.

Use of Artificial Neural Networks in the Production Control of Small Batch Production

P. Németh[1], **T.B. Ladinig**[1], **and B. Ferenczi**[2]
[1]Department of Logistics and Forwarding, Széchenyi István University, Győr, Hungary
[2]Rába Axle Ltd, Győr, Hungary

Abstract - *Our aim with this paper is to test a new performance measurement and control system for small batch production in the automotive industry with the help of Artificial Neural Networks. After the introduction of small batch production at an automotive company a possible use of this method for production control is presented.*

Keywords: Artificial Neural Networks; Logistics; Production control

1 Introduction

The foundation of a sustainable operations in manufacturing are stable and predictable production systems. Without adequate understanding of the factors – both internal and external – that affect those systems no business unit can operate in the long run. Studying and examining these factors alone is not enough though. During the research activities of the Department of Logistics and Forwarding the use of Artificial Intelligence emerged in the recent decade. Artificial Neural Networks in particular helped predicting and controlling logistics issues, such as:

- Warehouse management and performance monitoring
- Human resource effectiveness
- Prediction of defect product production

Our aim with this paper is to show the possibility of a new production control and performance management system based on the use of Artificial Neural Networks. First, we describe some earlier research of the Department than a possible solution will be presented for small batch production control.

2 Use of Artificial Neural Networks

During the last five years our Department used Artificial Neural Networks (ANN) for a series of logistics issues. Some of them will be detailed later. The reason for choosing ANN was the complexity of connections between logistics factors.

2.1 Artificial Neural Networks

Artificial Neural Networks are based on the arrangement and functional features of biological neural networks [4][7]. A neural network consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. This learning phase is important for the use of the model. With enough data from the past the ANN is able to learn the connection between the input and output data. Parallel with the learning phase a test phase is taking place where the network is testing itself with other data from the same problem.

As a result, after the learning the ANN can be used to predict output data from input data. This is the most important result of this method. With this, we are able to use the trained algorithm for modeling the problem and use it as a decision making tool.

Various different types of artificial neural networks are proposed in the literature [4][7]. In our application we use Multi-layer Perceptron (MLP), which is one of the most widely known types of ANNs. The topological structure of the MLP type neural network is illustrated in Figure 1.

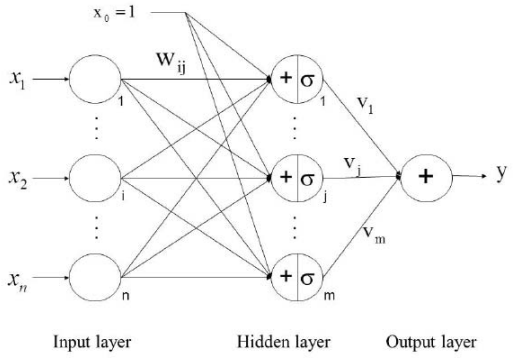


Figure 1: Topological structure of artificial neural network (MLP)

The network contains three layers. The input layer contains the inputs x_i. In the hidden layer m hidden neurons are applied. Between the input and hidden layers there are connection weights w_{ij} connecting the i-th input and j-th hidden neuron.

There is also a bias input $x_0 = 1$ with weight w_{0j} to the j-th hidden neuron. There is an output layer with output y. Between the hidden and output layers there are weights v_j connecting the j-th hidden neuron with the output.

The output of the neural network can be computed as:

$$y = \sum_{j=1}^{m} v_j \cdot \sigma\left(\sum_{i=0}^{n} x_i \cdot w_{ij}\right) \quad (1)$$

In (1) s is the sigmoid function:

$$\sigma(s) = \frac{1}{1 + e^{-K \cdot s}} \quad (2)$$

where K is the slope parameter of the sigmoid function.

2.2 Training Algorithm

Training or learning is the method of modifying the parameters (e.g. the weights) of the neural network in order to reach a desired goal. We can somehow classify learning with respect to the learning mechanism, with respect to when this modification takes place, and according to the manner how the adjustment takes place. In this paper the artificial neural network is trained in supervised, off-line manner by bacterial memetic algorithm [2].

Nature inspired evolutionary optimization algorithms are often suitable for global optimization of even non-linear, high-dimensional, multi-modal, and discontinuous problems. Bacterial Evolutionary Algorithm (BEA) [6] is one of these techniques. BEA uses two operators; the bacterial mutation and the gene transfer operation. These operators are based on the microbial evolution phenomenon. The bacterial mutation operation optimizes the chromosome of one bacterium; the gene transfer operation allows the transfer of information between the bacteria in the population.

Evolutionary algorithms are global searchers, however in most cases they give only a quasi-optimal solution to the problem, because their convergence speed is low. Local search approaches can give a more accurate solution; however, they are searching for the solution only in the neighborhood of the search space. Local search approaches might be useful in improving the performance of the basic evolutionary algorithm, which may find the global optimum with sufficient precision in this combined way. Combinations of evolutionary and local-search methods are usually referred to as memetic algorithms [5].

A new kind of memetic algorithm based on the bacterial approach is the bacterial memetic algorithm (BMA) proposed in [2]. The algorithm consists of four steps. First, a random initial population with N_{ind} individuals has to be created. Then, bacterial mutation, a local search and gene transfer are applied, until a stopping criterion (number of generations, N_{gen}) is fulfilled. BMA can be applied for training neural networks [1][3]. In this case the parameters to be optimized, which are encoded in the bacterium, are the w_{ij} and v_j weights of the neural network (see Fig. 1). The details of BMA can be found in [1][2][3].

3 Production process

The small volume production business unit of the automotive OEM has a diverse product mix and many machine set-ups for different production orders that take only 4-12 hours on average to complete. Quality issues arise caused by the many machine set-ups and various products, which results in unstable processes and high variability, as well as complexity. The business unit is an extended workbench and tier-1 supplier, which is focusing on increasing efficiency and productivity within a functional job-shop manufacturing layout with time-phased MRP planning and a multi-level dependent demand system.

The production system consists of three departments that contain all production processes from metal disc to final assembly of all the main products like doors, side panels, roofs, bonnets and hatches for premium small volume sports cars. The first department is the component production where metal components are pressed and laser cut out of aluminum or stainless steel discs that enter the facility pre-cut by a supplier.

The second department contains all the assembly processes where the final assemblies are put together out of two to four main components and several smaller purchased parts like reinforcement parts to increase the stability of the products. The third and final department is called “finish” and it is responsible to ensure proper quality in terms of surface quality (which can be impacted by difficult pressing processes) and dimension (which is mostly influenced by the different mating technologies used at the assembly department). At the end the final products are ready to be shipped to the main body shop and subsequently the paint shop of several internal customers of the production system. Furthermore, there is one more department within the small volume job-shop segment which is the project and process management department, responsible for launching new customer projects on time and to ensure continuous smooth operations of all running projects throughout their life cycle.

In addition to the small-series job shop the business unit contains four other, more or less separated segments which are: quality department, logistics, fixture construction and robotics (development and construction of the fixtures, manufacturing cells and robot programs for the assembly department and other customers), tool making (development and construction of pressing tools for the component production and other customers). The high degree of vertical integration makes it difficult to create an adequate value stream and material flow because processes with very different batch sizes and processing times must be tied together efficiently. Compared to normal production series the production system would be

divided into two separate segments, one being the pressing plant and the other would be the body shop where relatively high amounts of Work In Progress (WIP) inventory are quite common. Therefore, it is necessary to have large amounts of inventory to balance the differences in processing time and batch size. Also asset utilization for integration of new projects into existing production assets is quite common, so not all the available time can be used continuously for series production, but also periodical time frames must be planned as implementation time.

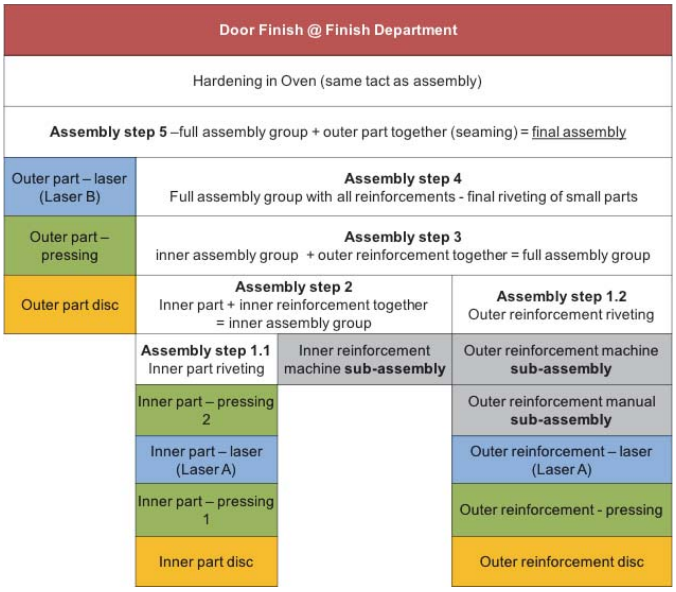


Figure 2: Example of the assembly steps based on the Bill of Materials

3.1 Need for production control

The above described production system – mainly because of the small batches – requires an adequate control and performance measurement system for the operation. The business unit is facing quality issues and continuous pressure from the customers for shorter lead time. With a tailor made control system the planning of the production would be able to serve the needs of the clients and enable long run effectiveness and sustainability of the business unit. In the next chapter a possible solution is described.

4 Proposed method

Our aim was to use ANN first for the prediction of the production planning accuracy. This accuracy is one of the most important factors in small batch manufacturing. If this accuracy is high enough the production can fulfill the orders in the given time window and can achieve competitive advantage in the long run.

As an input the following data will be used:

- Production quantity
- Production complexity (number of assembly steps)
- Capacity utilization

As an output we regarded the production planning accuracy. This setup can be seen in the following figure

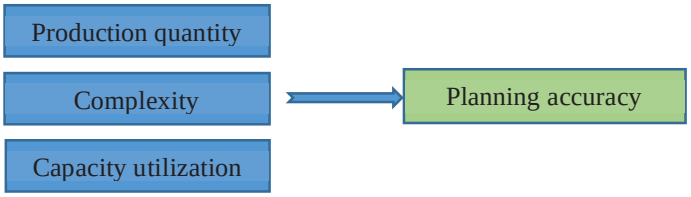


Figure 3: Input and output data of the model

With enough data of previous production shifts – together with the measured accuracy of production planning – we will be able to simulate this output value with changing the input data according to the production plan. This could be used as a decision making tool: By given number of orders (fix production quantity and complexity) the adjusting of capacity utilization – by adjusting maintenance time for example – can increase the accuracy of production planning. Or, the other way around, by given maintenance time or pre-defined capacity utilization the number of orders will give the predicted accuracy thus helping the production control with possible output of the current shift - even before production started.

Table 1: Example of collected data

Production quantity	Complexity	Capacity Utilization	Planning accuracy
200	4	0.90	0.95

We collected data for the testing of our method. With this data at hand we started the simulation with the following parameters:

4.1 Test of the method

With these collected data we performed 10 simulations using 6 hidden neurons and the mean relative error (MRE) of the train and test set was investigated which is defined as:

$$MRE = \frac{1}{p}\sum_{i=1}^{p}\left|\frac{y_i - t_i}{y_i}\right| \quad (3)$$

where y_i is the output of the network for the i-th pattern, t_i is the desired output for the i-th pattern, and p is the number of patterns. With the trained algorithm we performed further evaluation of the test results.

We created three scenarios for the testing of the algorithm. The best scenario was a large quantity order with few assembly steps (complexity) and average capacity utilization. According to our assumption these parameters will give the best output (production accuracy). The worst scenario is a low quantity order with complex assembly and high capacity utilization.

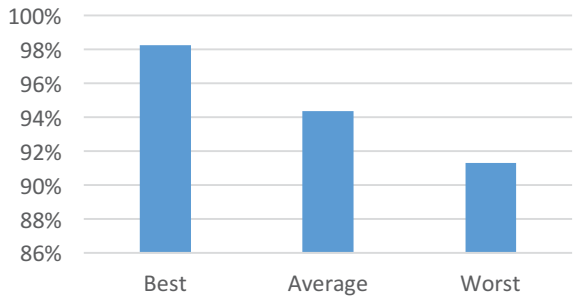


Figure 4: Planning accuracy based on the scenarios

5 Conclusion

This test showed the viability of our model. The accuracy of production planning given as a result of the trained algorithm follows our assumption: Bad scenario result in lower level of accuracy. Further data collection can enhance the weights and thus the algorithm. Long term aim of our research is to give the production control this tool for daily use in planning operations.

6 Acknowledgements

This work was supported by OTKA Fund (K-105529) of the Government of Hungary.

7 References

[1] Botzheim J, Földesi P, "Fuzzy neural network with novel computation of fuzzy exponent in the sigmoid functions." In: Proceedings of the 8th International Symposium on Management Engineering, ISME 2011, Taipei, Taiwan, pp. 285–291

[2] Botzheim J, Cabrita C, Kóczy LT, Ruano AE, "Fuzzy rule extraction by bacterial memetic algorithms." In: Proceedings of the 11th World Congress of International Fuzzy Systems Association, IFSA 2005, Beijing, China, pp. 1563–1568

[3] Gál L, Botzheim J, Kóczy LT, Ruano AE, "Applying bacterial memetic algorithm for training feedforward and fuzzy flip-flop based neural networks." In: Proceedings of the 2009 IFSA World Congress and 2009 EUSFLAT Conference, IFSA-EUSFLAT 2009, Lisbon, Portugal, pp. 1833–1838

[4] Hecht-Nielsen R, Neurocomputing. Addison-Wesley (1990)

[5] Moscato P, "On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms." Tech. Rep. Caltech Concurrent Computation Program, Report. 826, (1989) California Institute of Technology, Pasadena, California, USA

[6] Nawa NE, Furuhashi T: "Fuzzy system parameters discovery by bacterial evolutionary algorithm." IEEE Transactions on Fuzzy Systems 7(5) (1999), pp. 608–616

[7] Zurada JM, Introduction to Artificial Neural Systems, West Publishing Co., St. Paul, 1992

[8] Bhagwat R, Sharma MK, "Performance measurement of supply chain management: A balanced scorecard approach", Computers & Industrial Engineering 53(1), 2007 pp. 43–62

Can Neural Networks Help Diagnose?

Dr. Wayne E. Smith (contact author)
Graduate School of Business and Management
Argosy University
5001 LBJ Freeway, Suite 180
Dallas, Texas, 75244, USA
wesmith@argosy.edu

Dr. George L. Singleton
Graduate School of Business and Management
Argosy University
5001 LBJ Freeway, Suite 180
Dallas, Texas, 75244, USA
gsingleton@argosy.edu

Abstract - *The coupling of artificial intelligence technologies with advanced medical practice may improve the practice of medicine and deliver efficient results for patients and providers alike. Advancements in data collection, electronic records and neural networks synergistically advance the goal. Case studies on various types of cancer and other serious life-threatening diseases are available for artificially intelligent technologies to utilize in the furtherance of disease detection and treatment. Artificial intelligence is advancing on several fronts and the synergy of these advances is exciting for the medical community.*

Keywords: artificial intelligence; electronic records; colorectal cancer; diagnostics; treatment

Type: Short Research Paper

I. INTRODUCTION

The goal of the work presented in this paper is to offer possible methods to advance the study of artificial intelligence in medicine. The use of artificial intelligence in e-commerce was designed to facilitate transactions for buyers as well as sellers. It has been proposed that artificial intelligence may also facilitate disease fighting. Can neural networks assist physicians to diagnose? The synergistic possibilities offer advances in medicine around the world.

II. LITERATURE REVIEW

A. Early Concepts of Artificial Intelligence

Although the term artificial intelligence is little more than sixty years old, ancient Greek myths describe a giant man of bronze who guarded Europa from attackers by patrolling the island three times a day [1]. Peoples of Egypt joined Greeks in ancient times to worship cult images in human form. Further, Intelligent non-human beings were popular in Western fictional writings in the United States' nineteenth and twentieth centuries.

Innovations in ancient times by philosophers and mathematicians originated logical or mechanical reasoning. Greeks Pythagoras and Heraclitus developed concepts that ultimately formed the basis of programming computers. Meticulous logic led to mathematical solutions to complex problems. Hundreds of years later, during World War II, Alan Turing used logical, cryptology, and electrical talents to propose and prove that a mechanical and digital machine could make mathematical decisions using a binary numbering system consisting only of zero and one [1].

Expansion of artificial intelligence reached beyond myths, modern fiction, mathematical theories, and computer design. Guided by companies leading technological and social media applications, retail stores broadcast greetings from humanoids; manufacturers of smart phones build in voices clever enough to guide drivers across city and state routes; and robotic assistants record schedules and remind children and adults alike of appointments [4].

B. Advancements in Data Collection for Medicine

Electronic medical records (EMR) give doctors and scientists laboratory results, eliminating paper reports. EMRs require fewer resources and reduce waste. Analysis considered needs assessment for every report to determine elimination. After two years, 99% fewer monthly reports were printed – from 84,000 to 370. The initiative to combine data analysis and practitioners' needs was effective in lowering cost and time while maintaining accurate patient information [3].

C. Cancer Diagnostics

A deficiency in analyzing cancer with somatic mutation and sequencing techniques was identified. Even though long-standing, conventional order and report processes are effective for testing single genes, obstacles are numerous, and stimulating computers to develop artificial intelligence is crucial to complex genetic tests [9].

Complex tests require storing, tracking, and summarizing variants of sequences and correlating metabolic pathways. Thereafter, physicians need visual representations of profiles to pursue indicators for custom-designed patient treatments [9]. Conclusions dependent on integrating large volumes of clinical, molecular, social, and environmental data are not possible without efforts to increase artificial intelligence.

D. Diagnosis of Colorectal Cancer

Doctors cannot construct accurate expectations for groups of patients with symptoms of colorectal disease, and it is often unknown who has polyps, cancer, or colitis [5]. The common question for patients with colorectal symptoms

is does colorectal pathology give significant indicators to order a Lower Gastrointestinal Endoscopy (LGE)?

Clinical experience, judgment, and statistics from linear models are the foundation for predictable expectations. Assessments by clinicians are inaccurate and have an unacceptable rate of 20-40% of colonoscopies that are not necessary or not helpful in a large sample of cases. The protocol, to wait no longer than two weeks before ordering a LGE, was changed to improve the percentage. Selecting the time period was subjective and the revised protocol did not meet new goals of diagnosis. Specifically, reducing the time to order procedures produced only 14% accurate diagnosis of colorectal cancer. The lack of abnormal findings in most patients indicated that models for expert systems were deficient in sophisticated dependable artificial intelligence as diagnostic systems [5].

E. Artificial Intelligence and Pathology

Pathologists are occupied with interpreting slides and describing their interpretations in reports to colleagues, administrators, and government officials. Working to examine slides with blood and tissue samples is incomplete without documenting findings, possible findings, and lack of findings. Scientists continually try to find methods to lessen the administrative time to produce reports.

Efforts to develop artificial intelligence methods to complete clerical activities used a system designated as Secretary-Mimicking Artificial Intelligence (SMILE). As pathologists describe slides orally, systems developed by Sunquest Information Systems and Microsoft generate voice commands caused by pathologists and slide barcode scans. The composite of computer programs that make up SMILE responds to the commands and produces reports. SMILE also broadcasts comments to pathologists and exhibits comments on computer monitors; comments are helpful for pathologists making changes before reports are final [10].

Results of the artificial intelligence body of systems are favorable and constructive; SMILE allows pathologists to concentrate on analyzing slides with less effort to document reports. The system is fast and consistent, increasing pathologists' productivity, reducing errors, and deflecting stressful reporting tasks. Furthermore, SMILE learns continually from repetitive encounters; continuous improvement is an element of its design and function [10].

F. Case Study of Data Mining of Irritable Bowel Syndrome

A plan was developed to determine how quality of health and life deteriorates in the presence of irritable bowel syndrome (IBS). Conceiving of the study, they emphasized the dearth of predictive variables that can influence quality of life for patients with the ailment. Studying quality of life related to IBS meant expanding artificial intelligence with data mining techniques for a select group of patients. In fact, the researchers intended to examine and compare methods of data mining to determine the degree of correlation between demographic groups and IBS symptoms. Degrees of correlations could support or dispel associations of IBS and quality of life in certain groups [7].

A design was concieved consisting of a cross-sectional survey of the general population of the United Kingdom (UK). The methodology was to select people exhibiting IBS symptoms and subject them to a longitudinal cohort survey using a UK-wide newspaper alert for recruitment. They measured quality of life related to health with a succession of survey questions. Artificial intelligence in the form of data mining models resolved which factors associated with reduced health-related quality of life were relevant in the study; models comprised logistic regression, a classification tree, and artificial neural networks [7].

Results indicated that emotional melancholy and social status, such as married and employed, were significant factors in health-related quality of life for IBS patients. It was concluded that data mining was not so easy to interpret as logistic regression; however, data mining methods identified subsets of factors significantly related to IBS signs and emotional states of subjects. There was a strong statistical correlation between health-related quality of life in the UK. The hypothesis that quality of life was damaged for UK citizens with IBS was accepted. Further studies, to identify the effect of emotional and social status on health-related quality of life, might reveal more correlations with IBS. Such studies could magnify understanding emotional, social, and physical factors. Relevant studies could contribute to improving clinical practices to diagnose and treat the patient population [7].

G. Case Study of Nonsurgical Treatment

A study concluded that it is common for physicians to manage patients diagnosed with colorectal cancer without resecting the original tumor. Following discussion of specific cases, a multidisciplinary team (MDT) made the decision. Effectiveness of case management without resectional surgery to remove the primary tumor is questionable, especially because most patients have distant metastatic disease [8].

However, data for such a patient population indicate that 20–30% of patients with recent confirmation of colorectal cancer have synchronous distant metastatic disease. Subsequent to diagnosis, most patients are classified as incurable using surgical methods. The statistics were contraindicative notwithstanding evidence that the primary tumor appeared to be easily resectable. Arguments continue for and against resecting the primary tumor in patients thought to be incurable, although statistical evidence indicates benefits only for a small group. Nonetheless, the primary goal for treating patients with unresectable metastatic disease is palliation, focusing on controlling symptoms and maintaining quality of life [8].

H. Case History of Surgical Treatment

Not all elements of colorectal resection are advantageous, nor do they deliver superior results in treating patients [2]. For example, duration of an operation was

greater with procedures that involved robot assistance: 270 minutes compared to 138 minutes for right hemicolectomy; 200 versus 140 for sigmoid colectomy; and 180 over 165 for rectopexy. The information originated in a matched cohort study using a robot. The robot's function was only to reposition the colon. Further, a bowel grasper assisted the robot through a port created to provide access.

Robot-assisted surgeries evaluated in the study consisted of two right hemicolectomies, three sigmoid colectomies, and one rectopexy. Partial reports of outcomes contained information about a patient in the robotic population who contracted atelectasis following a right hemicolectomy. Another patient suffered a late incisional hernia after a sigmoid colectomy; the patient was in a group receiving laparoscopic treatment.

Another case involved a sigmoid colectomy changed after 50 minutes to laparoscopy due to a problem locating the ureter. The two groups, differentiated by robot assistance regardless of surgical procedure, experienced approximately the same amount of blood loss and similar incision lengths to remove specimens [2].

III. SYNERGY

Future of Artificial Intelligence in Medicine

Diagnosis is a prominent term in making decisions to treat patients. A combination of patient case history and symptoms combine to help physicians diagnose illness. Doctors' education, experience, memory, and association with colleagues are the foundation for interpreting medical history and signs of illness associated with disease. Designing a neural network increases the degree of accuracy in choosing treatments. A neural network imitates doctors' diagnostic thinking; the network depends on a knowledge base of information and practice cases to learn to diagnose diseases [6].

A type of artificial intelligence, neural networks employ multiple artificial neurons joined to function as information processors. Neural networks make medical diagnosis available online by data mining patient record repositories of history and selecting information to apply to current patients.

Artificial intelligence computer applications run on neural networks with the capacity to learn from historical data. The strength of neural networks is to create deductive results from data accumulated in networks' knowledge bases. Automation to manage facilities, equipment, finance, training, and communication is valuable in medicine; however, automation to diagnose patient conditions is perhaps most valuable. Learning makes artificial intelligence applications particularly useful in diagnosing illness. Neural networks that process information compiled with data mining are likely to enable medical professionals to give patients options to view medical advice online from artificial intelligence software [6].

IV. SUMMARY

The possibility of combining advanced data collection with electronic medical records and neural networks is promising. In a medical environment focused on cancer and other life-threatening disease, every advancement is needed to find cures. The application of artificial intelligence to this battle may give new hope to those in the fight.

V. References

[1] Del Monte, L. A. (2014). The beginning of artificial intelligence. In *Science questions and answers.* Retrieved January 15, 2015, from http://www.louisdelmonte.com/the-beginning-of-artificial-intelligence-part-12/

[2] Gutt, C. N., Oniu, T., Mehrabi, A., Kashfi, A., Schemmer, P., & Buchler, M. W. (2004). Robot-assisted abdominal surgery. *British journal of surgery, 91,* 1390-1397.

[3] Henricks, W. H., & Shirk, T. (2007, May). Eliminating paper laboratory reports in a complex medical center environment. *Advancing practice, instruction and innovation through informatics (APIII 2006), scientific session presentation abstracts and scientific poster session abstracts, Arch Pathol Lab Med – 131*.

[4] Kratochwill, L. (2015, December). The real deal on artificial intelligence. *Popular Science, 287*(1), 34.

[5] Maslekar, S., Gardiner, J. R., Monson, & Duthie, G. S. (2010). Artificial neural networks to predict presence of significant pathology in patients presenting to routine colorectal clinics. *The Association of Coloproctology of Great Britain and Ireland, 12,* 1254-1259.

[6] Nicks, V. (2011). Artificial intelligence for medical diagnosis. Retrieved January 20, 2015, from http://www.decodedscience.org/artificial-intelligence-for-medical-diagnosis/13

[7] Penny, K. I., & Smith, G. D. (2011). The use of data-mining to identify indicators of health-related quality of life in patients with irritable bowel syndrome. *Journal of clinical nursing, 21* 2761-2771.

[8] Turner, P. S., Burke, D., & Finan P. J. (2013). Nonresectional management of colorectal cancer: multidisciplinary factors that influence treatment strategy. *The association of coloproctology of Great Britain and Ireland, 15. e569-e575.*

[9] Wilkerson, M. L., Henricks, W. H., Castellani, W. J., Whitsitt, M. S., & Sinard, J. H. (2015). Management of laboratory data and information exchange in the electronic health record. *Arch Pathol Lab Med – 139,* 2015.

[10] Ye, J. J. (2015). Artificial intelligence for pathologists is not near – it is here. *Arch Pathol Lab Med – 139, 929-935.*

Task learning improves two-dimensional movement control in brain-computer interfacing using only two electrodes in a noninvasive, low-cost, brain-computer interface (BCI)

Jeffrey DeVince[1] and Arthur Ritter[1] (Contact Author), *Member, IEEE*
[1] Department of Biomedical Engineering, Chemistry, and Biological Sciences, Stevens Institute of Technology, Hoboken, NJ, 07030, USA,
Type of Submission: Regular Research Paper
Keywords: two dimensional movement control, (BCI), electroencephalography (EEG), noninvasive, motor imagery, Event related spectral perturbation (ERSP),

Abstract—BCIs are used for control applications in severe motor disabilities. Electroencephalogram (EEG) based BCIs are promising because they are non-invasive. Presently, expensive acquisition systems are used having a large number of electrodes. We demonstrate real-time, two-dimensional movement control using a noninvasive, low-cost ($1,000) BCI system using two electrodes. Four human subjects were trained on the BCI system. Using only motor imagery, subjects successfully reached a 2-D presented target 89%, 87%, 68%, and 58% respectively of the trials. The probability of randomly selecting the correct target was 12.5%. This is the first demonstration of 2-D control using motor imagery with a low-cost BCI system. The success of this system questions the necessity of expensive EEG or invasive ECoG systems for users focusing on functionality. Training significantly improves 2-dimensional motion control for the limited number of subjects tested.

1 Introduction

Electroencephalogram (EEG) based Brain-Computer Interfaces (BCIs) are a growing research focus because, unlike other types of BCIs such as electrocorticogram (ECoG) or intracortical methods, brain surgery is not required for EEG. However, EEG based BCIs are not without their own challenges. Because the electrical brain signals are measured from outside the scalp, EEG based BCIs face difficulties of a low signal-to-noise ratio, low spatial resolution, and low characteristic amplitudes [1]. In an attempt to overcome these challenges, BCI researchers have typically used expensive EEG acquisition systems with at least 64 electrodes [2, 3, 4, 5].

Using these systems, researchers have obtained a few excellent results in noninvasive BCI motion control. For example, researchers, such as Wolpaw et al. 2004 [2] and Trejo et al. 2006 [3], have exhibited two-dimensional motion control. Even three-dimensional control has been shown by a few researchers such as McFarland et al. 2010 [4] and LaFleur K et al. 2013 [5]. Although these systems have produced useful results, they also have inherent disadvantages. One significant disadvantage is that these systems are prohibitively expensive for many researchers and end users. The systems typically used in BCI research cost between $20,000 and $250,000 [6]. Another disadvantage is a concern of practicality since the large number of electrodes requires a significant amount of time to properly apply to the scalp.

In this study, the necessity of these systems and their disadvantages was challenged. Two-dimensional motion was attempted using a low-cost ($1,000) and easy-to-use (only two electrodes for movement control) BCI system. The results show that this system can support two-dimensional motion control comparable to the expensive, 64 electrode systems as well as similar studies using invasive ECoG methods [7].

2 Methods

2.1 Study Overview

The study's purpose was to teach human users to control two-dimensional motion of the computer cursor using motor imagery. The experiment had an initial calibration phase, a one-dimensional motion phase, and a two-dimensional motion phase. For all testing, the users faced a computer screen while seated comfortably in a chair with his/her arms resting on a table in front. Overall, the study consisted of eighteen, hour-long sessions over the period of six weeks for each user.

2.2 Human Subjects

Four adults (three men and one woman; ages 19-21) were the BCI users in this study. The study's protocol was approved by the Institutional Review Board (IRB) of Stevens Institute of Technology and all users gave informed consent. All of the participants were generally healthy and none had previous BCI experience.

2.3 Data Collection and Processing

EEG data was measured through a Neurobit Optima 4 data acquisition device with electrodes at Cz, C3, C4, and CPz. The electrode locations C3 and C4 were fully responsible for the movement control. The Cz and CPz electrodes were used solely for extra data collection purposes. Silver-silver chloride (Ag|AgCl) cup electrodes with NuPrep preparation gel and Ten20 conductive paste were used. Data was recorded at 125 Hz and all electrodes were referenced to the right earlobe. The open-source software, Brainbay, was used to record the EEG data and control the real-time cursor movement. EEGLAB v12.0.2.5b and MATLAB 7.12.0 (R2011a) were used for offline data analysis.

Cursor movement was controlled based on the EEG signal's power in a specific three-Hz-wide frequency band from a certain electrode for each dimension of motion. The frequency band for each user was determined based upon Event Related Desynchronization (ERD) data collected from the initial calibration session with each user. The power in the frequency band was calculated using a Fast Fourier Transform and a 4th order Butterworth-Bandpass filter. The

calculations were performed through Brainbay's "Magnitude" function.

To decide on movement direction, the average power of the last 400ms time period was compared to a constant value, specific for each user. This comparison constant was initially decided upon during the initial calibration phase. The constant was set as the midpoint between the power in the specified frequency band when the user was actively using motor imagery and when the user was relaxing. In order to account for changes in the power levels observed due to learning effects throughout the experiment, the comparison constant was adjustable at the end of each session in order to reset the value to the correct midpoint level.

For horizontal motion, if the calculated average power over the previous 400ms time period was greater than the comparison constant value, the cursor would move one pixel left. Otherwise, the cursor would move one pixel right. For vertical motion, if the calculated power was greater than the constant value, the cursor would move one pixel down. Otherwise, the cursor would move one pixel up. The horizontal and vertical movement decisions were made simultaneously and both were at a rate of 125 movements per second. Measurements were controlled for eye (EOG) and facial muscle (EMG) movement for each subject during calibration.

2.4 Initial Calibration

The initial calibration for each user took place during the first two sessions. The goal of the calibration was to decide on the electrode location, the frequency band, and the value of the comparison constant for each dimension of motion for each user. The two calibration sessions consisted of the user performing four to six Motor Imagery Tasks. A Motor Imagery Task consisted of forty iterations of an arrow appearing for one second and then the screen being blank for 5 seconds. The task contained twenty arrows pointing right and twenty arrows pointing left. The arrows appeared in a random order. The user was instructed to perform a specific imaginary body movement while the arrow was on the screen. For example, the user may have been asked to imagine moving his right hand during the right arrows and his left hand during the left arrows. Figure 1 shows example screens that the user would be presented with.

Figure 1. The three different screens that the user would be presented with during the Motor Imagery Task.

The EEG activity was measured during the Motor Imagery Tasks and analyzed after each session. For this offline analysis, the event-related spectral perturbation (ERSP) plots (figures 6, 7 and 8) were graphed for each Motor Imagery Task using EEGLAB. From these results, the most consistent observations in the power spectra were chosen to be the initial control parameters.

2.5 One-Dimensional Motion

The third-sixth sessions consisted of each user performing eight sets of one-dimensional Motion Control Task (four horizontal and four vertical). A set of Motion Control Task consisted of twenty trials where a random target would appear and the user had to touch the target with the cursor within ten seconds. The user was instructed to perform a certain imaginary movement to move the cursor right (or up) and to imagine relaxing that body part to move the opposite way. Once the user hit the target or the ten seconds ran out, the cursor was reset to the center and a new trial would begin. Each set of twenty trials lasted about 3-4 minutes depending on the ability of the user and a one minute rest period was given in between each set. Figure 2 shows examples screens from horizontal one-dimensional Motion Control Task.

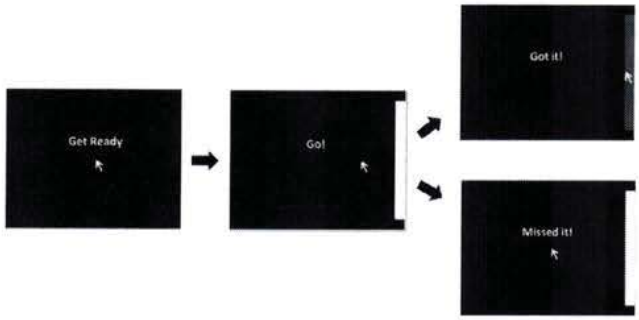


Figure 2. Horizontal Motion Control Task. The cursor is centered on the screen, a target appears on either the left or right side, and then the user uses motor imagery to control the cursor. The user must touch the target within ten seconds. If the user is successful, the screen says "Got it!" and the target lights up red. If the ten seconds run out, the screen says "Missed it!"

2.6 Two-Dimensional Motion

The seventh-eighteenth sessions consisted of each user performing eight sets of Motion Control Task (one horizontal, one vertical, and then six two-dimensional). The two-dimensional Motion Control Task was performed in the same manner as one-dimensional Motion Control Task, but the desired target location was selected out of eight possible locations. Each set consisted of twenty trials of a randomly chosen target location and the user was required to hit the target within ten seconds for a successful trial. Figure 3 shows all of the possible locations for two-dimensional Motion Control Task.

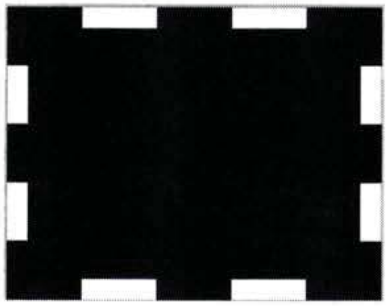


Figure 3. All target locations for two-dimensional Motion Control Task. For each trial, one random target appeared and the user attempted to navigate the cursor to that target.

ISBN: 1-60132-438-3, CSREA Press ©

3 Results

The success rates (target reached within 10 seconds) for the two-dimensional Motion Control Task were 89.2% (User C), 86.7% (User D), 67.5% (User B), and 57.5% (User A) for each user's final session. Figure 4 shows the success rates for two-dimensional Motion Control Task by the number of sessions completed for each user. Based on a Two Factor ANOVA without Replication analysis, the effect of sessions (practice) on the success rate of two-dimensional Motion Control Task was significant ($F=3.16$, $p<0.01$) indicating the users' performance improved with additional practice.

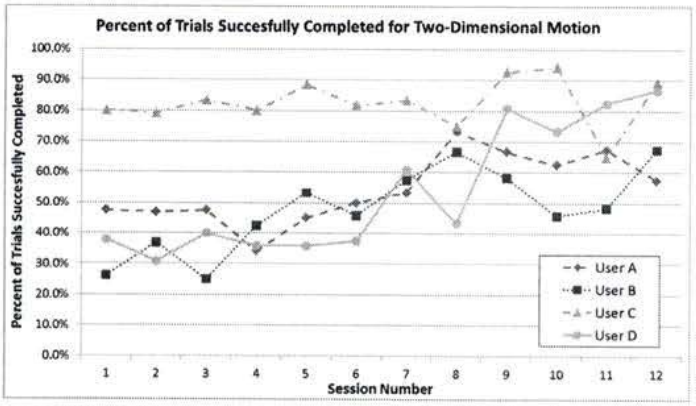


Figure 4. The success rates (target reached within 10 seconds) for two-dimensional Motion Control Task by the number of sessions completed for each user.

Based on a statistical Z-test, the percent of trials where the user reached the desired target without touching an incorrect target location was statistically significant at a very small p-level ($p<10^{-200}$). The desired target was the first target location to be reached (i.e. no incorrect target locations were reached) for 64.8% (User C), 57.5% (User D), 42.5% (User A), and 41.1% (User B) of the trials in each user's final session. The probability of randomly selecting the correct target was 12.5%. These data indicate that movement was not random and the users had very good accuracy in moving towards the desired target. Figure 5 plots the accuracy by the number of sessions completed for each user. Based on a Two Factor ANOVA without Replication analysis, the effect of sessions (practice) on the accuracy (desired target was reached without hitting an incorrect target) was statistically significant ($F=2.65$, $p<.015$). This indicates that the users improved their accuracy with additional practice.

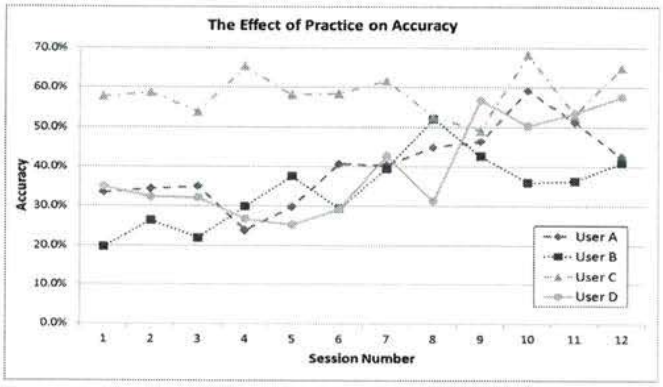


Figure 5. The effect of practice on accuracy for each user. Accuracy is defined as the percent of trials where the user reached the desired target without touching an incorrect target location.

All users used the C3 electrode location to control horizontal motion and the C4 electrode location to control vertical motion. The center of the 3-Hz-wide frequency band for users A and C was 11 Hz for both horizontal and vertical. User B used 10 Hz for both horizontal and vertical, while User D used 11 Hz for horizontal and 10 Hz for vertical.

During the initial calibration phase of the study, event-related spectral perturbation (ERSP) plots were generated from each user's EEG data while performing motor imagery in order to decide on the electrode location and frequency band for movement control. The benefit of an ERSP analysis over an Event-Related Potential (ERP) analysis is that ERSP analysis allows for simultaneous analysis of EEG activity with respect to time and the full spectrum of frequencies while ERP analysis can only analyze one frequency or frequency band per plot [8]. The ERSP plots show frequency along the vertical axis, time relative to the stimulus presentation on the horizontal axis, and EEG power though a color map. A red shifted color on the plot indicates that there was an increase in power at that frequency and time point. A blue shifted color indicates there was a decrease in power. The plots only display color points when the change in power was consistently present which was defined by significance testing at a specified level. Any points that were not statistically significant at that level appear green. Further information on the ERSP plots can be found in [8].

A sample of these ERSP plots from the initial calibration phase is shown in figure 6. For the plots in figure 6, the user was instructed to imagine wiggling his right fingers when presented with a right arrow and to imagine wiggling his left fingers when presented with a left arrow. In figure 6, the C3 electrode shows consistent desynchronization in the 8-14 Hz range when the

user imagined wiggling his right fingers. This User, C, from the initial calibration phase. The user imagined wiggling the fingers on his right or left hand for this test. The plot's vertical axis is the frequency range from 1 to 50 Hz and the horizontal axis is the time (-500 to 1000 ms) after the arrow is presented. The blue color indicates a statistically significant ($p<.05$) decrease in power compared to the baseline and the red color indicates an increase in power. The baseline was translated into motion by moving the cursor right when the desynchronization was detected and moving the cursor left when the desynchronization was not present. Also in figure 6, the C4 electrode shows consistent desynchronization in the 8-14 Hz range when the user imagined wiggling his left fingers. This was translated into motion by moving the cursor up when the desynchronization was detected and moving the cursor down when the desynchronization was not present.

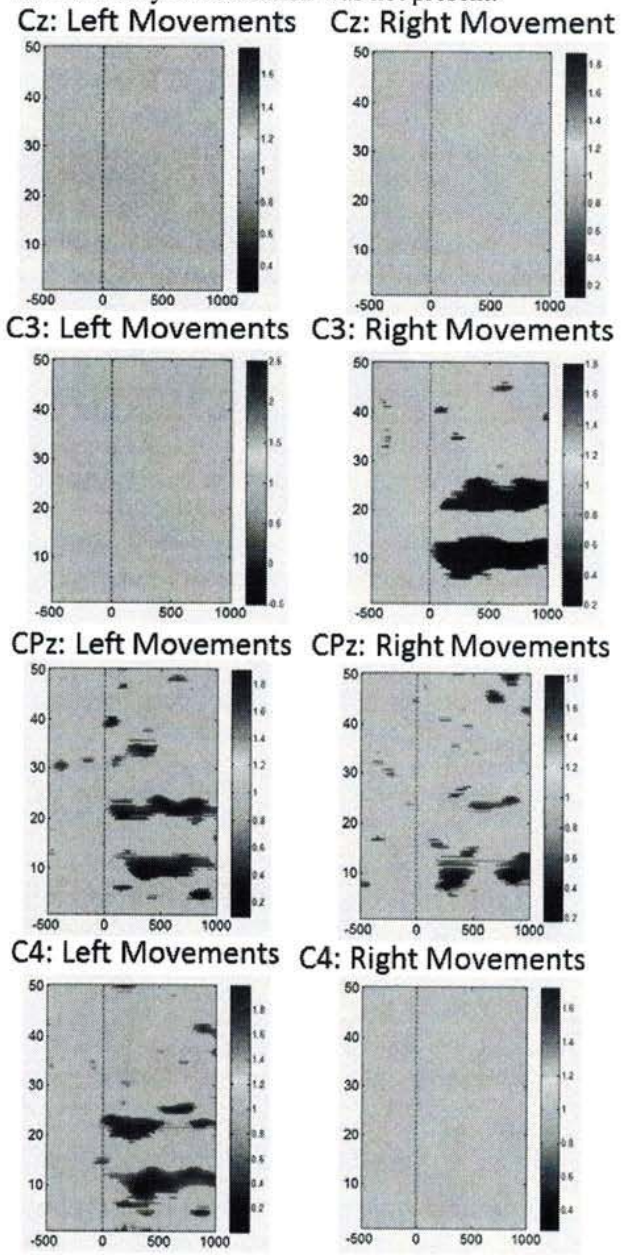


Figure 6. An event-related spectral perturbation (ERSP) plot for User C from the initial calibration phase. The user imagined wiggling the fingers on his right or left hand for this test. The plot's vertical axis is the frequency range from 1 to 50 Hz and the horizontal axis is the time (-500 to 1000 ms) after the arrow is presented. The blue color indicates a statistically significant ($p<.05$) decrease in power compared to the baseline and the red color indicates an increase in power. The baseline was defined as the one second prior to the arrow.

In order to ensure the users were truly controlling the cursor by modulating the brain's sensorimotor rhythms, ERSP plots were computed from one-dimensional Motion Control Task throughout the experiment. One-dimensional Motion Control Task was ideal for producing ERSP plots because the user would either actively be using motor imagery or be relaxing for the entire duration of the trial depending on the target location. ERSP plots were not able to be computed for two-dimensional Motion Control Task due to the dynamic nature of the task. For example, in two-dimensional Motion Control Task, the user may need to move right, but then back left if he overshot the target. The ERSP plots for one-dimensional vertical Motion Control Task for each user throughout the experiment are available on request to the contact author. They include ERSP plots focused on the 8 to 14 hertz range for the C4 electrode, which highlights the frequency band actually used for control as well as ERSP plots using a broad spectrum from 1 to 50 hertz for the C4 electrode in order to detect potential EMG or EOG artifacts.

4 Discussion

The results show that real-time, two-dimensional movement control can be achieved using a noninvasive, low-cost BCI system that uses only two electrodes. All four of the users were able to achieve quality two-dimensional movement control that would be useful to people with severe motor disabilities. The quality of control was found to be significantly enhanced with practice. This study only had four subjects and therefore, broad generalizations are difficult. However, all four of the subjects each improved with practice and these results support similar findings in [4]. The findings in this study support the conclusion that scalp-recorded EEG control is essentially a skill that the user needs to practice in order to master [4]. The participants in this study reported that, with practice, the initial motor imagery technique was used less to control the cursor. For example, one person stated that for optimal control he imagined physically pushing the cursor in the intended direction.

The quality of control from the present study was compared to previously reported two-dimensional control. Using a similar experimental setup, Wolpaw and McFarland reported that four users reached the target within the ten seconds allowed in 89%, 70%, 78%, and 92% of the trials [2]. In the present experiment, the four users reached the target within the ten seconds allowed in 89%, 87%, 68%, and 58% of the trials. Based on this comparison, the quality of control from the present experiment and Wolpaw and McFarland's experiment is deemed very comparable. In Wolpaw and McFarland's experiment, a 64 electrode EEG acquisition device was used. Two electrode locations were the controlling variables for motion, but a large Laplacian filter was used for each, therefore bringing the effective number of electrodes used to ten. The BCI system used in

this present study has a significant cost advantage and uses only two electrodes instead of ten, but still produces comparable results.

The success of this minimalist BCI system brings into question the necessity of expensive EEG acquisition systems with a large number of electrodes. EEG acquisition systems typically used for BCI studies cost between $20,000 and $250,000 [6]. For example, the Advanced Neuro Technology acquisition system (ANT Neuro, Netherlands) costs between $30,000 and $50,000 dollars [8]. According to the company's website, the 64 channel BioSemi acquisition system costs $36,000. The BCI system used in this study had two critical advantages over these types of systems: The EEG acquisition device only cost about $1,000 and only a short preparation time was required each session since only two electrodes needed to be placed on the scalp.

EEG systems with a large number of electrodes allow for analysis of the spatial localization of brain signals. This allows researchers to pinpoint regions on the scalp that correspond to certain actions. While this spatial localization is important for research, it is less important for a system intended for a person with a severe motor disability. The most important criterion for this user is that the device operates as intended. For this functionality, spatial localization is not necessary. The few electrodes used can be placed on scalp regions that, through previous high-density electrode readings in literature, have been indicated to generally correspond with a specific action. The EEG system with only a few electrodes can take advantage of this previous spatial localization information, but it does not necessarily need to be able to capture its own spatial localization information in order to be functional for the end user. The system presented in this study focuses on the end user, a person with a severe motor disability, and provides a functional system for a small cost with easy preparation.

This is the first time that real-time, two-dimensional motion control using motor imagery has been demonstrated using only two electrodes with a noninvasive, low-cost BCI system. Several researchers have used low-cost EEG systems for BCI applications, however, all have used BCI paradigms such as P300 [9, 10, 11], Steady state visually evoked potential (SSVEP) [12], or did not have real-time control [13]. Motor imagery based BCIs such as the one used in the study have the advantage of providing the users with an intuitive method of control capable of controlling motion in real-time.

Many low-cost EEG acquisition devices have recently become available, such as the Neurobit Optima 4 used in this study, Emotiv EPOC and Insight, iFocusBand, Muse by InteraXon , and OpenBCI. Each device will need be evaluated individually, but researchers should take this study as an example that low-cost devices can produce useful BCI results. The fact that low-cost devices can be useful in BCI research opens the door to many researchers because it greatly reduces the financial hurdle to become involved. This will enable more researchers to participate in BCI research and overall will accelerate the advance of this technology.

In the future, this experiment will be extended to three-dimensions to test the limits of this noninvasive, low-cost BCI system by adding a third electrode. Three-dimensional motion control is the current highest level of control shown by EEG based BCI systems [4, 5]. The future study will include a larger number of subjects in order to produce more generalizable results. Also, machine learning algorithms will be applied during this three-dimensional experiment in an attempt to improve quality of control without an increase in cost.

5 Conclusions

In this study, a noninvasive, low-cost BCI system proved capable of real-time two-dimensional motion control while using only two electrodes. This demonstrates that low-cost BCI systems can produce useful results and therefore questions the necessity of expensive EEG acquisition systems when developing systems for end users. The usefulness of low-cost BCI systems lowers the financial hurdle to become involved in BCI research and should lead to an increase of researchers and technological progress in the field. More importantly, as more people with severe motor disabilities begin using BCIs, it will become vital to make these systems affordable for these users.

References

[1] Schalk G *et al* 2008 Two-Dimensional Movement Control Using Electrocorticographic Signals in Humans *J Neural Eng.* **5** 75-84

[2] Wolpaw J R and McFarland D J 2004 Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans *Proc Natl Acad Sci USA* **101** 17849-54

[3] Trejo L J, Rosipal R, and Matthews B 2006 Brain–Computer Interfaces for 1-D and 2-D Cursor Control: Designs Using Volitional Control of the EEG Spectrum or Steady-State Visual Evoked Potentials *IEEE Transactions on Neural Systems and Rehabilitation Eng.* **14** 225-29

[4] McFarland D J, Sarnacki W A, and Wolpaw J R 2010 Electroencephalographic (EEG) Control of Three-Dimensional Movement *J. Neural Eng.* **7** 036007

[5] LaFleur K *et al* 2013 Quadcopter Control in Three-Dimensional Space Using a Noninvasive Motor Imagery-Based Brain-Computer Interface *J. Neural Eng.* **10** 046003

[6] Lee J C and Tan D S 2006 Using a Low-cost Electroencephalograph for Task Classification in HCI Research *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology* 81-90

[7] Leuthardt, E C, Schalk, G, Wolpaw, J R, Ojemann, J G, and Moran, D W 2004 A Brain-Computer interface using electrocorticographic signals in humans *J Neural Engineering* **1** 63 - 71

[8] Makeig S 1993 Auditory event-related dynamics of the EEG spectrum and effects of exposure to tones *Electroencephalography and Clinical Neurophysiology* 86: 283-293

[9] Duvinage M, Castermans T, Petieau M, Hoellinger T, Cheron G, and Dutoit T 2013 Performance of the Emotiv Epoc Headset for P300-based Applications *BioMedical Engineering* **12** 56

[10] Debener S, Minow F, Emkes R, Gandras K, and Vos M 2012 How about taking a low-cost, small, and wireless EEG for a walk? *Psychophysiology* 49: 1617-1621
[11] Mayaud L, Congedo M, Van Laghenhove A, Fig`ere M, Azabou E, et al.. 2013 A comparison of recording modalities of P300 event-related potentials (ERP) for brain-computer interface (BCI) paradigm *Clinical Neurophysiology* 43: 214-227
[12] Liu Y, Jiang X, Cao T, Wan F, et al. 2012 Implementation of SSVEP based BCI with Emotiv EPOC *Virtual Environments Human-Computer Interfaces and Measurement Systems* 34-37
[13] Zich C, De Vos M, Kranczioch C, Debener S 2014 Wireless EEG with individualized channel layout enables efficient motor imagery training *Clincial Neurophysiology*

OCR for Unreadable Damaged Characters on PCBs Using GSC Algorithm and kNN Classifier

Carlos F. Nava-Dueñas, *member IEEE*
Skyworks Solutions, Inc.; Engineering Institute
UABC, Mexicali, Mexico
nava.carlos@uabc.edu.mx

Felix F. Gonzalez-Navarro, *member SMIA*
Engineering Institute
UABC, Mexicali, Mexico
fernando.gonzalez@uabc.edu.mx

***Abstract*— In this paper, we propose to change the actual implemented pattern matching method to have optical character recognition by implementing the Gradient, Structural, Concavity (GSC) algorithm to extract the features of damaged, unreadable or incomplete numerical digit characters from images on printed board circuits (PCBs). Grayscale color images are acquired from a charge-coupled device (CCD) camera, assembling a dataset of 500 matrix images samples for the character digits from 0 to 9. The GSC feature extraction method is applied to get the characteristics that will be used in the character recognition step. Experimental results show that applying GSC algorithm to extract the features and using k-Nearest Neighbor (kNN) Classifier with the Euclidian Distance can improve optical character recognition (OCR) detectability of damaged characters from actual 95% to more than 97% in early tests.**

Keywords*— machine vision, optical character recognition (OCR), kNN Classifier, GSC, pattern recognition, printed circuit board (PCB).

I. Introduction

Optical character recognition (OCR) has been an important technology used to convert characters from a digital image to a digital text. There are basically two types of OCR algorithms: the first technique is related with the matching of matrix images, where an alphabet of stored character images is used to compare with an input image [1], [2]. This pattern matching does not work well when new fonts are encountered or input character images are unreadable. The second technique decomposes an input image to extract the principal features [3], [4], [5]. Then, classifiers are used to compare the input image features with some stored image features and choose the best match.

Our actual system implemented at the Skyworks factory uses the traditional OCR technique i.e. pattern matching. Our implemented vision system reads identification characters on printed circuits boards (PCBs) for lot integrity and machine control. This commonly-used technique is not robust enough because many of the images on PCBs present some damaged characters due to dirt or as a result of bad previous processes [1]. Actual OCR detectability is around 95% at best. It starts with a monochrome VGA image acquisition of the upper left section of a PCB, using a NI-1752 smart camera, with full resolution, 640x480 pixels and maximum data transfer @60 fps using a GigE port. The selected resolution and data transfer speed parameters meet the factory production schedule of inspected PCBs. The camera has a grayscale output image type with a maximum character resolution to cover the entire PCB characters positions, as shown in Fig. 1.

Fig. 1. PCB with no damaged characters.

Due to some problems with previous processes in the production line, some PCBs will have some residual dirt over the characters, making some characters unreadable for the pattern matching technique, as shown in the following Fig. 2.

Fig. 2. PCBs with evident residual dirt over characters.

The principal problem is that operators have lower throughput than automatic OCR software, and this leads to manually writing down the information from the screen when the actual recognition software fails, increasing the process time, making possible errors from wrong readings, resulting in higher production costs. Taking into consideration these facts, a better approach has to be considered [3].

This paper presents a proposal for implementing a character recognition technique for unreadable characters using Gradient, Structural and Concavity (GSC) extraction features and K-Nearest Neighbor Classifier using Euclidian Distance [6], [7], [8].

II. Data Set

The experimental data set consists of 500 character images, 50 images correspond to each numerical digit from 0 to 9. Fig. 3 shows some examples of damaged digit image samples.

Fig. 3. Some damaged digit images from dataset

For our previous dataset, a pre-processing step is applied as follows:

Let I_i be any digit image of size (k,l) -see Fig.4- from the original dataset, $\forall I_i$:

.

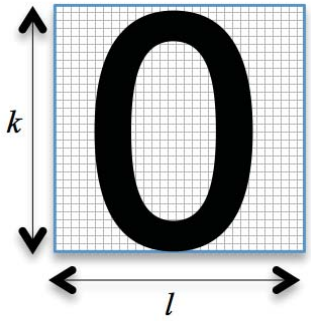


Fig. 4. I_i digit image matrix with size (k,l). k=50 and l=30 in our experiments

1. *Convert to gray-scale (if previous images are color RGB type).*
2. *A threshold is applied to binarize.*
3. *I_i is split in 60 non-overlapping regions (10 x 6 grid), as shown is the Fig. 5.*

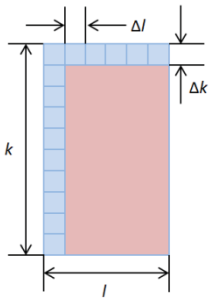


Fig. 5. I_i. is split in a 10 x 6 grid of $\Delta k = \Delta l = 5$ pixels

III. Gradient, Structural and Concavity (GSC) Recognition Algorithm

The GSC algorithm to extract information from the image was implemented [3]. It *constructs* features of an image by applying a three-step feature extraction process: 1-*Gradient step* detect local features by analyzing the stroke shape on small distance; 2-*Structural* step, extract features from stroke trajectories by extending distances of gradient; 3-*Concavity* analysis detects stroke relationships across the image. Fig. 6 shows the final Total Feature Vector (TFV) constructed for each image.

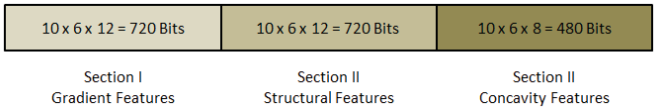


Fig. 6. Total Feature Vector of each I_i digit image with size of 1920 pixels

Section I - Gradient Features. Two dimensional convolutions in the X and Y direction is applied to get the gradient features using an 3 x 3 Sobel operators on the original I_i binary image, -see Fig. 7. Gradients from an image representation of the Digit Character 9 are shown in Fig. 8.

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$G_x = K_x * M \qquad G_y = K_y * M$$

Fig. 7. G_x and G_y are the 2D convolution of a 3 x 3 matrix for every pixel on original I_i digit image matrix. An extra zero padded border is added to I_i.

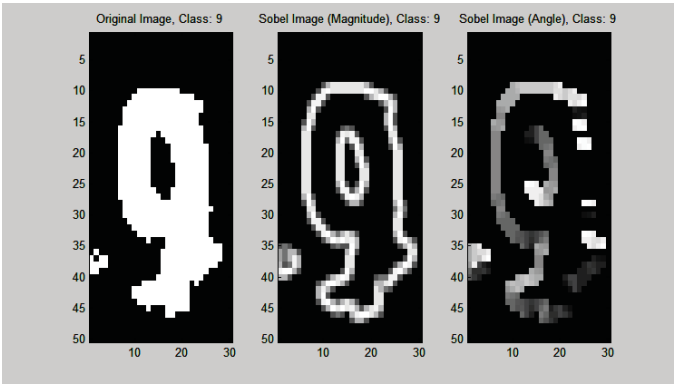


Fig. 8. Gradient magnitude and direction are shown for Digit Image 9

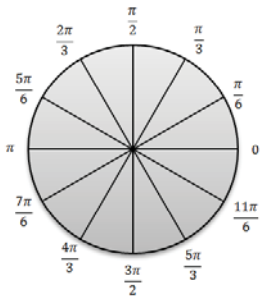


Fig. 9. Gradient range from 0 to 2π in 12 equal space regions

A histogram is applied for each of the 60 non-overlapping regions of the complete image grid, incrementing the counter for every Gradient Angle that falls in each region, as indicated in Fig. 9. A threshold is applied and a final 720 Bits is created for the first part of the TFV, as previously shown in Fig. 6.

Section II - Structural Features. For each pixel of the expanded I_i digit image with zero padded border, a set of 12 rules is applied using 8 pixels around the main pixel. These rules look for specific gradient patterns form with the nearest pixels, like horizontal lines (0, 4), vertical lines (2, 6), diagonals [(5, 1), (3, 7)] and corners [(0, 2), (2, 4), (4, 6), (6, 0)]. Fig. 10 shows these rules in a graphical positioning:

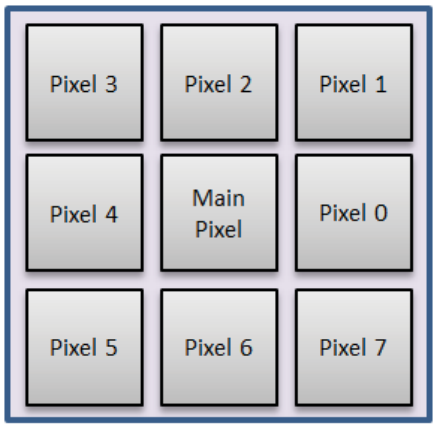


Fig. 10. Eight nearest pixels around Main Pixel

A threshold is applied for each of the 12 rules result, for each of the 60 non-overlapping regions to binarize the complete set. A final 720 Bits set is created for the second part of the TFV, as previously shown in Fig. 6.

Section III - Concavity Features. Three feature sections form the last part of the GSC algorithm:

1) **Coarse Pixel Density:** A histogram is applied to count all the character pixels at each of the 60 non-overlapping regions. Then, a threshold is applied to binarize the result. For this, a 60 Bits new set is included as the first part of the Concavity Features.

2) **Large-Stroke:** Like the previous section, two histograms are applied, one for the horizontal and one for the vertical pixels strokes in each direction. A threshold is applied to binarize the result. For this section, a 2 x 60 = 120 Bits new set is included as the second part.

3) **Upward, Downward, Left, Right and Holes:** In this last section of the concavity features, for every pixel in each of the 60 non-overlapping regions, rays are fired to hit character pixels, borders and check if we have holes or character pixels in all directions. In this last section, a 5 x 60 = 300 Bits new set is included.

As shown in Fig. 6, the TFV for the Section III has the following Bits set:

$$1\,x\ (10\,x\,6) + 2\,x\,(10\,x\,6) + 5\,x\,(10\,x\,6) = 480 \text{ Bits}$$

IV. K-Nearest Neighbor Classifier

The k-Nearest Neighbor (kNN) Classifier Algorithm was chosen for the damaged character recognition step. It was selected because of its simplicity and fast performance, and the absence of *prior* assumptions about data set probability distributions. The classification occurs when a majority vote among kNNs with respect to any particular test set is given. In the complete experiments *k* parameter was fixed to 3.

The resulting *Total Feature Matrix* has the form as shown in Fig. 11 –see *Class* is added at the end column:

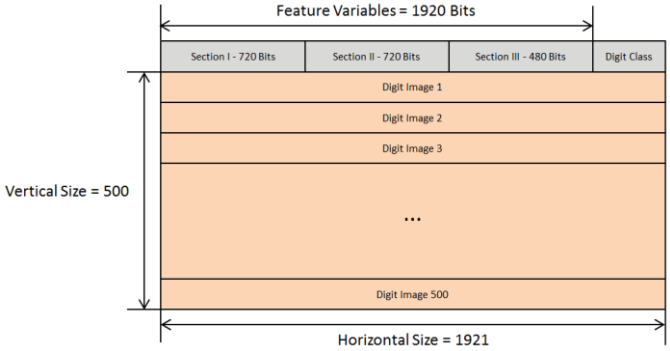


Fig. 11. Final Total Feature Matrix for all Image Dataset with Digit Class included (0 – 9)

The Euclidian Distance was chosen as the first approach to compute the distance metric as follows:

$$D_{Euclidian} = \sqrt{\sum_{i=1}^{l}(Test_i - Training_i)^2} \qquad (1)$$

V. Experimental Results

A first-round experiments were conducted as follows: a kNN Euclidian Distance and a K = 3 was trained and tested varying the proportion of training-test samples. A range of 10% to 90% training and 90% to 10% test sets were analyzed. Fig. 12 shows this experimental setting result. Fig. 13 shows a zoom of the 60% to 90% section.

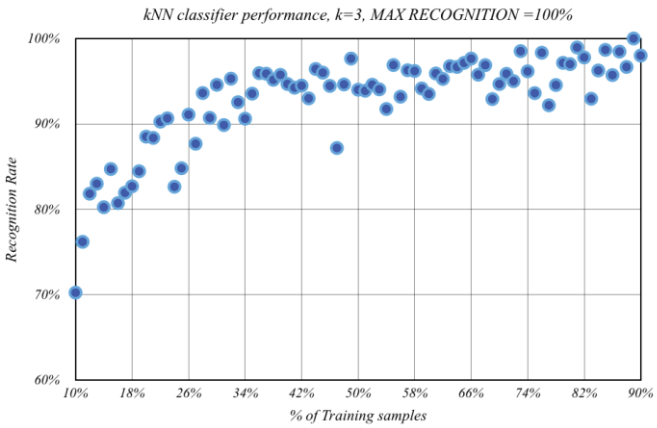


Fig. 12. kNN Classifier performance for all image dataset

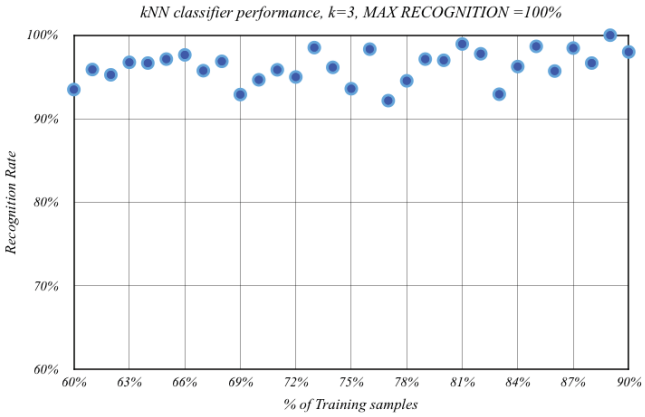


Fig. 13. Zoom in the kNN Classifier performance.

It's seen that classification stage yields promising results. The kNN Classifier shows interesting peaks of 100% recognition rate at high training percentage by using the GSC algorithm.

In order to assess more precisely the classification performance of the GSC+kNN proposal, Monte Carlo cross-validation strategy was selected [9]. A total of 100 random data splits of 90% training and 10% test samples were analyzed. Fig. 14 shows results and main statistics.

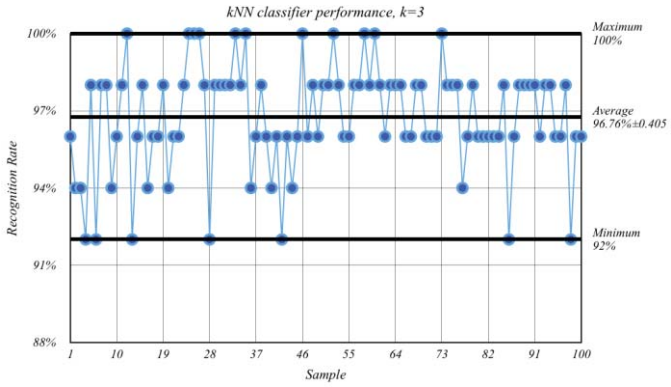


Fig. 14. Monte Carlo 100 cross-validations with a 90%-10% training-test split.

VI. Conclusions

The implementation of the GSC Algorithm and kNN Classifier with Euclidian Distance shows an improvement for the readings of damaged or incomplete characters using optical character recognition from a previous detectability strategy (Pattern Matching) from 95% to 96.76% ±0.405. Future work will consider increasing the dataset samples and the use of other distance metrics, as well as other classification algorithms.

References

[1] S. Mori, C. Suen and K. Yamamoto, "Historical review of OCR research and development", *Proceedings of the IEEE*, vol. 80, no. 7, pp. 1029-1058, 1992.

[2] R. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 690-706, 1996.

[3] Favata, J., "A Multiple Feature/Resolution Approach to Handprinted Digit and Character Recognition", *Proceedings of the International Journal of Imaging Systems and Technology*, Vol. 7, pp. 304 - 311, 1996.

[4] D. Desrochers, Z. Qu and A. Saengdeejing, "OCR readability study and algorithms for testing partially damaged characters", *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing. ISIMP 2001 (IEEE Cat. No.01EX489)*, 2001.

[5] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. John Wiley & Sons, New York, 2001.

[6] Ballerini, L., Fisher, R., Aldridge, B., Rees, J., "Non-Melanoma Skin Lesion Classification Using Colour Image Data in Hierarchical KNNClassifier", *Proceedings of 2012 IEEE International Symposium on Biomedical Imaging*, 2012.

[7] C. Rodriguez, "An Incremental and hierarchical KNNclassifier for Handwritten Characters", *Proceedings of 2002 IEEE International Conference on Pattern Recognition*, vol. 3, pp 98-101, 2002.

[8] J. Hu, C. Yan, "Predicting Protein Subcelluar Localizations Using Weighted Euclidian Distance", *Proceedings of 2007 IEEE 7th International Symposium on BioInformatics and BioEngineering*, pp 1370-1373, 2007.

[9] X. Qing-Song, L. Yi-Zeng, "Monte Carlo cross validation", *Chemometrics and Intelligent Laboratory Systems*, Volume 56, Issue 1, 16 April 2001, Pages 1-11.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

NATURAL LANGUAGE PROCESSING, MACHINE TRANSLATION, WORD CATEGORIZATION, AND RELATED TOPICS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

CAT'S Cradle: A Cognitive-Parametric Model of Computer Assisted Translation

Boris Gorbis
NAFI, 1741 Sunset Plaza Drive, Los Angeles, CA 90069,USA

Abstract:

Establishing inter-lingual textual equivalency, requires cognitive-linguistic competence as a metalinguistic bridge between different codes. This CAT model offers a 9 + step language-independent protocol to correlate Multi-Word-Complexes in different languages as a product of a parsing algorithm ('Point of View') which actualizes one cognitive 'aspect' of a referent shared with many operants. Every parsing choice is assigned a unique alpha-numeric code and each word-form obtains a sequence of codes as its unique parameter. Each sequence is a digital analogue of all cognitive-informational 'aspects' of a referent, operant, modifier, quantifier actualized by all instances of parsing. A protocol of matching word-form codes identifies parametric coordinates of a Multi-Word-Complex employed as a "unit of translation". Correlation is established when Multi-Word-Complex components with the same code parameters are located in the target language. Mapping code progressions in the two texts establishes and verifies cognitive equivalency. The paper offers theoretical basis and practical solutions of implementing cognitive-parametric design for CAT.

Key words: *machine translation, computer assisted translation, artificial intelligence, relonics, cognitive modeling, shared reality, linguistic competence, metalinguistics,*

1. Translation as Replication of Cognitive - Linguistic Behavior

Adequate translation is a result of replicating author's cognitive-linguistic choices through selection of cognitively equivalent means in the 'to-language'. Consequently, CAT application must provide both cognitive and linguistic equivalency.

Deconstructing natural languages for *'equivalency of meaning*' presents well-known obstacles even for human translators. Efforts to formalize CAT protocols run into hyper-polysemy, where authors' subjective *'meaning'* can fit many lexical canisters. Adequate machine-enabled translation requires a language-independent *mediator* to establish equivalence of text-forming processes expressed by different means in the original and target languages. [1][2].

This model of '*cognitive-linguistic competence'* for CAT is based on parametric formalization of human "*relational knowledge*" as *'prior knowledge'* provided to a machine. If adopted, it may advance to autonomous computer generation of texts. [3][11]

We assume that *adequate translation* of a text means that: (a) text was generated for *many recipients*; (b) text is a corpus of *recognizable* lexical forms word-forms); (c) each lexical form is a substitute *for* some *cognitive element;* (d) recipients can *identify* these cognitive elements; (e) lexical forms appear in a certain *order* to form *connections recognizable* by recipients; (f) progression and connections of lexical forms reflect *relations* of their cognitive elements as *intended* by the author; and (g) these relations cannot be changed by means (rules) of another language.

Individual *'competence'* of moving through these stages depends on one's *'relational knowledge'*, i.e. skills of establishing and recognizing relations between referents. Lexical forms and their semantic meaning play an important but not the main part in our modeling *'competence'* as a person-independent, all-inclusive complex of cognitive *skills*. A computer has none of these skills; they cannot be *'taught'*, but they can be *simulated* in a *generic* i.e. person/language independent model. This design is described in what follows.

2. Cognitive-Linguistic Competence of 'Shared Reality'

To explain how we offer to replicate *'relational knowledge'* as *capacity to operate with any referent as a cognitive potentiality,* we shall review well known facts about children. Regardless of their language, children "know" that anything can be a *'toy'* and as a *'toy'* can be treated the same way and that one can *play, bang, hug, drop, share* and do other things with any and all '*Toys*.' Moreover, any speaker, older than 3, can place any *'toy*' into mental 'boxes' such as: "*things that fall*", "*things that present danger*", "*things I can put in my mouth*", "*heavy things", "dad's things, 'things that break'* and so on. We quickly learn that many other "*objects"* do not fit into these 'boxes'; especially those not allowed in the mouth.

This hard-wired skill to categorize anything by organizing *classes of elements* is matched by our skill to

differentiate *the same object* by placing it into as many cognitive 'boxes' as we can. The *'toy'* becomes a *'time-piece', a 'pipe'* or a *'phone'* - a new *element* with name, uses and actions different from those of a *'toy'* and the skill to differentiate objects matures into *'competence'*.

Human *cognitive competence is the skill of creating multiple representations ('pictures') of reality with the same referent(s).*This language-independent ability *to create and recognize classes of objects with shared properties* is our main social asset. Using *cognitive means* we can relate anything that is *'the same'* to anything *'different*' from everything else as a personal *map of reality* that must be socially and cognitively shared with others.

By grasping protocols of naming, grouping and separating actions and objects, we develop *'relational knowledge' as a universally-shared, comprehensive, person- and language- independent complex of skills of parsing and combining elements of the 'shared reality'*.[1] It is enabled by both diversity of means and economy of language that parallel the relonic and metarelonic structure of cognitive maps[2].

With this nod to linguistic part of *'competence'*, this paper describes protocol of replicating human *'relational knowledge'* outside of any natural language.

3. 'Shared Reality' with Embedded Relational Knowledge

Human 'relational knowledge' is a child of *'shared reality*' - a cognitive abstraction of the multi-dimensional nexus of *all existing and imaginary referents in every possible relation to all other cognitive referents and to themselves.*

As a "cognitive universal" it incorporates all specific and individual representations and any individual representation of *'shared reality'* is both partial and cumulative.[2] To enable comprehension, any such representation must be scaleable, yet limited. As a minimum, it must enable our recognition of two or more referents in every relation to or with each other, including itself. [6]

The number of 'shared reality' elements has no limit as new objects expand this complexity. However, this infinite complexity is typologically constrained and organized into a *mutually recognizable nexus* by the unchanging *typology of relations* between object-referents and *meta-relations* between their relations.

Typologically, the nexus of relations between any referent (person) and everything and everyone else (itself included) is a defined set of Relational Spheres. The number of Relational Spheres that establish, permit, define or require specific types of relations between any cognitive referents is a *universal constant*.[3]

This unchangeable structure frames (collectively and individually) our cognitive *'competence'* of parsing *'shared reality'* and processing the results. It is cognitively embedded in every individual as relonic skills and means "*of control, regulation and coordination in complex, super-complex, systems and in systems with infinite complexity.*" [7][8]

The relational-operative nature of our *cognitive knowledge processing* is simultaneously fluid and preset. This duality enables us to identify and establish patterns and association rules "*in systems of any complexity with presently unpredictable dynamics and outcomes.*" [4] [5]

Specifically, the relonic (metarelonic) duality of *'shared reality'* allows us to operate with any number of cognitive referents irrespective of their metrics and parameters (e.g. origin, composition, size, measurement, age, status, etc) as *objects of thought* that are: (i) the same (for any specific operation), (ii) equally available, and (iii) independent of anyone's language or code.

4. Cognitive Referent as A Potentiality

A *'referent'* is any distinct element of 'shared reality' but here we shall apply this term to *cognitive objects*. Our knowledge processing begins with perception of a cognitive object as a complex of all of its *possible attributes.* Any attribute we focus on becomes *a cognitive aspect of the referent* that may be *shared* or not shared by other referents. Any association that we establish for any two referents is an actualization of their *shared attribute*. In fact, we are free to choose any *cognitive-informational aspect* of each referent from the many possible. This is how languages develop.[6]

A cognitive referent in one's *'shared reality'* exists as *an objective potentiality of all relations* in which this referent can be engaged, and as *a subjective complex of all of its cognitive-informational aspects* one knows. A name-*word* of a referent is likewise an *informational potentiality* of its many attributes. The same '*ball*' in a store, in the kitchen or on a play-field has different cognitive-informational aspects as an *object of desire, a commercial product, a nuisance, an object of physical action, or a moving object to avoid* and so on.

We model our '*relational knowledge'* of a *'ball'* in a text as *cognitive processes of choice, identification and recognition of one relation between 'ball' and any other referent.* These *'knowledge'* skills are universal for both

[1] The suggested design, its methods and procedures are not the same as associative *'semantic mapping'* typically defined as "*a visual strategy for vocabulary expansion and extension of knowledge by displaying in categories words related to one another*."

[2] *'Relonic'* refers to relations between 'object referents', whereas *'metarelonic'* describes relations between 'relations'. Any cognitive structure is thus a unification of relonic and meta-relonic properties.[4][5]

[3] Typologically, any two-way relations, with living objects, physical objects, moving objects, objects of value, etc., that are available to us today are the same that were available to the first 'Homo Sapienses.' Only the context and tools have changed.

author and recipient, but their *operation* is different for each one. Text recipient must rely on word-forms to *identify* and *differentiate* referents and then proceed to *recognize* the intended *relationship* between them.

The tasks of text author are not the same. The author's choice of cognitive elements is dictated by his situational (task-related) *choice of an intended relationship* between them. The choice of word-forms is not arbitrary; they must represent one attribute of these elements and make this *'angle'* recognizable and comprehensible by all recipients (or a select few).

Author's choice of a certain relation between referents is a cognitive event of *application of a parsing criterion*. The more parsing criteria are recognizable as author's deliberate choices of word-forms, the greater is the *cognitive value*[4] of the text.

These tools of natural language processing by our *'relational knowledge'* can be simulated in CAT as collective *'prior knowledge'* by providing an inventory of all *parsing criteria* that relate as many cognitive elements as we choose to place in the multi-language databases of lexical forms. The main focus of our modeling would then be replication of *'shared reality' parsing* by two interdependent cognitive algorithms of *categorization ↔ differentiation.*

5. Parametric POV Design

Categorization is a process of parsing *'shared reality'* by one criterion that actualizes *one specific relation* for two or more referents. Each parsing criterion used organizes *a class of referents* that acquire *a shared aspect* as their *cognitive value*, but shed all or much of their individual *semantic meaning*.

To illustrate, the use of such parsing criteria as *'object of commerce'*, *'man made object', 'object of value', etc.,* facilitates our relonic recognition that such diverse word-forms as *'car', 'love', 'Baby Ruth ball', seasonal ticket', 'space'*, *'wipers'* or *'loyalty'* share the same *cognitive aspect* and thus have the same *cognitive value*, while in other instances they would not. The parsing focus is always on one attribute (*cognitive aspect*) of an element whose primary or essential attributes are ignored.

The author's choice of a parsing criterion results in a string of word-forms selection bringing the intended attribute into focus. To combine this with recipient's tasks, our *'cognitive-linguistic competence'* model must replicate (a) objective recognition of a subjectively chosen parsing criterion, (b) knowledge of cognitive elements it differentiated (c) knowledge of lexical means identifying these elements by their cognitive-informational aspect, and (d) *predicting* what parsing criteria and cognitive elements might appear further in the text.[5]

The totality of parsing criteria available to any person might appear unlimited, but it is not. Aside from constraints of individual knowledge and cultural-social environments, there is an outside boundary. *The number of all parsing criteria is recognized as a cognitive constant*[6] and forms a finite inventory[7].

The choice/recognition of a parsing criterion is defined in what follows as *a choice of a Point-of-View* (a POV). The use of POV as an analogue of cognitive algorithms of *categorization↔differentiation*, is the *heart* of our parametric CAT design. Let's illustrate.

Instead of the above word-form *'heart'*, we could have used word-forms *'center', 'substance', 'core', 'cornerstone', 'backbone,' 'blood'*, etc. These forms are now *recognized* as belonging to a *class* of referents with the same POV-supplied cognitive *aspect* - a product of our choosing the *'main object'* as a parsing criterion.

By itself, any word-form of referents from this class has *undifferentiated cognitive value* in translation. When parsed by other POVs, *'heart',* will manifest many different *cognitive-informational aspects.* As a cognitive phenomenon, a *'heart'* referent in any language is a fuzzy cluster of its many *'aspects'*. In this model it can be seen as a *potentiality* of all POV criteria by which it can be parsed. How do we identify our *'main object'* POV for machine translation purposes?

The *differentiation algorithm* is used as a protocol of finding the *parsing criterion* that joined word-forms into a cognitive unit. As a *potentiality* of all possible parsing outcomes, a stand-alone word-form of a referent (or any other element) is a semantic *uncertainty*. The key is that correct identification of the POV choice in the text deconstructs the *potentiality of at least two word-form(s)* and resolves the semantic uncertainty of each.[9] [10]

Let's take it step-by-step. In general, our POV-based parametric design separates *'shared reality'* referents: objects, states, relations, actions, conditions, processes and qualities, organizes them into different *classes* and *sets* and will matchs them by using a parametric analogue of a *'cognitive aspect*. Each POV referent class has a certain *volume* - 'wide volume' classes are bulky, like a POV of *'human object'* or *'physical object'*, while some have 'narrow volumes', like

[4] *'Cognitive Value'* is *a measure* of *comprehension*, i.e. recognition of cognitive aspects by recipients, and adequate translation of a combination of word-forms or text as a whole. *'Cognitive value'* must be sufficient to formulate an appropriate response (speech) and to identify adequate correlates (translation).

[5] In this model *'predictive processing'* is utilized as means of verification by comparing progression of POV codes *(textual mindprint*) in the original and the translated texts.

[6] This limit derives from the universal constant of Relational Spheres, the unchanging *types of relations* that any referent (man, rock, rabbit or deity) can enter into with any another referent.

[7] Full description of the POV inventory architecture is outside the scope of this paper.

POVs of *'perishable objects*' or *'reflecting objects'*[8] however, the huge size of a POV referent class presents no difficulty whatsoever. It is true that POVs form an *open class* permitting any author to place any referent in the class, but only for a price. By becoming a POV class member, *a referent is cognitively framed by the same attribute as other class members* while its meaning (encyclopedic properties) become non-essential.[9] The essential part here is that any POV supplies *identical cognitive value* to its class of referents. As the key to POV identification, this underlies our *nine+steps* of parametric modeling.

Step 1. *Assign a different alpha-numeric code* to each POV from the comprehensive inventory provided.
Step 2. Identify referent class members of a POV from the databases of word-forms (in any number of languages) and assign this *code to every POV class member word-form* like a cattle brand.

Before proceeding further, let's consider text with a POV code '*xaxa10'* that groups *apples*, *oranges*, *figs, plums* and *pears* together with *toothpaste, refrigerators, carpets, coffee cabbages, and condoms* as *'objects of commerce'*. Some of these word-forms *might* also appear in the text as *'edible objects'* with the code '*xaxa16*' while all would also carry code *'xaxa 01'*as *'physical objects'* and code *'xaxa 55'* for '*quantifiable objects'*.

To account for shorthand references abound in any text, our model provides an *avatar-coordination protocol*, whereby pronouns (*he/she/we/it*) and other referent substitutes, (*thing/this/it/gizmo/issue*, etc.,) are branded with the same POV code as the referent for which they stand. Because *plurality* is parsed differently than a single referent, a plural word-form would carry POV codes different from its singular form: e.g. *'hands', 'armies', 'reactors'*, and *'scholars'* carry code of *'multiple objects'* in addition to *'countable objects'*. To enhance recognition of object-referents, grammatical markers e.g. *article* and *case* will refer the program to a database of case forms while other markers provide access to databases of proper names, toponyms, (hydronyms/oronyms), acronyms, etc.

Step 3. Identify all POV codes for each referent word-form in the databases and obtain the full code sequence for each one.

An 'apple' word-form from our previous example would be represented by a sequence of *xaxa10'+ 'xaxa 01'+ 'xaxa 55'+ 'xaxa16* codes. There are many other codes for an 'apple' - as many as there are POV's that its cognitive referent might be parsed with. A code sequence obtained by separately coding all cognitive aspects (POVs) of a word-form represents a cognitive whole where each code represents a separate parsing event. The complete code sequence of a word-form becomes its *proprietary aspect parameter* (PAP). In most instances, a word-form PAP is sufficient for its differentiation from other word-forms in the text[10].

Because categorization requires differentiation, most POVs form *nesting structures*, where a wide volume POV may have fully or partially embedded narrow POVs - each forming its own sub-class of referents, and its own alpha numeric code correlated to the wide volume code. This is reflected in the code sequence organized by associative geometry.[11]

Let's use '*xaxa511*' as the POV code of *'liquid object'* as embedded in POVs of *'physical object', 'man-made object'*, *'measurable object'* (and others) and thus branches off their codes. In turn, the POV "*liquid objects'* embeds numerous narrow POV's, such as: *'liquids to ingest', 'liquids to traverse', 'liquids to buy', 'internal liquids', 'external liquids, 'harmful liquids,' 'physical objects'. 'man-made objects'*, *'measurable liquids'* etc and each narrow POV code would be a branch off the *main xaxa511* code. The code sequence for such common word-forms as *'tea'* or *'milk'* or 'apple' would have significant *length* (number of main codes) and *depth* (branches) This geometry of POV 'nesting' results in a spatial 'map' of any code sequence.

The *complete code sequence* is not a language-defined 'semantic map' of a referent but its *metalingual cognitive map.* The longer and deeper the sequence of codes we obtain, the more likely that it is a *proprietary code sequence* - a *unique* parameter of a referent in all its complexity in the *'shared reality'*.

6. Differentiation Algorithm & Operants in a Set

We now provide the second key to identification of a POV from a text that is *differentiation* from *parsing* by any other POV. As a good host, each POV provides its guest-referents with its *aspect* code and the staff to service it. All referents of a POV class have access to cognitively preset elements, here called *operants,* that manifest their POV aspect. For example, the *'physical object'* POV includes relations established

[8] These and other examples of POVs are a shorthand. The POV inventory elements are formalized as "*referents with an XYZ capacity"* or *"referents with a capacity to be XYZ"*

[9] From the *'writing surface'* POV, we have seen people write on their skin, on business cards, on $100 bills as well as on yellow pads, computers, iPhone screens and even on trees and in the sky.

[10] There is at least one parsing criterion that actualizes one unique aspect of a referent to exclude other referents no matter how similar in other aspects.

[11] The code sequence is not a fixed chain. It is a multi-dimensional flexible organization of codes sensitive to text typology and aspect frequency for each text type.

by anything viewed as a physical object, e.g. anything that *'stands', 'occupies', 'moves, 'deteriorates', 'breaks', 'remains', etc.* Each referent of this POV class (from a *bush* to a *Bush*) can be recognized as such and 'exa*mined','seen','found','touched','weighted,'measured, 'moved','located', 'observed'' and so on.*

An 'operant' is a *cognitive-linguistic subroutine* that relates a referent to actions or conditions of the POV and lexically appears as verbs and their equivalents. For example, referents with a *'xaxa10'* code of *'commercial object'* POV are bundled with and serviced by a limited group of *cognitive subroutines* of this *'aspect'*, such as *'buy', 'sell', 'package', 'display' 'price', 'ship', 'receive', 'discount', 'stock,', 'find', 'brand', 'advertise', 'exchange', 'move', 'locate', 'run out'', 'replace', ''warranty', 'cost', 'deliver', 'demand', 'distribute', 'store', wherehouse', 'retail. 'wholesale', 'promote', 'label', merchandize', 'liquidate', 'recall', 'tax', 'license' 'import', 'export', ban' 'deliver' ,'scan' ,'receive', 'invoice', 'pay for', etc.*

The *'etc'* above is superficial since English and all other languages have a *limited number of lexical means dedicated to differentiating any referent* as *'an object of commerce'* from its other cognitive aspects that form a specialized '*spectrum'* or *'operant-set'* for this wide volume POV. Thus, most 'operant sets' are tiny compared to 'referent classes' and easy to identify. Because this set *'belongs'* to the entire POV class and any operant from the set can *'service'* any referent member, POV class is called a *Musketeer Class* [9] with each POV supplying the same *cognitive value* to its operants and referents.

Step 4. Identify all 'operant set elements' of a POV from the databases of word-forms (in any language) and assign POV *code to every operant-word* in the set.

Just as object-referents, an operant can service more than one POV and thus obtain a sequence of codes as its parameter, e.g. an operant-word *'have'* would carry POV codes of *'object of possession',* and *'object with attributes',* while operant-word *'like'* would carry POV codes of *'object of preference', 'object of comparison'* and*. 'object of approval'.*

Step 5. Identify all POV codes for each operant -word- in the databases and obtain the full code sequence for each one.

Upon completion of Steps 1 to 5, any word-form in this model databases is represented as a sequence of all POV alpha-numeric codes of its cognitive referent or operant 'aspects' and is available for *'matching'* as an analogue of 'cognitive pairing'.

7. Protocol of Parametric Matching

Thanks to the *cognitive aspect* supplied by a POV, a common verb, like *'have'*, *get'*, *'do'* or *'need'* becomes a powerful operant that *isolates an entire class of referents* and makes itself available for *cognitive pairing* with any referent of that class. Cognitive pairing produces an MWC - *a unit of lexical components with the same aspect and cognitive value* in any language.

Step 6 (a). Identify word-form units with the same POV codes (as products of cognitive pairing through shared aspect).

We begin this step with identification of operant(s). Because an operant code sequence (PAP) is much shorter than that of a referent and an operant set is smaller, 'operants' provide better discrimination than 'referents'. Furthermore, regardless of language, *each code in the operant PAP sequence is a parametrically defined connection to any word-form from the class of referents with the same POV code.* In general, (there are exceptions) each code in the word-form sequence provides a *parametrically defined* link to any operant with the same code.

Three rules of parametric matching thus emerge: (1) *any referent word-form also identifies all operant-words provided by each POV in its code sequence*, and (2) *any operant-word also identifies all referents it can service for each POV code the operant carries in its sequence.* (3) There are exceptions to the above: when a referent and operant with the same code do not match, there is *a POV where mismatch occurs*[12] *and the usage is a novelty or a metaphor.* [6]

To illustrate, let's use the *xaxa511* as the POV code for *'liquid objects'*. The program 'anticipates' that when a word-form with the *'liquid objects'* code of *xaxa511* is found in the text, e.g. 'serve*', 'dry up', 'flow', 'inject', 'drink', 'add', 'fill', 'contained', 'drain', 'spill', 'run over*', etc, there must be at least one word-form of a referent with the same *xaxa511* code. A *match* with the word-form *'spill'* occurs when a word-form with a *xaxa511* code in its PAP (e.g. *'milk')* is found.[13]

When no POV code is provided, each text must be subjected to parametric analyses. The recommended procedure is to *'read the entire text'* before translation begins, and so it should be in a CAT model.

---

[12] For example, not every referent from the *'man-made object'* POV can be *'repaired'* and while referents *'love' or 'loyalty'* can use operants *'sold', 'bought'* , *'purchased'* and thus carry the *xaxa10* code of *'object of commerce'* POV in their PAP sequence - they *do not gain normative access to other operants* from this POV operant-set.

[13] Due to space limitations, we omit parametric coding of cognitive *qualifiers* and *quantifiers* (adjectives, adverbs and numerals) etc, and their important role in the matching protocol.

Step 6 (b) Create a parametric text map where each word-form is represented by its PAP sequence.
Step 6(c) Identify word-forms with the common POVs and locate 'operants' code-paired with referents', i.e. the MWC units.

If no 'prior knowledge' of these relations is programmed, our search protocol (1) segregates operants by verb-markers and referents by noun-markers, (2) identifies operant-word appearing most frequently in the text and uses its PAP sequence to compare with (3) the PAP sequence of most frequently appearing referent word-form looking for code match with the highest cognitive value[14] and (4) repeats these procedures for the next frequent operant and referent.

A text results from *cognitive pairing of operants and referents* into "*units of thought*". Their word-forms become *a verbal equivalent of a 'unit of thought'* called a Multi-Word-Complex(MWC) when a code of their PAP sequences match. The matching code(s) of word-form components defines MWC *coordinates* as a "*unit of translation."*

Step 7(a). The text is next reformatted as a progression of MWCs in the format of *V+N, N+V, N+V+P, A+N, N+V+*P[15] each with their *POV code coordinates*.

Step 7(b) Generate a *mindprint* map as an order of progression of POV codes throughout the original text. (to be used for verification of the translation outcome)

The steps 1 to 7 replicate *'comprehension of text'* in any natural language through language-independent (metalingual) procedures and features: a POV *inventory* with POV *codes*, word-form PAPs (*code sequences)*, and MWC *coordinates* and the text becomes available for translation.

8. Translation Protocol

The matching POV code found with Step 6 procedure in the original text metalinguistically identifies MWC-1 components as a *unit of translation.*. This POV becomes our search criteria - the key to MWC-2, (a correlate of a MWC-1) in the 'to-language' database as the pre-verbal *'unit of thought'*.

Step 8(a) Identify all word-forms in the data-base with the same POV code as MWC-1.

If a perfect match to MWC-1 coordinates from the original text is not found, the use of full PAP sequences for all elements (N,V, P, A) is necessary.

Step 8(b) Identify MWC-2 components from prior step results with PAPs identical or sufficiently similar to each MWC-1 component and find a lexical correlate for each component of the MWC-2 correlate.

Step 8(c) Repeat the procedure for other MWCs of the original text.

This discussion represents step 8 tasks as a sequence that follows identification of results of text analysis from Steps 5, 6, 7, but as soon as the first PAP becomes available, the program begins extracting word-forms from the target language database that match the code sequences in the text. This process continues as Steps 8 (a) (b) (c) are processed simultaneously with Steps 5-6 until all MWC correlates are found.

Step 8(d) Organize translated text by placing MWC-2 in the order of MWC-1s appearance in the original text.

9. Disambiguation and Verification

Any CAT model needs protocols for *intra*-language disambiguation and verification of *inter*-lingual correlation of the MWCs. The following is an outline of these procedures in addition to those already in place.

Intra-language disambiguation is needed for differentiation of MWCs in the given text with nearly identical coordinates. Language economy loads many operant-words (e.g. *'give'*, *'take' 'have'*, *'be'*) with such versatility that their MWC's coordinates may appear almost indistinguishable. Likewise, many MWCs are products of POV's common to their components. As the program locates different codes in PAPs of MWC components, a semantic problem is parametrically resolved.
Step 9 (a) Consistent comparison of all PAP sequences for each component of each MWC in the text will identify at least one *unique code* but a mismatch in parameters requires disambiguation.

Differences in code sequences are more important for differentiation then complete correlations. This is the basic rule of CAT *inter*-language analysis - *parametric parameters that do not coincide in at least one instance result in rejection of both word-forms and PAP comparison continues until proper code matches are found for each of the mismatched referent and operant.*

To provide more consistent resolution, this model is set in a *learning mode*, so that prior operant-referent matches accumulate in a 'cognitive dictionary' module as a database of coded MWCs integrated with and searchable through the POV inventory. [1]

Inter-lingual correlation faces other challenges as MWCs in one language (English) may be represented by a single word-form (German) or by a larger word-form

[14] To achieve match with the highest cognitive value, the *higher code* in the operant PAP should match the *lowest code* in their referent PAP sequence.
[15] Where *N* represents a noun or its equivalent, *V* stands for a verb and its equivalent, *P* is a directional vector (pre- or post position and *A* is a qualifier-quantifier (i.e. an adjective, an adverb or a numeral.)

groupings (Russian). Thus, verification that *metalingual structure* of text remains the same is always necessary.

Step 9(b). Compare progression of POV codes of the MWCs found in original and the translated text.

Progression of MWCs (Steps 7(b) and 9(b)) represented by the order of their POV codes in the original and translated texts creates two *text mindprints*, each replicating *patterns of change in author's focus* The *relational map* of original text is then compared with the *mindprint* of the translated text. Verification is complete when the *mindprints* coincide. We ignore positional variations due to word-order rules, but if the two progressions do not match, the protocol calls for comparison of full MWCs coordinates[16] for each 'node' of 'mismatch'. If this fails, human intervention may be required. It is expected that the *learning-mode* of this model would eventually make such intervention unnecessary.

Each text *mindprints* are 'unique' in that they contain repeating patterns of POV code succession and repetition that can be *anticipated,* as one POV follows another and also reappears to form a coherent narrative with a limited number of subjects. Coupled with patterns of universal grammar [11], the metalingual parametric design holds great promise for *predictive text processing* for translation (especially of technical texts) and other tasks.

Summary

We suggest a CAT model design based on parametric coordinates of multiple word complexes (MWCs) - as linguistic analogues of a *'unit of thought'*. Step-by-step use of POV codes replicates cognitive operations with elements of 'shared reality' as "*units of translation*". This, of course, is an architectural plan; not yet a working program.

Coordinated efforts to implement its features would result in more reliable translation outcomes and may contribute to development of fully integrated translation as well as autonomous computer speech generation.

References:

[1] Gorbis, B. (2006a) "The COG: Making Sentences From Concepts*"*. In *Proceedings of the 2006 International Conference on Machine Learning; Models, Technologies & Applications*, pp. 43-49 (MLMTA 2006), CSREA Press (2006).

[2] Gorbis, B.(2006c) *"*Cognitive Dictionary: A Representation of Shared Reality*",* In Proceedings of the 50th Annual Meeting of the ISSS, - Sonoma State University, July 9th - 14th 2006, ISSN: 1999-6918.

[3] Gorbis, B. (1977) "Psycholinguistics and Generative Lexicography: A Preliminary Description*"* in "*Translator's Journal*" ("*Tetradi Perevodchika*") issue 14, "Foreign Relations Publishing House Moscow, (1977) (in Russian);

[4] Kvitash, V. (2002) "Relonics: balascopy-based systems-specific technology" in Kybernetes; 31; 1471-1480; Systems and cybernetics: new theories and applications - Part I by MCB University Press;

[5] Kvitash, V. (2002) "Relonics Languages for Systems Information Modeling" in Modeling and Simulation"; pp. 385-386; ACTA Press,; 2002

[6] Gorbis, B. (2006b) *"*Borrowing With Interest: Aspect Semantics View of Language Extension and Expansion*"*. *In 'Proceedings of the 2006 International Conference on Machine Learning; Models, Technologies & Applications*, (pp. 50-56) (MLMTA 2006), CSREA Press, (2006).

[7] Kvitash V., Gorbis B. (2006) "Relonics Properties of Living Systems". in Cybemetics and Systems 2006, Proceedings of the Eighteenth European Meeting on Cybernetics and Systems Research, Vienna, Austria, vol. 1, pp. 167-172,

[8] Kvitash, V., Swanson, G.A., Gorbis. B. (2008). "Non-Euclidity of Living Systems, System Specific Relationships and Self-Scaled Relationships of Relationships: The Evidence Beyond Obvious and Measurable." in Cybemetics and Systems 2008. Proceedings of the Nineteenth European Meeting on Cybemetics and Systems Research, Volume 1, Austrian Society for Cybemetic Studies, Austria, pp. 197-202.

[9] Gorbis, B. (2014) "Cognitive Machine Translation Model: POV and Musketeer Classes of Shared Reality" In *Proceedings of the 2014 International Conference on Intelligent Linguistic Technologies,* pp. 74 -119. (ILINTEC'14) CSREA Press (2014).

[10] Gorbis, B. (1972) "The Psycholinguistic Volume of Meaning" in "*Methods of Foreign Language Acquisition: Issues and Developments*", Kiev, (1972) (original in Ukrainian).

[11] Gorbis, B. (2005) "A Primitive Model of Metalanguage for Universal Grammar". In *Proceedings of the 2005 International Conference on Intelligent Linguistics*, (pp. 39-45). (MLMTA 2005),. CSREA Press, (2005).

[16] Full MWC coordinates in a given text include PAPs of *qualifiers, modifiers, quantifiers* etc., that play a powerful role in translation and disambiguation.

Answering Decision Questions involving an Adjective

K.H. Cheng
Computer Science Department, University of Houston, Houston, Texas, USA

Abstract – *To communicate sensibly using a natural language, it is important to understand the sentence spoken by the human. In this paper, we tackle the problem of understanding and subsequently finding the correct answer to a decision question involving an adjective. We identify and address several sub-problems: the knowledge representation problem of adjective, the word-sense disambiguation problem for adjective, the ambiguity problem resulting from the absence of the reference for comparison in the given question, the sensibility problem of the given question, and the reasoning problem in arriving at the correct answer. These problems are addressed by using the program's ability to infer relationships among categories and the use of a linear scale to represent the various adjectives describing an aspect.*

Keywords: adjective, reasoning, knowledge representation, word-sense disambiguation, natural language processing.

1 Introduction

It is desirable for a computer program to possess the capability to communicate with humans using a natural language, specifically English. To accomplish this, the grammar of English together with the intention of each grammatical part must be acquired by the program. For examples, a declaration sentence is to declare a fact, whereas a question is to ask for information. The decision question is a kind of questions asking for the confirmation of the given. The answer to the question is yes if the given agrees with existing knowledge in the knowledge base, the answer is no if the given contradicts with known knowledge, and the answer is unknown if there are missing knowledge. A decision question has the sequential grammatical structure: main verb followed by complete subject and then by subject complement, omitting the question mark punctuation to simplify our presentation. In order to understand and give the correct answer to the question, knowledge about the specific subject area being asked is also required. 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 a noun and may involve an adjective describing an aspect that is either quantitative such as "Is John a tall person" or qualitative such as "Is John happy". It may even include adverbs such as "Is John very happy". Due to the vast amount of knowledge of different subject area, our focus is on the understanding and subsequently finding the correct answer to the subset of decision questions in which its subject complement involves an adjective. The problem of understanding a decision question is to decide the sensibility of the question, which addresses the common sense problem faced by a computer program. For instance, in order to say the question is understood correctly, the program must be able to decide that the question "Is John a tall human" is sensible, but the question "Is John a long corporation" is not sensible. The problem of finding the correct answer to a decision question is about finding a "yes" or "no" answer, which depends on the availability of appropriate facts. It is equivalent to the problem of deciding the sensibility of the corresponding declarative sentence. If the answer to the decision question is true, the declaration is sensible; the declaration is not sensible if the answer is false; and the sensibility of the declaration is unknown if the answer is unknown, since there is no basis to support or refute the declaration. For instance, the answer to the question "Is John a tall human" is false if John is known to be short, and the corresponding declarative sentence is not sensible since it contradicts to a known fact. In addition, finding the correct answer may require reasoning if the known fact is descriptive. For example, a person is considered happy or not sad if known to be ecstatic.

There are several additional sub-problems that need to be addressed: the knowledge representation problem, the word-sense disambiguation (*wsd*) problem, and the missing reference problem. The knowledge representation problem of adjectives involves how to represent an adjective, what related knowledge needs to be stored, where to store them, and how to link them so that each can be accessed efficiently if given the related knowledge. It also involves how to link the various adjectives describing the same aspect so that the correct meaning of the adjective can be recognized. For example, the meaning of "Is John taller than Jack" should be the same as "Is Jack shorter than John", so the answer to both questions should be the same. Similarly, the meaning of "Is John early" should mean that the time is before the agreed-upon time, unlike punctual and late. In addition, ecstatic should be recognized as happier in intensity than happy although both adjectives have the same positive sense. An adjective may be used to describe several different aspects, e.g., long and short may describe aspects length, distance, and time duration. If the aspect is not given explicitly in the question and the adjective describes several aspects, the intended aspect of the adjective has to be resolved. The *wsd* problem is to find the

ISBN: 1-60132-438-3, CSREA Press ©

intended aspect of the adjective in a given sentence. For example, short should refer to height in "Is John short", but length in "Is the string short". The missing reference problem is to determine the reference for comparison in order to decide whether the use of the adjective in the question is justified or not. If the reference is given explicitly in the question, then the correct answer can be determined readily, e.g., in "Is Giraffe a tall animal", the reference is the "normal" height of animal. However, if it is missing in the given question, e.g., "Is John tall", the reference for comparison has to be determined before the answer can be decided.

Recently, we have developed a solution that makes use of several kinds of knowledge to answer decision questions where the subject complement is a noun representing a category [2]. It is a part of a natural language processing system [3] for a learning program system, ALPS [4]. After learning a small subset of the English language [5], the program is capable of understanding declarative sentences involving actions [6], forms-of-be verbs [7], pronouns [8], and time adverbs [9]. Our program can also tolerate some grammatical errors by designating the corresponding grammatical rules as "error-tolerable" instead of absolute rules that must be satisfied [10]. There are systems that focus on answering questions such as IBM's Watson [11,12] and PENG Light [13]. IBM's Watson has the ability to rank several plausible answers using a machine-learning framework, but it only expects Jeopardy-style questions formed as declarative statements and responds in the form of an interrogative sentence, whereas our program handles any sentences typically used during a conversation including the decision question. PENG Light uses a First-Order Predicate Calculus [14] semantic representation with annotations for syntactic information. It requires an interface to a compatible logical reasoning system, while our program handles all stages of the process internally so we can control the scope, scale, and granularity between the semantics and the reasoning algorithm. In addition, any new modifications to the parsing grammar in PENG Light needs changes to the rules for sentence generation which threatens the coherency between reading and writing, whereas our program automatically generates the role sequences [15] for writing when the parsing grammar is learned, so coherency is kept. The *wsd* problem has been heavily studied and summarized in surveys [16,17]. Some disambiguation solutions are supervised algorithms which require training for a specific knowledge domain [18,19], and some are unsupervised where training is not needed [20,21]. Supervised solutions usually work better than unsupervised ones, but they only works well in the targeted domain. Most solutions are centralized algorithms and use statistical methods in resolving ambiguity. Recently, we developed a solution to the *wsd* problem that resolves ambiguity in a distributed manner [22] rather than in a single disambiguation module. It does not need to use any statistical methods nor require any training, but it requires the program to know the various meanings of the ambiguous word. Our solution uses the program's ability of inferring the category of words to disambiguate multiple meanings of an ambiguous word in the given sentence. In this paper, we extend that solution to resolve ambiguities when an adjective has multiple meanings.

The rest of the paper is organized as follows. Section 2 reviews some basic knowledge about adjectives and presents in detail how to represent them and their related knowledge. Section 3 presents our solution on how the given decision question is understood including how multiple meanings of an adjective is disambiguated, and the process to find the correct answer to the question. Section 4 discusses some problems that still need to be resolved. Section 5 concludes the paper.

2 Adjective and its Representation

An aspect may have many different states each described by a specific adjective. These states may be in the same or opposite senses of the aspect. For instance, thin and skinny describe the appearance of the body shape in the same sense and they have similar meaning; while tall and short are in opposite senses for height. For adjectives that describe an aspect in the same sense, they may have different degrees of intensity, e.g., happy, elated, ecstatic, euphoric are all adjectives in the same joyful sense for emotion, but with different intensities. Similarly, both good and excellent are in the same sense for performance, but excellent is surely more positive than good. For some aspects, there may be a special adjective to describe the neutral state of the aspect; while some aspects do not have that neutral adjective. For examples, early, late and punctual are three adjectives for the aspect punctuality where punctual is the neutral adjective. On the other hand, thin and thick are the adjectives for thickness, and there is no neutral adjective for the aspect. For many aspects that do not have a special neutral adjective, one may use the word normal such as normal thickness, height, and weight. Usually, the positive sense of an aspect represents the preferred state of the aspect. In addition, some aspects may have an adjective that describes the most desirable situation, while some aspects do not have such optimal adjective. For example, punctual is most desirable for punctuality, but the aspect length does not have such optimal adjective. It is also possible that different adjectives may be more desirable in different usage, for instance, both a thin film and a thick wall may each be desirable in different situations. One complication in understanding an adjective is that some adjectives may be used to describe several aspects. For instance, both long and short are used in aspects such as length, distance, and time duration, while short may also be used in height as well. As a result, the aspect that it intends to describe needs to be correctly determined in order to obtain the correct understanding of the sentence. Finally, an adjective may be used in comparative or superlative sense in a sentence. When used for those senses, some adjectives have special forms, while others may need to use adverbs more, most, less, and least.

In our solution, each adjective is represented by an instance of a class called description. Each description is linked to the aspect that it describes. Since the same adjective may be used to describe many aspects, each description contains a list of aspects that the adjective describes. For adjectives that are used to describe one aspect only, this list contains one aspect. Each aspect contains an instance of a class called scale, which not only stores the various descriptions of the aspect, but also maintains a linear ordering relationship among them. A scale has two directions of sense: positive and negative; and a mid-point to represent the neutral state. For aspects that are quantitative, positive sense corresponds to using the operator > for comparison, and the operator + for arithmetic computation; whereas negative sense corresponds to operators < and –. A description is in the positive sense if the quantity that it describes is larger than the quantity described by the neutral description. For example, heavy is a positive description for the aspect weight. If the weight of John is greater than the weight of Jack, the answer to the question "Is John heavier than Jack" can be determined to be true by using the operator > associated with heavy, a description in the positive sense. In other words, the use of the positive description is correct. On the other hand, a description is a negative description if the result using the operator < is true. Light is a negative description for weight because the weight of a lighter object is less than that of a heavier object. Consequently, by simply using the comparison operator associated with the description identified from the given question, it is very easy to find out the correct answer to the given question. The arithmetic operator + allows the aspect value to move in the positive direction, while operator – allows it to move in the opposite direction. In other words, in order to get heavier, additional weight is needed, and vice versa. For aspects that are qualitative, the positions of two different descriptions only reflect their relative positions within the scale. Two descriptions are opposite in meaning if they are in different senses; but they represent similar meaning with different intensities if they are in the same sense of the scale.

The descriptions for an aspect are stored in two lists within the scale object. The first list maintains the relative ordering of the various descriptions by storing descriptions in the order from negative to neutral to positive. This list allows the correct understanding and reasoning of the relationship between two different adjectives of the same aspect. The second list uses the map in C++ library with the description name as the key. This list allows the program to locate efficiently the description using the name of the adjective. It also allows an efficient determination of whether a given name is or is not a description for the aspect of the scale. Our implementation requires the teacher to teach an initial set of a positive, a negative, and optionally a neutral description at the time a scale is taught to associate with a specific aspect. Additional descriptions for an aspect may be added to the scale with its position relative to existing descriptions being specified. In our program, a description cannot be created in isolation. In other words, each description must be associated with at least one scale, and that other descriptions for the scale must either be existed already or created concurrently. When a description is added to a scale, either the description is already known to describe some aspects or this is the first aspect it describes. If it already describes some aspects, our program simply adds the aspect to the list of aspects that the description describes. The action is done successfully if this is a new aspect; otherwise no addition is done since the knowledge is already known. If this is the first aspect for the description, a new description is created and added to the scale according to its taught sense. In addition, the aspect is added to a newly created aspect list, which is stored within the description. Finally, the specific spelling of comparative and superlative forms of an adjective may be taught, and they are stored within the description.

3 Our Solution

Given a sentence that has the grammatical structure of a decision question, our natural language parser decides that it is a decision question. A decision question role is created to store the logical objects of target and classification, where target is the subject and classification is the subject complement of the sentence. For example, given "Is John tall", John is identified as the target and tall the classification. If the classification involves an adjective, the first step of our solution is to identify the intended aspect described by the adjective. There is no ambiguity if the aspect is given explicitly in the sentence. For example, in "Is the height of John normal", the aspect is uniquely identified as height although normal may be used to describe many aspects. Similarly, the aspect is identified if the adjective is known to describe only one aspect even though the aspect may not be given explicitly. However, ambiguity exists when the adjective describes several aspects but the aspect is not given explicitly. In [20], ambiguities are classified into two categories. The first category of ambiguity results from words whose different meanings arise when the word is used in different parts-of-speech (*pos*). Ambiguity in this category can be resolved automatically during the parsing stage of a sentence based on the grammar's requirement that certain *pos* be at a specific place of a sentence. The second category of ambiguity arises when a word still has multiple meanings when used in the same *pos*. As a result, the ambiguity cannot be resolved at the parsing stage, and has to be resolved during the understanding of the thought, created by parsing the sentence. It is obvious that we are dealing with ambiguities of the second category since the adjective can be used to describe multiple aspects. These ambiguities are resolved by using context information and we will focus on using the context information available from the current sentence. Several pieces of context information available in a given sentence may be used. One is the target which corresponds to the subject of the decision question, e.g.,

John in "Is John tall". Another is the noun in the classification if it is given, e.g., mammal in "Is John a tall mammal". Still another piece is the object of comparison in the dependant clause if given, e.g., Jack in "Is John taller than Jack". Since the target is always present, our solution will use it to identify the intended aspect, and the others will be used to determine the sensibility of the question. If the target is a category, then for each aspect in the list, we will keep the aspect if the category can possess that aspect. If the target is an object, then the category of the object is used instead. Note that most aspects that an adjective describe may be possessed naturally by objects. For examples, height, emotion, and appearance can all be possessed by humans. However, the aspects described by some adjectives may not be possessed naturally, but they are universally applicable to all kinds of objects. For examples, a specific house may be considered as important since it is the childhood house of an extremely famous person; a specific pen may be important because it is used by a president to sign an important legislation. Similarly, one may discuss about the importance of many different items such as human, meeting, event, etc. It is obvious that importance is not an aspect possessed naturally by any of these items. Our solution allows these aspects or notions to be designated as universally applicable. So, besides aspects that can be possessed by the target, an aspect is also kept if it is known to be universally applicable. If only one aspect is kept, then the ambiguity is resolved. For example, height is the only aspect kept for "Is John short" since all other aspects that short can describe, such as time, distance, length, etc., are not universally applicable and cannot be possessed by human. If several aspects are kept, the question is ambiguous, and the ambiguity has to be resolved using other context information, e.g., "Is John normal" is ambiguous since normal may be used to describe multiple aspects of a human. The case that no aspect is kept will be discussed in the next paragraph on the sensibility of the question. If the given sentence is a declaration, our solution will use the object-of-interest which represents the subject of the sentence [6,7].

The next step of our solution is to complete the understanding of the given question, i.e., to decide its sensibility. Note that part of that has already been done during the process of finding the unique aspect of the involved adjective. First, the question is not sensible if the aspect is given explicitly, but the adjective does not describe the aspect, e.g., "Is the weight of John big". Second, the question is also not sensible if the adjective can describe multiple aspects, but no aspect is kept in the disambiguation process because none can be possessed by the target or universally applicable. For example, "Is Republican short" is not sensible since the target does not possess any aspect that short can describe, and none of those aspects is universally applicable. Now, assuming the aspect has been uniquely identified, several sensibility questions still need to be answered. First, if there is a noun in the classification, we need to make sure the target belongs to the noun, e.g., "Is John a tall human" is sensible since John is a human, but "Is John a tall house" is not sensible since John is not a house. Note that without an adjective, the question "Is John a house" is sensible although the answer to the question is false. Second, it is possible that the target of the decision question cannot possess the identified aspect and the aspect is not universally applicable since the aspect may not be obtained by using the target. For example, for "Is the emotion of Mount Rushmore normal", emotion is identified since it is given explicitly, but emotion is neither universally applicable nor can be possessed by a mountain, so the question is not sensible. Besides the target of the question, the object of comparison can also cause the question to be not sensible if the question contains a dependent clause. For example, "Is John taller than Microsoft" is not sensible since the corporation Microsoft cannot possess height and height is not universally applicable. The given question is sensible only if the answers to all the sensibility questions are affirmative; otherwise it is not sensible and the decision question cannot be correctly understood, and so will not be answered.

The next step is to find the correct answer to the given decision question. Our solution will decide the correct answer to the question only if the given question is sensible and all ambiguities have been resolved. Before the answer can be determined, the reference for comparison needs to be identified. Assuming no adverb is involved, an adjective may be used in three different forms: base, comparative and superlative. There are two categories of questions. The first category all have the reference given explicitly in the question and so there is no ambiguity, e.g., "Is John a heavy human", "Is John heavier than Jack", and "Is John the heaviest person". Questions in the second category have ambiguity since no reference is given, e.g., "Is the weight of John normal", "Is John heavy", "Is John the heaviest", and "Is John heavier". Our solution resolving the ambiguity caused by the missing reference is to use the following intuitive rule assumed to be agreed upon by reasonable humans. If the target is a category, then the reference is its parent category. For example, in "Is giraffe tall", the reference is animal assuming that it is the parent category of giraffe. If the target is an object, then the reference is the category of the target, e.g., in "Is John tall", the reference is understood to be human, the category of John.

The next step of our solution locates the scale object corresponding to the uniquely identified aspect. Using the scale object, we can determine the sense of the adjective within the scale. If the adjective is describing a quantitative aspect, then our solution uses the comparison operator associated with that sense to compare the aspect values of the target and the reference. If at least one aspect value is unknown, then the answer is unknown, otherwise, the answer is either true or false based on the comparison result. For example, given the decision question "Is John heavier than Jack", first heavier is converted to the base form, heavy, and marked as comparative. After the sentence has been parsed into a decision question thought, the

understanding process first uses the located description knowledge object of heavy to uniquely identify the aspect as weight. Since the answers to all sensibility questions are true, the question is determined to be sensible. The reference for comparison is also set since it is given explicitly, so the stored scale object for weight is retrieved. Since heavy is stored as a positive description, the associated comparison operator > is used to compare the weight of John with the weight of Jack.

If the adjective is describing a qualitative aspect, the above solution may also be used if an appropriate quantification scheme can be devised. However for most qualitative aspects, it is not possible to use a comparison operator to decide the answer since the aspect value cannot be quantified. Currently, we can still decide the correct answer if the aspect value of the target is known to be a specific adjective. If the known adjective is the same as the adjective in the given question, the answer is obviously true. For example, the answer to the question "Is John happy" is true if John is known to be happy. Since the scale object allows us to distinguish whether the adjective is describing the aspect in the positive or the negative sense, and stores different adjectives in a linear order according to their relative intensities; we can decide the correct answer even if the two adjectives are different. The correct answer is false if the two adjectives are not in the same sense. For example, if the emotion of John is known to be happy, so he is not sad because the two adjectives are in opposite senses. On the other hand, if both adjectives describe the aspect in the same sense; then the answer is different depending on their relative position. The correct answer is true if the adjective of the known fact is further in the sense than the adjective in the question; otherwise, the answer is false. For examples, if the emotion of John is known to be ecstatic, then he is obviously happy since ecstatic is further than happy in the positive sense of emotion; but if John is known to be happy, then he is not ecstatic since ecstatic is happier. Similarly for adjectives used in comparative form, our solution also can decide the correct answer if the aspect value of both the target and the reference are known descriptively. For example, if John is known to be happy, while Jack is known to be ecstatic; then Jack can be determined to be happier than John. On the other hand, John should be happier than Jack if the known facts are interchanged. Note that this reasoning scheme also makes sense even for quantitative aspects and when the aspect value of both the target and the reference are only known descriptively. For example, if John is known to be heavy, while Jack is known to be light. Even though the exact weights of both persons are unknown, using the above reasoning scheme can decide correctly that John is heavier than Jack, and Jack is lighter than John. However, if Joe is also known to be heavy, it is unknown whether Joe is heavier than John or vice versa without knowing the exact weight values of them. Similarly, it is also possible to decide the correct answer for some cases when one aspect value is known descriptively while the other is known quantitatively. For example, if John is known to be heavy, while the weight of Jack is known to be 4.5 kg. Assuming that 4.5 kg is much lighter than an average human, we can decide that Jack is light; and hence John should be heavier than Jack. Currently, our solution does not handle these mixed-mode cases, and they are left for future development.

4 Discussion

For the problem of finding the unique aspect, if the adjective can describe several aspects, it is possible that the ambiguity cannot be resolved by using the target of the sentence. For example, "Is John normal" is ambiguous since many aspects described by normal can be possessed by human or universally applicable such as height, weight, emotion, importance, etc. Context information outside the current sentence is needed to resolve the ambiguity correctly. If more than one aspect is possible, our solution currently produces a warning that the question is ambiguous and uses the first aspect. This temporary solution is obviously not satisfactory, and the issue will be addressed in the future. In addition, although it is perfectly sensible to ask questions involving two aspects such as "is the height of John shorter than the length of the box", such questions are also left for future development.

For the problem of deciding sensibility of the question, if there is a noun in the classification, it is possible that the noun may not possess the aspect described by the adjective. We have decided not to declare that the question is not sensible due to the following reasons. First, the question has already been decided as not sensible when the target does not belong to the noun irrespective of whether that noun possesses the aspect or not such as "Is John a tall house" or "Is human a tall corporation". Second, for questions where the target belongs to the noun, there are several cases. If both have the aspect such as "Is John a tall human", the question is obviously sensible. If both do not have the aspect such as "Is Microsoft a tall corporation", it has also been decided as not sensible since the aspect is not universal and the target does not possess the aspect. Finally if the target is known to possess the aspect such as "Is giraffe a tall animal", note that animal may or may not be known to possess height since some animals such as snake and fish do not have height aspect. If animal is not known to possess height, the question is still considered to be sensible, but the answer to the question is unknown. Since most animals possess height and if animal were known to possess height, obviously the question is sensible and the question can be answered if the "normal" height of animal is known.

If the aspect involved is quantitative, one problem of our answer is that it may not be considered to be reasonable in some cases. Specifically, even if the two aspect values differ by a very small amount, comparison will give a definitely true or false answer. However, most humans would agree that they should not be described differently. For example, if the weight of John is heavier than the

average weight of human by a very small amount, say 0.01 oz, then John is decided by our solution to be a heavy human, which is obviously an unreasonable answer. There are two possible solutions for this problem. The first solution is that instead of storing a single aspect value, a range should be taught and stored, and only aspect values outside this range will be considered to be different. This solution forces the teacher to provide a range instead of a single aspect value when teaching aspect values for a category. The second solution is to consider the aspect values to be different only when the target aspect value is outside a specific percentage of the reference aspect value. However, no matter which solution we choose, the specific ranges or percentages chosen for different aspects may or may not be appropriate since there is no absolute standard.

Our solution of using the category or the parent category as the intuitively agreeable reference for comparison may not be the most appropriate choice for the given question if more specific context information is available. For example, if the target is known as a youth or of a specific sex or belongs to a class of students, then the intended reference may be the specific subset, not the whole category of human. However, such knowledge may not be so easily detected since it is difficult to find out what is available, and to decide what to use if several of them are available. No matter what, our chosen reference may be used as the last resort when more specific context information is not available. Another problem of using the parent category is that the reference for comparison may change over time. For example, in "Is human tall", the parent may be animal initially, but as more knowledge is learned, the parent may be mammal. Much work still needs to be done in order to find a satisfactory solution to this problem.

If the aspect involved is qualitative, although our solution is able to decide the correct answer if the knowledge base has a description of that aspect, it is currently unable to use other knowledge to judge the answer. For example, if it is known that John is ecstatic, then our program can reason that John is happy since ecstatic is very happy. On the other hand, our program is currently unable to deduce that John is happy because he just graduated from college or got a promotion. The solution to this problem is left for future development. Another problem is that a linear scale may not be sufficient to capture all the adjectives of a qualitative aspect. For example, disgust, angry, and fear are all emotions that are not in the same or opposite senses of happy and sad. The adjectives of these aspects may be in a multi-dimensional scale instead of linear, so extensions of the linear scale into multi-dimension may be needed to handle such aspects. In addition, the same aspect may have different scales of adjectives for different target domain. For example, a female may be pretty or gorgeous but a male is handsome instead of pretty. To accommodate that, our system needs to allow multiple scales for the same aspect, each applicable to a specific category of targets. Furthermore, note that although it is sensible to compare the same quantitative aspect between different kinds of objects such as the height of a human and a door, it may not make sense to compare for some qualitative aspects such as "Is the rose prettier than Mary" or "Is John more important than Microsoft". However, we will treat those as sensible questions, but the answer to the question is unknown since we don't know how to compare them. One reason we treat them as sensible is to accommodate cases in which a human knows how to compare, e.g., "Is Oxygen more important than water" or "Is money more important than love". These are sensible questions although the answers to many of these questions are subjective since different persons may come up with different reasonable answers based on their own evaluation criteria.

Note that the spelling of comparative and superlative forms of many adjectives may follow some general rules. For those adjectives, instead of teaching the spelling of comparative and superlative forms for each adjective, morphology rules may be taught as part of the English grammar instead. Teaching the spelling of these special forms may only need to be done on a small number of adjectives which does not conform to any morphology rules such as better and best are the comparative and superlative forms of the adjective good.

Finally, whether the aspect is quantitative or qualitative, the aspect value of many aspects changes over time and their frequency of change vary considerably. For example, the emotion of a person may change many times each day, while the weight and height changes extremely slowly and/or minutely. It may or may not be sensible to record each of those changes. The decision to record which value and how often to record is complicated, and will be handled in the future.

5 Conclusion

We have described in this paper how to answer a decision question that involves an adjective. We identify several sub-problems and provide a solution to all of them. The knowledge representation problem is addressed by using a linear scale to maintain the relationships among the various adjectives describing the aspect. The *wsd* problem is addressed by using the subject to decide the unique aspect that the adjective is referring to in the given sentence. The missing reference problem is addressed by using an implied understanding among humans to determine the reference, but it should only be used when no additional context information is available. The sensibility of the given question that corresponds to the common sense problem faced by a machine/robot is addressed by checking that there is no contradiction to existing knowledge. Our solution uses the knowledge about a category of objects may or may not possess an aspect and the capability of the program to infer that a category is or is not a sub-category of another category. Finally, the correct answer to the given

question can be determined if the appropriate specific knowledge, quantitative or qualitative, is known by the program. The correct answer is arrived quantitatively by using the mathematical operator associated with the sense of the adjective in the linear scale, and qualitatively by performing some simple reasoning based on the positions of the involved adjectives in the linear scale that reflects their senses and relative intensities.

As pointed out in the discussion, there are still many issues that need to be addressed. Much work still needs to be done in order to understand the correct meaning of the various sentences when communicating with a human during a conversation. In particular, we are currently working on a solution to use reasons to support the use of a certain adjective of a qualitative aspect. This requires the program to learn about the various reasons to justify why the use of a certain adjective is correct. In addition, we are also working on the understanding of sentences involving the use of adverbs 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] K. Cheng. "Using Multiple Methods to Infer Classification with An Incrementally Expanding Knowledge Base"; Proceedings of the International Conference on Artificial Intelligence, 221-227, 2015.

[3] W. Faris & K. Cheng. "Performance of a Natural Language Framework"; Proceedings of the International Conference on Artificial Intelligence, 98-104, 2014.

[4] 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.

[5] K. Cheng. "An Object-Oriented Approach to Machine Learning"; Proceedings of the International Conference on Artificial Intelligence, 487-492, 2000.

[6] 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.

[7] 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.

[8] W. Faris & K. Cheng. "Understanding Pronouns"; Proceedings of the International Conference on Artificial Intelligence, 850-856, 2010.

[9] 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.

[10] W. Faris & K. Cheng. "A Grammatical-Error Tolerant Parser"; Proceedings of the International Conference on Artificial Intelligence, 711-717, 2012.

[11] D.A. Ferrucci. "This is Watson"; IBM Journal of Research and Development, *56*, 3.4, 1-1, 2012.

[12] D.C. Gondeck. "A framework for Merging and Ranking of Answers in DeepQA"; IBM Journal of Research and Development, *56*, 3.4, 14:1-14:12, 2012.

[13] 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.

[14] A. Church. "A Formulation of the Simple Theory of Types"; The Journal of Symbolic Logic, *5*, 2, 56-68, 1940.

[15] W. Faris & K. Cheng. "Generating a Sentence from a Thought"; Proceedings of the International Conference on Artificial Intelligence, 554-560, 2011.

[16] R. Navigli. "Word sense disambiguation: A survey"; ACM Computer Survey, 41, 2, Article 10, 2009.

[17] X. Zhou &and H. Han. "Survey of Word Sense Disambiguation Approaches"; Proceedings of the 18th International FLAIRS Conference, 2005.

[18] G. Escudero, L. M`arquez, & G. Rigau. "Naive Bayes and exemplar-based approaches to word sense disambiguation revisited"; Proceedings of the 14th European Conference on Artificial Intelligence, 421–425, 2006.

[19] Y. K. Lee & H.T. NG. "An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation"; Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, 41–48, 2002.

[20] O.Y. Kwong. "Word Sense Disambiguation with an Integrated Lexical Resources"; Proceedings of the NAACL WordNet and Other Lexical Resources Workshop, 2001.

[21] R. Mihalcea & D. Moldovan. "An Iterative Approach to Word Sense Disambiguation"; Proceedings of the 13th International FLAIRS Conference, 219-223, 2000.

[22] W. Faris & K. Cheng. "A Knowledge-Based Approach to Word Sense Disambiguation"; Proceedings of the International Conference on Aritificial Intelligence, 111-117, 2013.

Encoding Words into String Vectors for Word Categorization

Taeho Jo
Department of Computer and Information Communication Engineering, Hongik University, Sejong, South Korea

Abstract— *In this research, we propose the string vector based K Nearest Neighbor as the approach to the word categorization. In the previous works on the text categorization, it was successful to encode texts into string vectors, by preventing the demerits from encoding them into numerical vectors; it provides the motivation for doing this research. In this research, we encode words into string vectors instead of texts, define the semantic operation between the string vectors, and modify the K Nearest Neighbor into the string vector based version as the approach to the word categorization. As the benefits from this research, we expect the improved performance by avoiding problems in encoding texts or words into numerical vectors and more compact representations than numerical vectors. Hence, the goal of this research is to implement the word categorization system with its better performance and more compact representation of words.*

Keywords: Word Categorization, String Vector, K Nearest Neighbor

1. Introduction

Word categorization refers to the process of classifying words into a particular category or relevant categories as an instance of classification task. As its preliminary task, a list of categories is predefined and words labeled with one of the predefined categories are prepared as the sample data. The labeled words are encoded into their structured forms and the classification capacity is constructed by learning them. A novice word is encoded into its structured form, and classified into one of the predefined categories. In this research, we assume that the supervised learning algorithms are used as the approach to the word categorization, even if other types of approaches are available.

Let us mention some challenges with which this research tries to tackle. Many features are required in encoding words into numerical vectors for using the traditional classifiers in order to keep the robustness [4]. The numerical vectors which represent words tend to be sparse; zero values are usually dominant in each numerical vector [8][12]. In previous works, texts and words were encoded into tables as alternative representations to numerical vectors, but it is very expensive to compute the similarity between tables [8][12]. Hence, in this research, we try to solve the problems by encoding words into string vectors.

Let us mention what is proposed in this research as its ideas. We encode words into string vectors whose elements are text identifiers as alternative representations to numerical vectors. We define the operation on string vectors which corresponds to the cosine similarity between two numerical vectors. We modify the KNN (K Nearest Neighbor) into the string vector based version and apply it to the word categorization. Hence, in this research, the words are categorized based on their semantic relations by the KNN.

Let us mention some benefits which are expected from this research. The string vectors may become more representative representations of words than numerical vectors; the size of each string vector is smaller than that of each numerical vector. The discriminations among string vectors are much better than among numerical vectors, since the sparse distribution is not available in each string vector. We expect the better classification performance from this research by solving the problems caused by encoding words into numerical vectors. Therefore, the goal of this research is to implement the word categorization systems with the benefits.

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. In this research, based on [13], we consider the grammatical and posting relations between words and texts as well as the frequencies for defining the features of string vectors, and encode words into string vectors in this research.

3. Proposed Approach

This section is concerned with encoding words into string vectors, modifying the KNN (K Nearest Neighbor) into the string vector based version and applying it to the word categorization, and consists of the three sections. In Section 3.1, we deal with the process of encoding words into string vectors. In Section 3.2, we describe formally the similarity matrix and the semantic operation on string vectors. In Section 3.3, we do the string vector based KNN version as the approach to the word categorization. Therefore, this section is intended to describe the proposed KNN version as the word categorization tool.

3.1 Word Encoding

This section is concerned with the process of encoding words into string vectors. The three steps are involved in doing so, as illustrated in Figure 1. A single word is given as the input, and a string vector which consists of text identifiers is generated as the output. We need to prepare a corpus which is a collection of texts for encoding words. Therefore, in this section, we will describe each step of encoding the words.

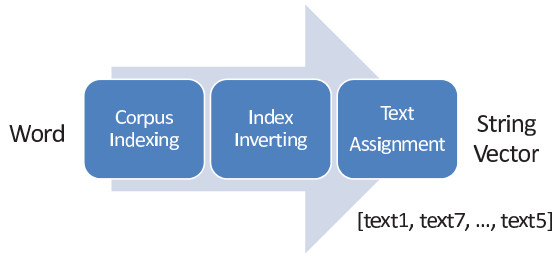


Fig. 1: Overall Process of Word Encoding

The first step of encoding words into string vectors is to index the corpus into a list of words. The texts in the corpus are concatenated into a single long string and it is tokenized into a list of tokens. Each token is transformed into its root form, using stemming rules. Among them, the stop words which are grammatical words such as propositions, conjunctions, and pronouns, irrelevant to text contents are removed for more efficiency. From the step, verbs, nouns, and adjectives are usually generated as the output.

The inverted list where each word is linked to the list of texts which include it is illustrated in Figure 2. A list of words is generated from a text collection by indexing each text. For each word, by retrieving texts which include it, the inverted list is constructed. A text and a word are associated with each other by a weight value as the relationship between them. The links of each word with a list of texts is opposite to those of each text with a list of words becomes the reason of call the list which is presented in Figure 2, inverted list.

Each word is represented into a string vector based on the inverted index which is shown in Figure 3. In this research, we define the features which are relations between texts and words as follows:

- Text identifier which has its highest frequency among the text collection
- Text identifier which has its highest TF-IDF weight among the text collection
- Text identifier which has its second highest frequency among the text collection

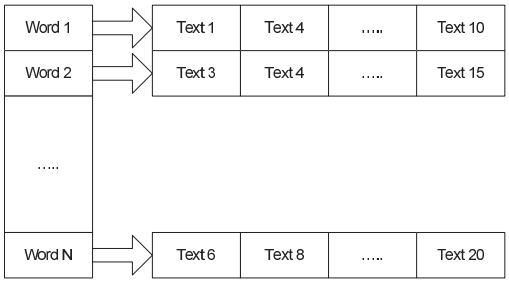


Fig. 2: The Inverted Index

- Text identifier which has its second highest TF-IDF weight among the text collection
- Text identifier which has its highest frequency in its first paragraph among text collection
- Text identifier which has its highest frequency in its last paragraph among text collection
- Text identifier which has its highest TF-IDF weight in its first paragraph among text collection
- Text identifier which has its highest TF-IDF weight in its last paragraph among text collection

We assume that each word is linked with texts including their own information: its frequencies and its weights in the linked texts and their first and last paragraphs. From the inverted index, we assign the corresponding values which are given as text identifiers to each feature. Therefore, the word is encoded into an eight dimensional string vector which consists of eight strings which indicate text identifiers.

Let us consider the differences between the word encoding and the text encoding. Elements of each string vector which represents a word are text identifiers, whereas those of one which represents a text are word. The process of encoding texts involves the link of each text to a list of words, where as that of doing words does the link of each word to a list of texts. For performing semantic similarity between string vectors, in text processing, the word similarity matrix is used as the basis, while in word processing, the text similarity matrix is used. The relations between words and texts are defined as features of strings in encoding texts and words.

3.2 String Vectors

This section is concerned with the operation on string vectors and the basis for carrying out it. It consists of two subsections and assumes that a corpus is required for performing the operation. In Section 3.2.1, we describe the process of constructing the similarity matrix from a corpus. In Section 3.2.2, we define the string vector formally and characterize the operation mathematically. Therefore, this section is intended to describe the similarity matrix and the operation on string vectors.

3.2.1 Similarity Matrix

This subsection is concerned with the similarity matrix as the basis for performing the semantic operation on string vectors. Each row and column of the similarity matrix corresponds to a text in the corpus. The similarities of all possible pairs of texts are given as normalized values between zero and one. The similarity matrix which we construct from the corpus is the $N \times N$ square matrix with symmetry elements and 1's diagonal elements. In this subsection, we will describe formally the definition and characterization of the similarity matrix.

Each entry of the similarity matrix indicates a similarity between two corresponding texts. The two documents,d_i and d_j, are indexed into two sets of words, D_i and D_j. The similarity between the two texts is computed by equation (1),

$$sim(d_i, d_j) = \frac{2|D_i \cap D_j|}{|D_i| + |D_j|} \tag{1}$$

where $|D_i|$ is the cardinality of the set, D_i. The similarity is always given as a normalized value between zero and one; if two documents are exactly same to each other, the similarity becomes 1.0 as follows:

$$sim(d_i, d_j) = \frac{2|D_i \cap D_i|}{|D_i| + |D_i|} = 1.0$$

and if two documents have no shared words, $D_i \cap D_j = \emptyset$ the similarity becomes 0.0 as follows:

$$sim(d_i, d_j) = \frac{2|D_i \cap D_j|}{|D_i| + |D_j|} = 0.0$$

The more advanced schemes of computing the similarity will be considered in next research.

From the text collection, we build $N \times N$ square matrix as follows:

$$S = \begin{pmatrix} s_{11} & s_{12} & \dots & s_{1d} \\ s_{21} & s_{22} & \dots & s_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ s_{d1} & s_{d2} & \dots & s_{dd} \end{pmatrix}.$$

N individual texts which are contained in the collection correspond to the rows and columns of the matrix. The entry, s_{ij} is computed by equation (1) as follows:

$$s_{ij} = sim(d_i, d_j)$$

The overestimation or underestimation by text lengths are prevented by the denominator in equation (1). To the number of texts, N, it costs quadratic complexity, $O(N^2)$, to build the above matrix.

Let us characterize the above similarity matrix, mathematically. Because each column and row corresponds to its same text in the diagonal positions of the matrix, the diagonal elements are always given 1.0 by equation (1).

In the off-diagonal positions of the matrix, the values are always given as normalized ones between zero and one, because of $0 \leq 2|D_i \cap D_i| \leq |D_i| + |D_j|$ from equation (2). It is proved that the similarity matrix is symmetry, as follows:

$$s_{ij} = sim(d_i, d_j) = \frac{2|D_i \cap D_j|}{|D_i| + |D_j|} = \frac{2|D_j \cap D_i|}{|D_j| + |D_i|}$$
$$= sim(d_j, d_i) = s_{ji}$$

Therefore, the matrix is characterized as the symmetry matrix which consists of the normalized values between zero and one.

The similarity matrix may be constructed automatically from a corpus. The N texts which are contained in the corpus are given as the input and each of them is indexed into a list of words. All possible pairs of texts are generated and the similarities among them are computed by equation (1). By computing them, we construct the square matrix which consists of the similarities. Once making the similarity matrix, it will be used continually as the basis for performing the operation on string vectors.

3.2.2 String Vector and Semantic Similarity

This section is concerned with the string vectors and the operation on them. A string vector consists of strings as its elements, instead of numerical values. The operation on string vectors which we define in this subsection corresponds to the cosine similarity between numerical vectors. Afterward, we characterize the operation mathematically. Therefore, in this section, we define formally the semantic similarity as the semantic operation on string vectors.

The string vector is defined as a finite ordered set of strings as follows:

$$\mathbf{str} = [str_1, str_2, ..., str_d]$$

An element in the vector, str_i indicates a text identifier which corresponds to its attribute. The number of elements of the string vector, $\mathbf{str}$ is called its dimension. In order to perform the operation on string vectors, we need to define the similarity matrix which was described in Section 3.2.1, in advance. Therefore, a string vector consists of strings, while a numerical vector does of numerical values.

We need to define the semantic operation which is called 'semantic similarity' in this research, on string vectors; it corresponds to the cosine similarity on numerical vectors. We note the two string vectors as follows:

$$\mathbf{str}_1 = [str_{11}, str_{12}, ..., str_{1d}]$$
$$\mathbf{str}_2 = [str_{21}, str_{22}, ..., str_{2d}]$$

where each element, d_{1i} and d_{21i} indicates a text identifier. The operation is defined as equation (3.2.2) as follows:

$$sim(\mathbf{str}_1, \mathbf{str}_2) = \frac{1}{d}\sum_{i=1}^{d} sim(d_{1i}, d_{2i}) \qquad (2)$$

The similarity matrix was constructed by the scheme which is described in Section 3.2.1, and the $sim(d_{1i}, d_{2i})$ is computed by looking up it in the similarity matrix. Instead of building the similarity matrix, we may compute the similarity, interactively.

The semantic similarity measure between string vectors may be characterized mathematically. The commutative law applies as follows:

$$sim(\mathbf{str}_1, \mathbf{str}_2) = \frac{1}{d}\sum_{i=1}^{d} sim(d_{1i}, d_{2i})$$

$$= \frac{1}{d}\sum_{i=1}^{k} sim(d_{2i}, d_{1i}) = sim(\mathbf{str}_2, \mathbf{str}_1)$$

If the two string vectors are exactly same, its similarity becomes 1.0 as follows:

$$\text{if } \mathbf{str}_1 = \mathbf{str}_2 \text{with } \forall_i sim(d_{1i}, d_{2i}) = 1.0$$

$$\text{then } sim(\mathbf{str}_1, \mathbf{str}_2) = \frac{1}{d}\sum_{i=1}^{d} sim(d_{1i}, d_{2i}) = \frac{d}{d} = 1.0$$

However, note that the transitive rule does not apply as follows:

$$\text{if } sim(\mathbf{str}_1, \mathbf{str}_2) = 0.0 \text{ and } sim(\mathbf{str}_2, \mathbf{str}_3) = 0.0$$

$$\text{then, not always } sim(\mathbf{str}_1, \mathbf{str}_3) = 0.0$$

We need to define the more advanced semantic operations on string vectors for modifying other machine learning algorithms. We define the update rules of weights vectors which are given as string vectors for modifying the neural networks into their string vector based versions. We develop the operations which correspond to computing mean vectors over numerical vectors, for modifying the k means algorithms. We consider the scheme of selecting representative vector among string vectors for modifying the k medoid algorithms so. We will cover the modification of other machine learning algorithms in subsequent researches.

3.3 Proposed Version of KNN

This section is concerned with the proposed KNN version as the approach to the text categorization. Raw texts are encoded into string vectors by the process which was described in Section 3.1. In this section, we attempt to the traditional KNN into the version where a string vector is given as the input data. The version is intended to improve the classification performance by avoiding problems from encoding texts into numerical vectors. Therefore, in this section, we describe the proposed KNN version in detail, together with the traditional version.

The traditional KNN version is illustrated in Figure 3. 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.

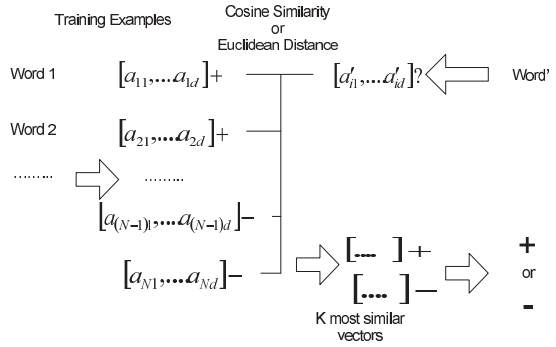


Fig. 3: The Traditional Version of KNN

Separately from the traditional one, we illustrate the classification process by the proposed version in Figure 4. The sample texts labeled with the positive or negative class are encoded into string vectors by the process described in Section 3.1. The similarity between two string vectors is computed by the scheme which was described in Section 3.2.2. Identically to the traditional version, in the proposed version, the k most similarity samples are selected, and the label of the novice one is decided by voting ones of sample entities. Because the sparse distribution in each string vector is never available inherently, the poor discriminations by sparse distribution are certainly overcome in this research.

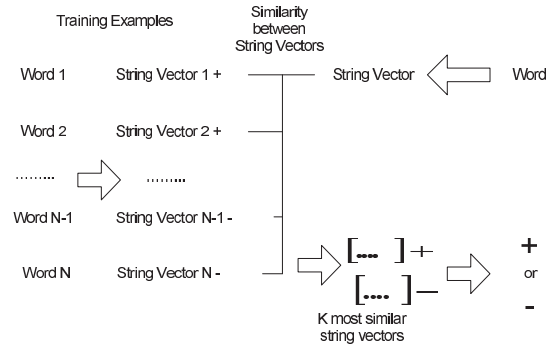


Fig. 4: The Proposed Version of KNN

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 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.

Because string vectors are characterized more symbolically than numerical vectors, it is easy to trace results from classifying items in the proposed version. It is assumed that a novice item is classified by voting the labels of its nearest neighbors. The similarity between string vectors is computed by the scheme which is described in Section 3.2.2. We may extract the similarities of individual elements of the novice string vector with those of nearest neighbors labeled with the classified category. Therefore, the semantic similarities play role of the evidence for presenting the reasons of classifying the novice one so.

4. Conclusion

Let us mention the remaining tasks for doing the further research. The proposed approach should be validated and specialized in the specific domains: medicine, engineering and economics. Other features such as grammatical and posting features may be considered for encoding words into string vectors as well as text identifiers. Other machine learning algorithms as well as the KNN may be modified into their string vector based versions. By adopting the proposed version of the KNN, we may implement the word categorization system as a real program.

5. Acknowledgement

This work was supported by 2016 Hongik University Research Fund.

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.

Graph Logic Model Framework for Predictive Linguistic Analysis

M. Charnine[1], I. Kobozeva[2], S Loesov[3] and I. Schagaev[4]
[1]Institute of Informatics Problems, 119333, Moscow, Vavilova St., 44/2, Russia
[2]Department of Theoretical and Applied Linguistics, Moscow State University, Leninskiye Gory, 1, Moscow, Russia, 119991
[3]Institute for Oriental and Classical Studies, Russian State University for the Humanities, Miusskaya Pl.6, korp. 1, 125993, GSP-3, Moscow, Russia
[4]IT-ACS Ltd, Shephall View, SG1 1RR, Stevenage, UK

Abstract - *In this paper we are trying to combine interests of authors and collaborators into one non-mutually exclusive concept and logic flow aiming to create a framework for searching of a migration of words from one cluster to another; this enables us to define a semantic shift well before it become obvious. We show how introduced graph-logic model can be applied for analysis of migration of the meaning of sentences indicating, quite often, a paradigm shifts. Using Artificial Intelligence as an example we illustrate development of AI from philosophy of mind to science, science fiction and technology, including games in science, technology, and further education. Several examples how proposed model with supportive searching framework applied in mentioned areas detecting evolving processes are presented.*

Keywords: A Graph Logic Model, Semantic search, Long-term trends, Google Books Ngram, Historical data, Predictive linguistic analysis

1 Introduction

During our previous research over migration of terms and areas in curriculum design [Bacon13] it was discovered that areas of research and knowledge in general are moving, changing, morphing and this "evolution" can be detected and even predicted. While practical use of this conception was already proved in number of papers [Bacon13], [Bacon14] and patent [Patent07] it is worth to investigate how it can be applied in linguistics, what are the limits and what it enables in terms of analysis and monitoring of migration of terminology and corresponding semantic shifts. At first we introduce a basic model – so called graph-logic model [Schagaev14], [Schagaev15] applying it for analysis of semantic shifts and migrations of terms. Every model that describes system initially using graph theory (GT) differentiates entities (nodes, vertices) and relations – (links, arcs, edges). GT model description is static; behavior aspect of the model is described separately by introducing rules and procedures allocated to nodes and links, such as activation, or algorithmic description of process to change state on a graph.

When rules for the graph path determination are applied algorithmically for every node then moves on the graph might be defined without contradiction.

The way of leaving/arriving to/from any node can be described with the help of logic operators (LO) from the basic {OR, AND, XOR}. LO might be chosen to apply to all and every node to define conditions of leaving/arriving. Say, if one applies "XOR" operator for the node leaving condition and chooses links along the graph, we are able to redraw the graph and actually mark activated links for each node, applying XOR rule for choosing only one link.

This scheme with some other restrictive conditions is used widely and known as Markov Process (MP). MP adds to the XOR rule (applied for every vertex) a normalization condition – either one adjacent link is activated or another, while probabilities – kind of "weights" - are used to "normalize" the chances of choosing one particular link (sum of probabilities to come out from a node equals 1, note that for incoming this condition does not stand).

To complete correct introduction of travel along the graph for XOR logic one has to introduce termination condition for this travel. Termination condition in MP graph as a whole is introduced by so-called aggregate state which must be reached and the condition that one of the nodes in a column is activated when the process is leaving i-th column (one of the states from there) and arriving to i+1-th column, as well as sum of probabilities in every column is equal to 1.

Again, MP works when "XOR" logic is applied to every node as a decision making rule to leave or to arrive the node. In system programming a using "XOR" operator can describe the mutually exclusive operations – i.e. concurrency, with separation of processes at the critical section entering. In

This work is partially supported by the RFFR, Grants 16-07-00756 and 15-07-06586

"classic" probability theory "XOR" logic applies when conditional probabilities are used. Models that use conditional probabilities require XOR operators for every node of process description by definition.

Let us consider another rule of leaving a node: AND logic. In this case semantically opposite graph model is introduced that describes transitions for every node at once, at the same time, instantly; when links exist from i-th node, say, to j-th, k-th,..., x-th nodes then all possible links are activated all together at the same time.

This logical condition assumes a parallel movement along the graph from any node to all connected nodes. We name this rule applied to a node as "AND" logic. Examples of the systems that are using AND logic for every node are:

- Broadcasting networking
- Salesman problem analysis,
- Diffusion processes in physics,
- Quantum effects model,
- Parallel calculations when hardware resources are unlimited, etc., etc.

Finally, "OR" logic might be applied for each and every node on the graph, assuming that only one or some links are selected, therefore flexible parallelism might be described – when it is not necessary to start everything at once.

Three natural questions arise here:

A. Can one apply various logic operators for various nodes?

B. Do conditions at output vertices dictate input conditions to other nodes?

C. How to separate logic applied? Do they affiliate to the vertex? Or Link?

It is clear that when every node has one input and one output it does not matter. On the contrary, if a graph has several links to or from its nodes it does...

Instinctive reply to question A is yes: what one does in parallel might be exclusive at the other end of the link; therefore an answer to question B is no. It is worth mentioning that conditions to leave a node (vertex) and arrive to another vertex should be attributed to edge not vertex, Fig.1

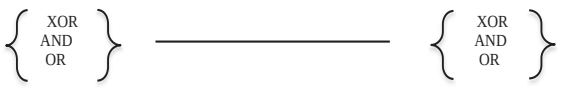


Figure 1. Logic of the edge

Their (logic operators) combination might be even more interesting: say leaving condition is "OR" but arriving condition is "AND" for each input - and we have Petri net described. To illustrate descriptive "power" of the combination of graph and logic models for behavior of graph let us draw a graph - Figure 2 - applying various logic operators for different nodes.

Let us describe Figure 2 in some details. Nodes (vertices) have output and input links. Those links that might be activated either one or another or both (node a links to node b and d are described in callout **ORo(b,d)** - that means that no order or imperative timing is required to move from vertex **a** (OR logic).

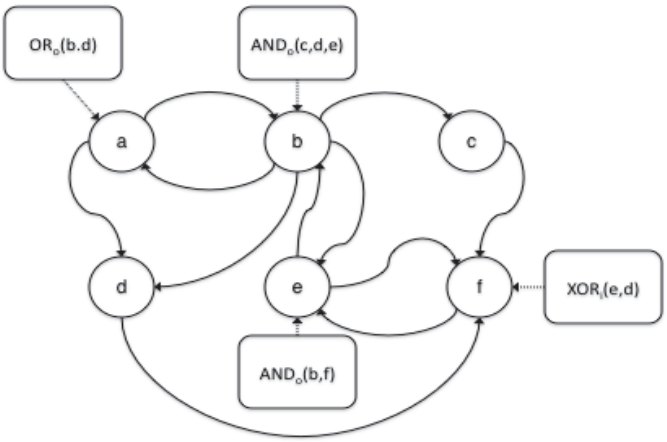


Figure 2. Various operators applied to leave and arrive

In turn, node **b** assumes parallel activation of links to nodes **d**, **e**, and **c** as it is described in special callout with operator **ANDo(c,d,e).** Note that link from node **b** to node **a** is not included in this list. Finally, input links that are required to be mutually exclusive at the node f are described by special callout as (XORi (e,d)). Again, note that incoming link from node c is not included as an input XOR operator of node f and thus might be analyzed separately.

The model proposed here in general is quite simple; it defines a graph behavior with various assumptions of leaving and arriving conditions for all vertices. This approach allows an analysis of large scale graph behavior in much more details and greater variety that the "standard" graph theory. Every node **x** of the graph such as Figure 5 thus might be described as a string:

x: AND-(j,j,k) AND+(l,m,n) OR-(p,q) OR+(r,s,t) XOR-(u,f) XOR+(g,h)

Minus "-" or "o" stand for every logic operator for output link, Plus "+" or "i" for every input link. Logic operators that have to be applied to various combinations of output links are explicitly presented in this notation.

Interestingly, leaving conditions do not obligatory match arriving ones: leaving one place together with all the rest such as parallel (AND-) calculations might be mutually exclusive at the arriving XOR – concurrency.

The addition of weights or normalized weights on the edges of GLM defines Bellman optimization model. For this we have to rename weights as rewards and penalties affiliated to each edge and input and output operators all must be "XOR".

2 GLM use

The model proposed here enables to define:

- Mutual dependence
- Structural unconditional parallelism
- Concurrency
- Transitive dependence
- Loops, including Hamilton ones
- Search of special sub-graphs that match a chosen of pattern

Graph analysis model is complete when termination condition for travelling along the graph is introduced. It is clear, that when AND, OR or XOR logic is becoming part of the model then the termination conditions might be absolutely different from the ones of the known models. Examples of termination condition might be:

A. Visiting (numeration) of all nodes, or
B. Finding the structures with:
i. Selected condition such as existence of Hamilton loop
ii. Searching of all sub graphs in the graph
iii. Searching of particular weighted sub trees
iv. Searching of selected sub-graphs to match required searching pattern
C. Quantitative termination;
D. Formation of the table about all distances for every node to every other;
E. Finding heaviest and shortest paths for each pair of nodes when they are transitively connected;
F. Defining strategies which node to choose to maximize gain along the travel from the selected node to the terminal node (horizon);
G. Development of balanced schemes (sub graphs) for each node that will define critical path along the graph travel.

Using the search of dependency in the graph of the model within complex system (like results of the search for new trends over the web, see [Bacon13],[Charnin15] might exceed dimension of the matrix 105 by 105.

When GLM applies for analysis and simulation of impacts propagation of event, say for aircraft, the model of aircraft can be described as <GT, AND,P>, where AND stands to the logic operator used and P is normalized weight of every node – say, probability of use [Patent07].

Note that there is no Markov condition to leave the node (XOR and sum of leaving probabilities is 1; Markov attributes vertex, while GLM model attributes an edge; the start and the end points of the edge might be completely different.

Instead of visiting all nodes and forming all possible outcomes a probabilistic weight of the links (edges) might be extremely useful to converge algorithm when ξ threshold is applied: $Pi \leq \xi i$, where Pi is probability of i-th link activation, ξ - empirically assigned value.

Thus for every vertex of potential dependencies of elements it is possible to form hierarchy of lists with various termination weights $\{\xi 1, \xi 2,...,\xi x\}$. The model briefly described here already has become a logic core of a "Method of active system safety" recently patented [Patent07]. Regarding monitoring and analysis of semantic change a model can be applied as follows.

2.1 GLM use in linguistic domain

An application of GLM enables us to analyze migration of the meanings, semantic shifts and observe behavior of a knowledge domain. We think that popular term of Artificial Intelligence might be interesting to analyze along its evolution.

At first logician Immanuel Kant explicitly introduced the way of thinking and defining one's own intelligence in the end of 18th century. It was done using categories that described elements' dependencies in terms of schemes "one to many", "one to one", "many to one". These categories we extend by GLM that makes logic operators map to natural language elements as Table 1 illustrates. Using GLM it is easy to introduce and analyze the leaving and arriving conditions in word dependencies [Bacon13]:

Table 1. Language statements and Logic Operators

Statement in language	Logic operator
One of	XOR
Maybe, some	OR
Always	AND
Never	NOP
Some	Combination of XOR, OR, AND

The meaning of the word can be modified in syntactic context: the word "bin" might be a noun (*a bin*) or a verb (*to bin*), as well as *Google* or *Hoover – googling* and *hoovering* indicate the action of using the companies made systems, thus noun became a verb in proper contexts while staying a noun in other contexts. A word can change its positive connotations into negative and then again to positive as Russian *mecenat* 'Maecenas'. Subtle historical semantic

shifts of the key words of different cultures are well represented in ethno-linguistic studies [Wierzbicka97].

Then defining terms and observing their shift we can:

- In the example above describe a shift from a noun to a verb;
- Change the imperativeness of use making another table hierarchy of will;
- Create new dependencies, reflect shifts of meanings or actions (from guessing and pure research to industry and detect unavoidable actions instead of possible ones);
- Do fine-tuning of curriculum - in other words we can check efficiency of advertising, detect technological revolution before it becomes visible even for authors;
- Detect direction of change where and when nobody actually sees that.

To summarize these bullet points: we will be able to see the changes of the world BEFORE the world itself realizes it.

The concept of language transformation and change of lexical meaning as a reflection of national, economic, territorial and ethnic changes as well as the change of the form of words is an established fact in historical linguistics.

See below an example of such changes in Babylonian branch of Akkadian languages – Table 2.

Table 2 Evolution of meanings in Babylonian languages

English	Old Babylonian	Middle Babylonian
so, thus	Kīam	akanna
All	Kalûm	gabbu
good, beautiful	Damqum	Banû
one another	aḫum aḫam	aḫāmiš
Urgently	arḫiš	ḫamutta
Work	Šiprum	dullum
towards (a person)	ana ṣēr	ana muhhi
chez (French)	itti, ina mah(a)r-	itu-
there is no	ul ibašši	Yānu
dispatch	ṭarādum	šūṣû
when (conjunction)	Inūma	kī
sunrise	ṣīt šamšim	napāḫ šamši
return	Târum	naḫāsum
Neglect	egûm,aham nadûm	mekû
suit, fit	redûm, naṭûm	alākum

Bodies of languages compared here are not equal Old Babylonia is extremely large, Middle Babylonian is much smaller, A comparison was made on a basis of what is possible in the 1800-1700 BC for Old Babylonian and 1350-1200 for Middle Babylonian [Loesov14].

The handling of textual data and extraction of required properties can be done nowadays using a GLM-based framework in combination with Google Ngram tool.

That help to see a growth of "neighbors" for keywords and therefore predict further shifts of their semantics and changes of knowledge in the corresponding knowledge domain.

2.2 GLM use in "macro linguistic"

Another approach of using GLM stretches up to "migration of ideas" [Charnin15], [Jacob13]. In this domain we can divide knowledge areas into almost non-overlapping segments allocating to them macro-nodes of GLM.

Then links that represent dependencies might be considered as weights, normalized weights or probabilities of changes/migration.

This approach enables us to trace a trend. E. g. the manifestation of new concepts in the Internet can cause their reflection in scientific papers and later in products, industrial systems or other developments.

This is also useful as an instrument to evaluate probability and delays between the appearance of an idea and its reappearance translated into some other language. Say, an appearance of "artificial intelligence" as a term in English required several years to reappear in Russian scientific works.

Probabilities and delays in this process are traceable and therefore define handicap or advances of various scientific groups.

"An idea" can be represented as a group of semantically close phrases and terms that are used similar to such methods as LDA [Blei2003]. Ideas can be organized or be part of hierarchy, elements of which can be again vertices or nodes of GLM.

Figure 3 illustrates main levels of hierarchy. Analyzing link's strength, frequency, occurrence area (academic or industrial, political) might help to evaluate where the world goes, and to do it automatically; when the program itself searches terms and creates nodes and links drawing a map that is a semantic map of research evolution.

As for the immediate application or implementation aspect, one can find dependencies between terms using Google Books Ngram [Roth13];

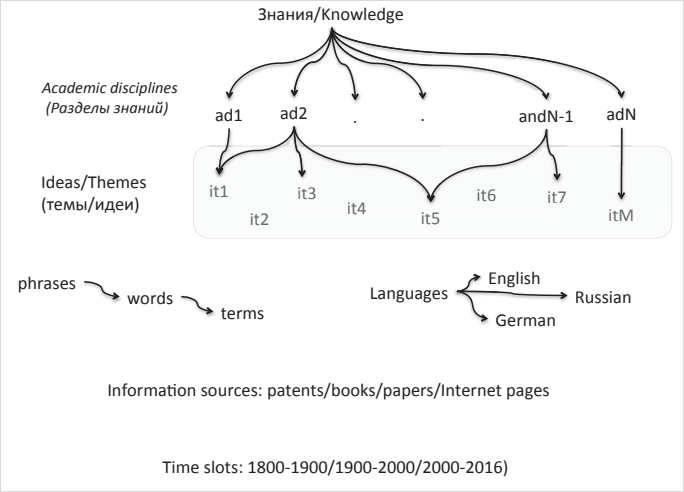


Figure 3. Hierarchy of knowledge

Figure 4 presents an example of dependency between appearances of "can machine think" in English after Turing's book [Turing1950] and the question "can machine think" in Russian scientific literature Russian that appeared after 1960.

3 Migration of term *Artificial Intelligence*: an example

Migration of terms from English books into Russian took several years in 1940-1960.

For example, the term "Artificial Intelligence" became used in English language books since 1957, and word-by-word translated as "искуственный интеллект" it migrated into Russian scientific literature in 1965, as Books NGram indicates, Figure 5.

Similar delays were observed in migration of a word combination "machines think" (from 1944 up to 1955) or a complex term "Turing test" (1960 up to 1966), Figures 6,7.

It is highly likely that initially the term *artificial intelligence* was not defined explicitly, denoting a kind of new phenomena related to computer science and computer technology.

It is worth to investigate further some other combinations of words closely related to the same area which did not become widely accepted terms *machine intelligence*, *intelligent machines*, *intelligent machinery* and Russian equivalents: *думающие машины*, *машинный интеллект*.

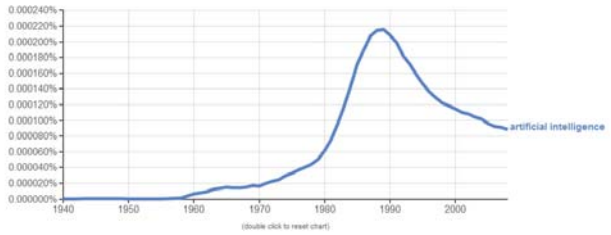


Figure 4. Trend of the term "Artificial Intelligence"

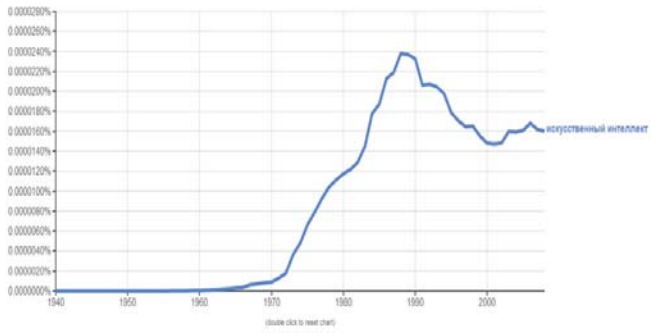


Figure 5. Trend of the term "искусственный интеллект"

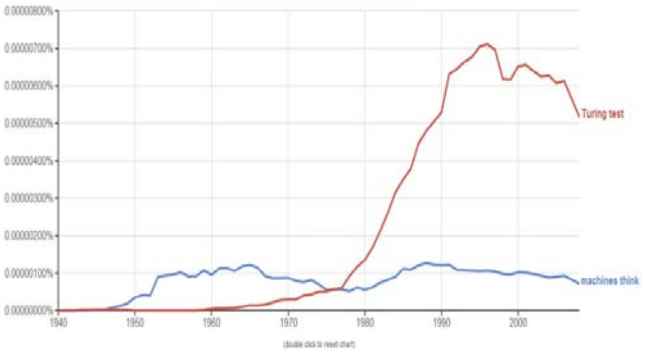


Figure 6. Trends of "Machine thinks" and "Turing test"

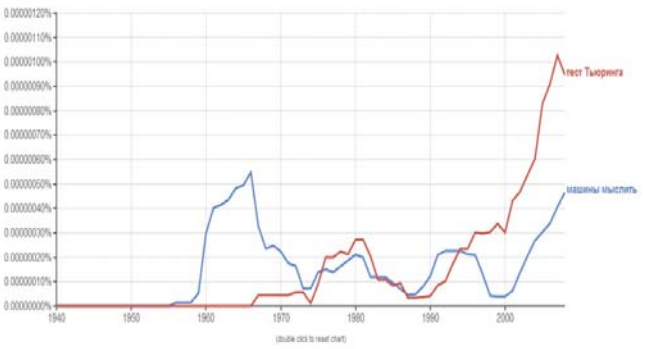


Figure 7 Use of words "машины мыслить" , "тест Тьюринга"

The biggest impact on frequency of use for a combination of words "machine thinks" was no doubt caused by Alan

Turing's paper "Can the Machine Think?" [Turing1956] initially published under the title "Computing Machinery and Intelligence" in 1950 [Turing1950]. This paper was translated into Russian in 1960 and the growth of frequency of Russian translational equivalent of "machine thinks" indicates this. John McCarthy who coined the term Artificial Intelligence (AI) in 1955, defined it as "the science and engineering of making intelligent machines". There is no strong evidence though that he produced a reasonable definition of an "intelligent machine" that is required for the explicit definition of artificial intelligence.

In 1955 the term AI appeared, that was used before as a free combination of words, associated with variety of meanings. Before this *artificial intelligence* was used in books in the context of philosophy, occurring in philosophical arguments about the nature of mind and ethics of creating artificial beings. Then it was not clear whether it was possible to create a complete working AI. It must have been highly unlikely as Turing test was not known then and therefore criteria of artificial intellect were vague. Surprisingly, there is no drastic change in this direction – while much more developed programs are appearing and passing various tests including the Turing test, there is still no clear understanding of what is possible and what is not in the domain of AI and to what extent AI can match human intelligence.

Since I. Kant [Kant34], followed by philosophers/novelists [Capek21], [Asimov50], [Asimov53], [Lukas65], [Minskly55],[Minsky58], [Chomsky12] a discussion of understanding, of how brain becomes mind and how we can make a schema of this process was primarily conceptual, virtual.

Philosophical arguments about the nature of mind and the ethics of creating artificial beings were pretty intensive. Initially, the term AI was used together with such words as *brain*(7), *consciousness*(5), *thinking machine*(3), *human mind*(2), *philosophical*(1), *philosophers*(1). Here in brackets we count number of uses of these words in Turing's paper "Can the Machine Think?" [Turing1956]. All these words belong to philosophic context and appeared well before 1800 – as Google Ngram demonstrates (Figure 8).

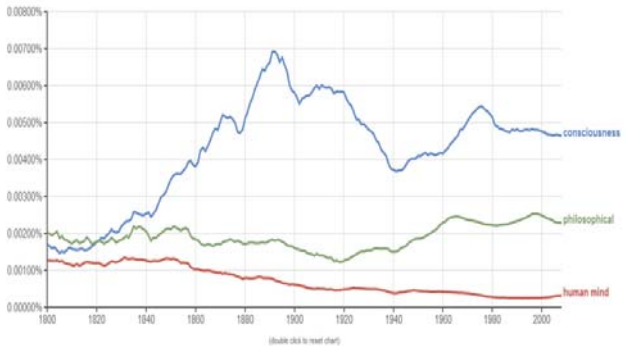


Figure 8. Trends of "consciousness", "human mind" and philosophical"

To some extent human ability of creation of an artificial mind that is equal to human one is far from clear, in spite of recent €2bn projects funded by EC and similar one in USA. This might be explained by the absence of adequate technology and math methods to attempt any implementation.

With appearance of automata theory and later transistors Information Computer Technology (ICT) emerged and the term AI migrated to the area of engineering and became associated with development of simulation systems as well as play of Chess and Go.

Other terms, which are now used most intensively in the context with AI, such as *robotics* (1970), *computer vision* (1974), *natural language processing* (1964) and speech recognition (1957) appeared after 1957. Here brackets show years of appearance of these terms – according to Google Books Ngram.

The strength of associative relation and relative frequency of AI in context with other terms (according to data of Google Search Engine) can be ordered as Table 3 shows.

It is clear that nowadays AI is considered as a technical term and is used in technical environment with terms such as "speech recognition", "robotics", "computer vision", "natural language processing").

Philosophical context is still present but appears less often: ("brain", "human mind", "philosophers", "philosophical", "consciousness").

Therefore paradigm shift from science to implementation is taking place, while philosophical issues concerning this concept are not completely resolved.

Table 3. AI and related terms

Terms used wih AI	Number of appearances	Freq-cy
"AI" "speech recognition"	1,670,000 / 6,020,000	= 0.277
"AI" "robotics"	16,500,000 / 65,700,000	= 0.251
"AI" "computer vision"	1,470,000 / 8,580,000	= 0.171
"AI" "natural language processing"	543,000/4,600,000	= 0.118
"AI" "brain"	19,300,000/532,000,000	0.0363
"AI" "human mind"	254,000 / 8,140,000	0.0312
"AI" "philosophers"	356,000 / 21,700,000	0.0164
"AI" "philosophical"	516,000 / 55,700,000	0.00924
"AI" "consciousness"	591,000 / 84,600,000	0.00698

4 Conclusion and next steps:

- Google searching machine is a tool that provides a raw data. Requirements and ways to interpret them, creation of the next iteration of search has to be formalized.
- Regretfully our subjectivism limits rigorous monitoring of trends in the languages and knowledge areas (the concept of an idea might be not really strong). Introduction of Graph Logic Model as a framework for autonomous monitoring of the knowledge provides the most powerful framework for analysis of objects in their dynamic.
- This paper illustrates it using linguistic knowledge. The growth of the amount of neighborhood terms means that the subject evolves. And vice versa. Applied in combination with Google tool Graph Logic enables a self-tuning framework up to automatic generation of scenarios for next search.
- This "self-adjustments" might help to modify, say, university curriculum, shifting some elements to skills and some others to advanced studies, or timely delete modules or disciplines that became obsolete.
- We propose an approach of using a graph logic model for linguistic methods of analysis showing "where the meaning goes".
- Shown that evolution of semantics in various domains, interaction of terms and change of their internal properties might be done using automatic searching tool "supervised" by proposed graph logic model.
- As immediate application of proposed approach we see:
  - Evolving scheme of application of GLM as a part of searching framework with reflections of knowledge in the chosen domain
  - Visualization support for terms migration and domain change in the way of how it works with the subject.
- Using term "Artificial Intelligence" evolution we demonstrated that our approach works.

5 References

[Bacon13] Bacon E, Schagaev I, Charnine M, Foggie R, Kravtsov G, Kirk B., "Web-enhanced design of university curricular", http://worldcomp-proceedings.com/proc/p2013/FEC4025.pdf

[Blei2003] D. M. Blei, A. Y. Ng, and M. I. Jordan. "Latent Dirichlet allocation", Journal of Machine Learning Research, 3:993–1022, Mar. 2003.

[Bacon14] "Essential Knowledge Aggregation, Delivery, and Assessment", E-learn Volume 2014 Issue 5, 05-01-2014 Article No. 1

[Capek21] Capek K., "R.U.R" Play, 1921

[Chomsky12] Chomsky N "Where AI went wrong", The Atlantic, Nov 1, 2012, http://www.theatlantic.com/technology/archive/2012/11/noam-chomsky-on-where-artificial-intelligence-went-wrong/261637/

[Charnin15] M.Charnine, S.Klimenko, "Measuring of "Idea-based" Influence of Scientific Papers", Proceedings of the 2015 International Conference on Information Science and Security (ICISS 2015), December 14-16, 2015, Seoul, South Korea, pp.160-164.

[Jacob13] Jacob S., Jerry A. 2013. "Receptivity Curves: Educational Research and the Flow of Ideas: Expanded Version", Population Studies Center, University of Pennsylvania, PSC Working Paper Series, PSC 13-10, http://repository.upenn.edu/psc_working_papers/50

[Kant34] Kant I. Critique of Pure Reason, Everyman, 1934, ISBN 0-460-87358-X

[Loesov14] Aramaic languages http://ivka.rsuh.ru/binary/78809_46.1236189872.35672.pdf

[Lukas65] Lucas, J. R. 1965. "Minds, Machines, and Gödel." Philosophy 36: 112-127.

[Minsly55] McCarthy, J., M. L. Minsky, N. Rochester, C. E. Shannon. 1955. "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence."

[Minsky68] Minsky, M. 1968, "Semantic Information Processing", Cambridge, MA: MIT Press.

[Patent07] Patent GB 0707057.6 http://www.it-acs.co.uk/files/Grant_for_a_patent.PDF

[Roth13] Roth, Steffen, "The fairly good economy. Testing the economization of society hypothesis against a Google ngram view of trends in functional differentiation (1800-2000)", Journal of Applied Business Research, 2013, Vol. 29 No. 5, pp. 1495-1500

[Schagaev14] Schagaev I, "Control Operators Vs Graph Logic Model", Proc of Int-l Conference on Foundation of Computer Science, FCS14, pp180-182,

[Schagaev15] Schagaev I, Kaegi-Traschel T Software Design for Resilient Computers Springer 2015 ISBN 978-3-319-29463-6

[Turing1950] A. M. Turing "Computing Machinery and Intelligence", Mind, New Series, Vol. 59, No. 236 (Oct., 1950), pp. 433-460.

[Turing1956] A. M. Turing, "Can the machine think?", The World of Mathematics, 1956, vol. 4, Ed. J R Newman (New York: Simon and Schuster), pp. 2099-2123.

[Wierzbicka1997] A. Wierzbicka, " Understanding Cultures Through Their Key Words: English, Russian, Polish, German, and Japanese", 1997, N.Y., Oxford Univ. Press, pp. 1-317.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

OPTIMIZATION ALGORITHMS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Harris's Hawk Multi-Objective Optimizer for Reference Point Problems

A. Sandra DeBruyne[1], B. Devinder Kaur[2]

[1,2]Electrical Engineering and Computer Science Department, University of Toledo, Toledo, OH, USA

Abstract - *This paper proposes a novel approach called the Harris's Hawk Multi-Objective Optimizer (HHMO), which is used for solving reference point multi-objective problems. This algorithm is based on the grey wolf multi-objective optimization algorithm and motivated by the cooperative hunting behaviors of the Harris's Hawk. These hawks are known as the wolf pack of the sky. The hunting party consists of a group of lookout hawks perched above the environment to identify prey and direct a second group of ground hawks towards potential prey, who encircle, flush out and attack the prey. By mathematical modeling these behaviors optimal solution sets around desired reference points for multi-objective problems can be found. HHMO was successful in locating clusters of solutions at or near these desired reference points. This new HHMO algorithm out preformed the previously tested Predator Prey algorithms both in terms of fitness achievement and shorter convergence time.*

Keywords: Reference Point Multi-Objective Optimization, Multi-Objective Grey Wolf Optimizer (GWMO), Meta-heuristics, Artificial Intelligence, Biologically Inspired Computing, Evolutionary Algorithms

1 Introduction

Multi-objective optimization algorithms have been a subject of study for the last 20 years or so. Many algorithms have been developed that effectively return evenly dispersed solution sets to the decision makers at or near the ideal Pareto front. One of the major hurdles these algorithms have to overcome is finding effective solution sets when there are a large number of objectives. First, four or more objectives are difficult to plot and visualize. Secondly, the number of solutions increases exponentially as the number of objectives increases. Thirdly, a large number of objectives correspond to a large number of non-dominated solutions being generated quickly as the algorithm iterates making it difficult for the algorithm to effectively select an ideal solution set along the Pareto front [1]. The concept of reference point based multi-objective optimization was developed to aid in solving problems with a large number of objectives. The decision makers are asked to select one or more reference points in the decision space prior to executing the optimization algorithm. The reference points are merely suggestions of desired regions to investigate during optimization. The decision makers are not interested in finding that specific solution but instead finding a solution set at or near the reference points. The algorithm then attempts to find the best non-dominated solutions at or near those points [1].

Many algorithms have been recently developed that mathematically model the hunting and foraging behaviors of animals. Several algorithms based on particle swarm optimization (PSO) techniques have been developed. For example Yang et al developed multi-objective algorithms that mathematically model bat echolocation (MOBA) and the cuckoo's search for nesting sites (CS) [2, 3]. The authors have developed a-priroi optimization algorithm (GWMO) and Mirjalili et al developed a-posteriori optimization algorithms that mathematically model grey wolf hunting (MOGWO) and moths attraction to a flame (MFO) [4, 5]. Algorithms of this type show that by mimicking animal behaviors directly corresponds to searching for the Pareto optimal solution sets in multi-objective optimization.

In this paper proposes a multi-objective optimization algorithm for reference point based problems by enhancing the grey wolf multi-objective optimization (GWMO) with the coordinated hunting behaviors of the Harris's Hawk. In research and testing HHMO required dramatically less iterations to return clusters of solution sets near the reference points as compared to Deb's multi-objective predator prey algorithms on the same problem sets [12]. Another advantage of HHMO is that the algorithm can be applied to problems with four or more objectives; Deb's predator prey algorithms were limited to finding solutions for problems with less than three objectives.

This paper is organized into six sections. Section 2 contains the explanation of Harris's Hawk cooperative hunting behaviors. Section 3 defines reference point based multi-objective optimization. Section 4 proposes the new reference point based multi-objective optimization based on Harris's Hawks (HHMO) algorithm. Section 5 and 6 outlines the comparative analysis of the results and conclusions.

2 Harris's Hawk

In this section the hunting behavior of Harris's Hawk is described. The Harris's Hawk (Parabuteo Unicinctus) is a

medium-large raptor that lives primarily in the deserts of south western United States and Mexico. The unique feature of this raptor species is its tendency to live and hunt in large social groups. All other raptor species are solitary animals. Attributed to the scarcity of food in the desert environments that they live in, the hawks have developed complex cooperative hunting and communication techniques to locate, encircle, flush out and ultimately attack potential prey. Their diet includes many of the large rabbits, lizards and small birds that live in the desert. During seasons of food scarcity these groups of hawks have also been known to go after prey that is larger than them. In order to successfully catch their fast moving prey, the hawks must coordinate and orchestrate a group attack. Biologists Bednarz and Coulson have spent numerous hours in the deserts studying and documenting the behaviors of these hawks, one such field study is shown in Fig. 2 [8, 10].

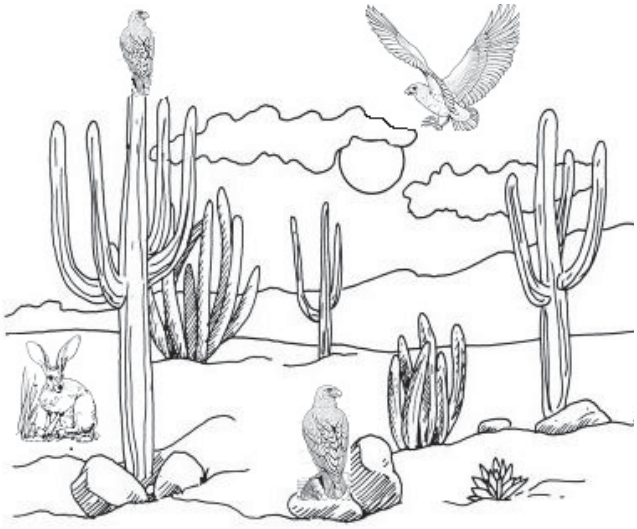


Figure 1. Harris's Hawk environment [6]

The biologists observed that the hawks' day begins by assembling all members of the hunting party in the pre-dawn light of the desert on one or more cacti near the nesting site. Hawks were heard communicating with each other by a series of chirps before beginning the hunt [8, 10]. Hawks divide themselves into two groups. The lookout hawks perch on cacti, high trees or poles to look out for potential prey and then coordinate with ground hawks to catch the prey. The lookout hawks in groups of one to three fly out on a series of short flights ranging between 60-200m. The hawks perch on of neighboring cacti to look for potential prey on the ground. When prey is identified, lookouts communicate its location to the remaining hawks by a series of chirping calls. The remaining hawks fly from their perches diving to the ground near the communicated prey location. Due to the dense brush that grows in areas of the desert floor, the prey is not always in plain sight. The ground hunters must coordinate their efforts in an attempt to flush out the prey. Several hawks will encircle the brush where the prey is hiding and take turns diving into the brush to force the prey out. Once out in the open hawks pursue the prey by hopping and running along the ground. If the prey escapes lookout hawks fly to another perch and the process begins again. Lookout hawks and ground hunters can exchange positions over time to conserve energy [8]. A successful capture leads to a meal that is shared by all hawks in the hunting party. Remaining carcasses if they can be carried are flown back to the nesting site to feed the young hawks; otherwise a few hawks are left to guard the carcasses while portions of carcasses are transported back.

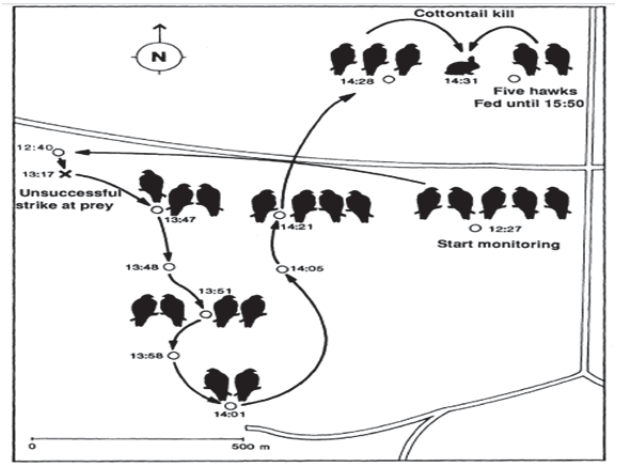


Figure 2. Field study of the position of Harris's Hawk in the desert while capturing cottontail rabbits [10]

3 Reference Point Based Multi-Objective Optimization

This section presents the problem definition for reference point multi-objective optimization problems. Reference point based multi-objective problems by definition have two or more objective functions $f_m(x_i)$, subject to multiple equality $g_j(x_i)$ and inequality $h_k(x_i)$ constraints, inside an objective decision space D defined by upper (x_i^U) and lower (x_i^L) boundaries on the input variables x_i. Instead of returning a solution set evenly distributed across the Pareto front as desired by standard multi-objective optimization, reference point based optimization seeks to return a solution set clustered near the provided reference points, defined by $v_p(x_i)$. These reference points, sometimes referred to as the fuzzy preferences for the solution [15], are used to drive the optimization towards these goals, shown in equation set (1).

$$\begin{aligned} &Minimize: && f_m(x_i) && m = 1,2,\dots,M \\ &Subject\ to: && g_j(x_i) = 0 && j = 1,2,\dots,J \\ & && h_k(x_i) \geq 0 && k = 1,2,\dots,K \\ & && v_p = [(a_1, a_2, \dots, a_i)] && p = 1,2,\dots,P \end{aligned}$$

$$\begin{aligned} &Solution\ vector: && x = (x_1, x_2, \dots x_i) && i = 1,2,\dots,N \\ &Decision\ Space\ D: && x_i^L \leq x_i \leq x_i^U \end{aligned} \quad (1)$$

Optimization begins by generating a random sample set of *N* variables x_i within the decision space *D*; next it checks that all the constraints $h_k(x_i)$ and $g_j(x_i)$ are satisfied; and lastly makes decisions on whether to maintain, modify or reject each variable x_i in the set *N*. To make these decisions solutions are chosen based on their distance to the reference points using the norm $\|\circ\|$ or magnitude of the vector v_p from the reference point to each solution $f_m(x_i)$ in the solution set. The verification and decision making steps are repeated until a desired outcome is either reached or found to be unattainable in the time allowed.

$$f_1(x) = \sqrt{1 + x^2}$$
$$f_2(x) = 4 + 2\sqrt{1 + (x-1)^2}$$
$$v_1 = (1.06,\ 6.13)$$
$$\{x \in D \mid -0.5 \le x \le 1.5\} \qquad (2)$$

As illustrated in Fig. 3, two sample objective functions given in equation (2) are plotted in decision space *D*. The minimum value for $f_1(x)$ occurs at x = 0 and the minimum value for $f_2(x)$ occurs at x = 1. The optimal solution *x* that minimizes both objective functions will fall somewhere in the interval $x \in [0,1]$. The decision makers have provided an ideal reference point v_1. Fig. 4 shows the graph of the resulting solution set, and a curve representing the Pareto front. Fig. 4 also shows the circle plotted around the reference point which represents the area of optimal solutions near that reference point. The algorithm seeks to returns a solution set that lies along the Pareto front near the given reference point as shown in Fig. 4.

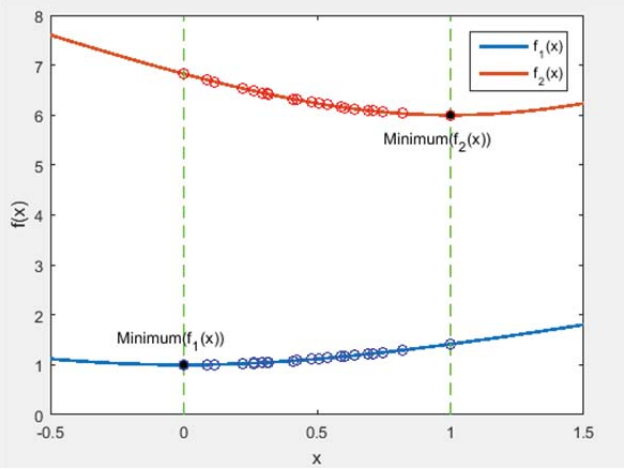


Figure 3. Sample objective function (2) graph with selected x values.

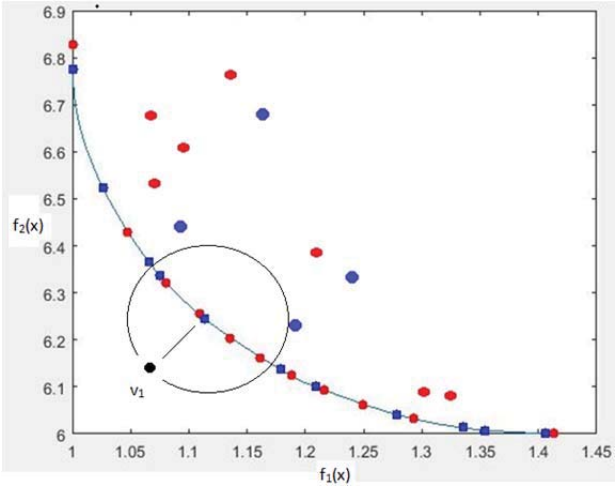


Figure 4. Sample objective function (2) solution set and Pareto front with reference point v_1

4 Harris's Hawk Multi- Objective Optimization (HHMO)

This section describes how we mathematical modeled the Harris's Hawk hunting behaviors to create a novel multi-objective optimization algorithm (HHMO) for reference point based problems that returns a solution set clustered about given reference points. HHMO builds upon our previously developed grey wolf multi-objective optimization algorithm GWMO since the searching for the prey by hawks closely resembles the encircling and attacking behavior of a pack of grey wolves. Harris's Hawks hunt in two distinct groups where wolves hunt in one group. The first group of hawks perch high above the environment to locate potential prey. These lookout hawks represent the reference point decision process. The second group of hawks waits on the ground to flush out and attack the prey. These ground hawks represent the search for optimal solution sets. Our previously developed algorithm GWMO was developed to return a diverse solution set of evenly spaced non-dominated optimal solutions along the Pareto front. HHMO refines and enhances this algorithm to return clusters of solutions sets near the specified reference points for multi-objective problems.

4.1 Grey Wolf Multi-Objective Optimization

The Grey Wolf Multi-Objective Optimization algorithm (GWMO) outlined in this section is based on the hunting behavior of a pack of grey wolves [16]. There is a social hierarchy in a pack of grey wolves. The dominant wolf in charge of the pack is known as the alpha wolf. The second and third place wolves are known as beta and delta wolves. The wolf pack is not a dictatorship and the alpha wolf is assisted by subordinate beta and delta wolves in the process of decision making. The remaining wolves in the pack are referred to as the gamma wolves. Communication throughout the entire pack provides vital information to the

alpha, beta and delta wolves. This information is vital for directing the pack to locations for tracking, encircling and attacking potential prey.

The GWMO algorithm starts by generating initial random values X_i in the decision space D, representing the initial positions of the wolves. Distance to the prey X_P defines the GWMO fitness function for selecting the best non-dominated solutions. All distances from the prey D_γ, are calculated and the top three shortest distances are selected and labeled as the positions of alpha, beta, and delta wolves $X_\alpha, X_\beta, X_\delta$.

The adjustment factors A_i which represents searching for the prey and C_i which represents the encircling of the prey are computed for the top three wolves with equation set (3). These factors ensure that the algorithm does not settle at a local minimum. Each of the top three positions of wolves $X_\alpha, X_\beta, X_\delta$ has a unique forward versus backward search movement in the decision space defined by $A_\alpha, A_\beta, A_\delta$ which randomly fluctuates between [-2a, 2a]. As variable 'a' becomes increasingly smaller the radius of search decreases over time. In nature the wolf pack will eventually discontinue its wide search pattern and collectively approach the best selected prey for attack shown in equation set (3).

$$\begin{gathered} A_i = 2a * r_1 - a \\ a = (2 - 1 * (2 / Iteration)) \\ r_1 = random\ number \\ C_i = 2 * r_2 \\ r_2 = random\ number \end{gathered} \tag{3}$$

The algorithm next applies factors A_i and C_i to modify the top three distances from the prey for alpha, beta and delta wolves D_α, D_β, D_δ. An average of the three top positions X_1, X_2, X_3 is computed which is used to modify the positons of all the gamma wolves in the pack as shown in equation set (4).

$$\begin{gathered} D_\gamma = \left| X_\gamma - X_P \right|;\ D_\alpha = |C_\alpha * X_\alpha - X_P|; \\ D_\beta = \left|C_\beta * X_\beta - X_P\right|;\ D_\delta = |C_\delta * X_\delta - X_P| \\ X_1 = X_\alpha - A_\alpha * D_\alpha;\ X_2 = X_\beta - A_\beta * D_\beta \\ X_3 = X_\delta - A_\delta * D_\delta \\ X_\gamma(t+1) = X_\gamma(t) + \frac{X_1 + X_2 + X_3}{3} \end{gathered} \tag{4}$$

The GWMO algorithm iterates until a desired solution set of non-dominated solutions is reached. It is important to note that alpha, beta, and delta positions are reevaluated and reselected during each iteration.

4.2 Harris's Hawk Multi-Objective Optimization

The Harris's Hawk Multi-Objective Optimization algorithm proposed in this section, builds upon the previously defined GWMO algorithm to guide the solution set selection process towards specific regions of the decision space as requested by the reference points. Initial reference points $v_p(x_i)$ are defined and initial random values X_i are generated in the objective decision space D, representing the initial positions of the hawks. Distance to the prey X_P defines the fitness function for selecting the best non-dominated solutions. Multiple preys X_P^i are maintained throughout the search, one for each of the reference points. Initially distances D_γ from all hawks to each of the prey X_P^i are calculated. The hawks are then divided into groups based on their relative distance to the closest prey. The top three shortest distances in each group are selected and labeled as the positions of alpha, beta, and delta hawks $X_\alpha^i, X_\beta^i, X_\delta^i$. The positions of the hawks in each group are modified based on equations in set (4) using each set of alpha, beta and delta hawk positions for that group. The HHMO algorithm is outlined in Fig. 5.

Given the set of objective functions
$f_1(x_i),\ f_2(x_i), \dots,\ f_m(x_i)\ \ m = 1,2, \dots, M$
Initalize the hawk population $x_i\ (i = 1, 2, \dots, n)$
within the boundaries of the decision space D
Initalize a, A_i, C_i
Calulate distances to reference point v for each search agent(hawk) x_i *on all objectives* $f_m(x_i)$
Divide the search agents into groups based on based on number of reference points v
For each refence point group
Calulate distances to reference point v_i *for all* $f_m(x_i)$ *on each search agent* x_i *in group*
x_α *= best non-dominated solution in group (alpha)*
x_β *= second best non-dominated solution in group (beta)*
x_δ *= third best non-dominated solution in group (delta)*
while (t < maximum number of iterations)
for each search agent x_i *in group*
update x_i *position based on equations (6)*
end for
update a, A_i, C_i *based on equations (7, 8)*
calulate objective function fitness for all $f_m(x_i)$ *on each search agent* x_i *in group*
reselect and update $x_\alpha, x_\beta, x_\delta$ *from new values* x_i
t = t+1
end while
return x_α *the best non-dominated solution for point v*
end for

Figure 5. HHMO Multi-Objective Optimization Algorithm

4.3 Elite Preservation

Elite preservation is the idea that in nature predators are likely to remember the locations of potential prey and make decisions whether or not to continue on the current path toward prey or go back towards previous prey. Smarter decisions of this type reduce the amount of time to locate the best prey. Mathematical modeling of this elite

preservation behavior introduces evolutionary strategies into the HHMO algorithm. Evolutionary strategies are different techniques applied during algorithm iteration to create the next generation from the current generation. Several different selection processes were developed by Rechenberg [18] and Schwefel [19]. Two of these evolutionary strategies are direct replacement (μ, λ) and selection of the best (μ+λ). In the (μ, λ) μ parents generate λ offspring [1]. The entire parent population is replaced by the offspring. In (μ+λ)-ES technique, the fitness of parents and offspring are sorted from best to worst [18, 19]. The top population consisting of both parents and first generation offspring is used to create offspring of the second generation

During the development of HHMO algorithm it was established that the (μ+λ)-ES technique returned better solutions in shorter convergence times over the standard (μ, λ)-ES technique. Current predator positions (μ) and recalculated predator positions (λ) are maintained. The top positions for the next iteration are chosen from the combined set (μ+λ) based on the best fitness values.

5 Results

The new HHMO algorithm was tested on a multi-objective optimization problem represented by equation set (5) and graphed in Fig. 6. This problem was chosen for the purpose of comparison as the previously tested reference point based multi-objective predator prey algorithms developed by Deb et al utilizing the same equation set [12].

$$\begin{aligned} &\textit{Minimize: } f_1(x_1, x_2) = x_1^2 \\ &\qquad f_2(x_1, x_2) = \frac{(1+x_2^2)}{x_1^2} \\ &\textit{Subject to: } \sqrt{0.1} \le x_1 \le 1 \\ &\qquad 0 \le x_2 \le \sqrt{5} \\ &\textit{Reference Points:} \\ &\qquad v_1(0.6, 0.18) \\ &\qquad v_2(0.15, 0.46) \end{aligned} \tag{5}$$

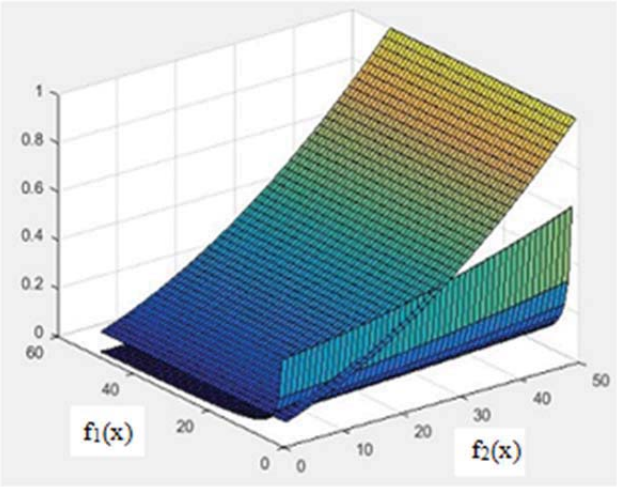


Figure 6. Decision Space *D* for sample problem (5)

HHMO enhanced by the elite preservation (μ+λ)-ES technique was next tested as shown in Fig. 7. The results were achieved after 18 iterations with 50 initial values. The results are shown to repeatedly cluster on the Pareto optimal front near the two specified reference points marked by '*'. Deb's predator prey algorithm with elitism ran for 200 iterations with 50 initial values to achieve similar results [2]. In addition Deb's algorithm required crossover and mutation techniques to assist the elite preservation methods and did not successfully converge on the Pareto optimal front [12]. HHMO did not require these additions to achieve the same results. Table 1 summarizes the average measurement of the solutions in the resulting solution set to the desired reference point along with the number of iterations to achieve this spacing and convergence times.

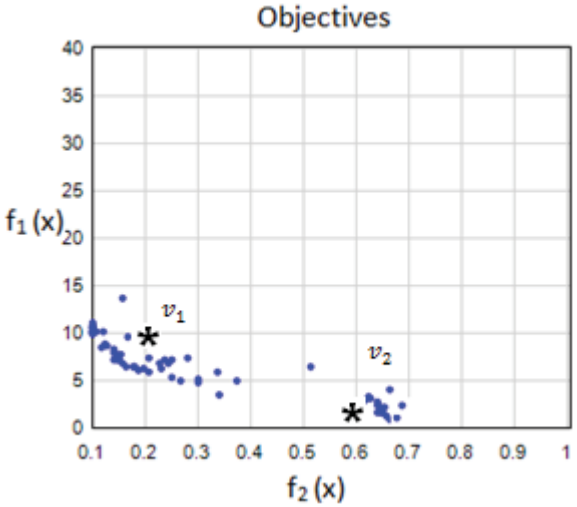


Figure 7. Results for HHMO Algorithm with Elite Preservation

TABLE I. HHMO TEST RESULTS

HHMO	*Iterations*	*Time (ms)*	*Average Distance v_1*	*Average Distance v_2*
Test 1	11	51802.73	2.5162	1.004
Test 2	7	9196.78	1.8716	1.978
Test 3	16	66794.34	2.0014	0.9906

6 Conclusions

The development of the HHMO algorithm was motivated by the unique cooperative hunting behaviors of the Harris's Hawk and the search for multi-objective optimization algorithms that solve the unique reference point based problems. It was shown that by combining our previously developed GWMO algorithm with the mathematical model of the Harris's Hawk hunting

strategies, HHMO was successful in returning solution sets clustered about the desired reference points. It was shown that by having an overhead view of the entire environment or objective decision space of the entire problem, the HHMO algorithm is better able to monitor and direct itself towards potential prey or reference points faster than other algorithms. Future proposed research would be to apply these finding to more robust engineering problems of four or more objectives.

7 References

[1] Deb, Kalyanmoy, and J. Sundar. "Reference point based multi-objective optimization using evolutionary algorithms." *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006.

[2] Yang, Xin-She. "Bat algorithm for multi-objective optimisation." *International Journal of Bio-Inspired Computation* 3.5 (2011): 267-274.

[3] Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*. IEEE, 2009.

[4] Mirjalili, Seyedali, et al. "Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization." *Expert Systems with Applications*47 (2016): 106-119.

[5] Mirjalili, Seyedali. "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm." *Knowledge-Based Systems* 89 (2015): 228-249.

[6] *"Hawks and Cacti",* Google Images, 2015. <https://www.pinterest.com/explore/animal-coloring-pages/>

[7] Kaufman, Kenn. "Harris's Hawk." *Audubon.* National Audubon Society, 13 Nov. 2014. Web. 18 Mar. 2016. <https://www.audubon.org/field-guide/bird/harriss-hawk>

[8] Bednarz, James C. "Cooperative hunting Harris' hawks (Parabuteo unicinctus)." *Science* 239.4847 (1988): 1525-1527.

[9] Dwyer, James F. and James C. Bednarz. 2011. "Harris's Hawk (*Parabuteo unicinctus*), The Birds of North America Online" (A. Poole, Ed.). Ithaca: Cornell Lab of Ornithology; Birds North of America <http://bna.birds.cornell.edu/bna/species/146>

[10] Coulson, Jennifer O., and Thomas D. Coulson. "Reexamining Cooperative Hunting in Harris's Hawk (Parabuteo unicinctus): Large Prey or Challenging Habitats?" *The Auk* 130.3 (2013): 548-552.

[11] "Harris's Hawk", *National Geographic.* National Geographic Society, 2015. <http://video.nationalgeographic.com/video/hawk_harris>.

[12] Deb, Kalyanmoy, and J. Sundar. "Reference point based multi-objective optimization using evolutionary algorithms." *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006.

[13] Deb, Kalyanmoy. *Multi-objective optimization using evolutionary algorithms*. Vol. 16. John Wiley & Sons, 2001.

[14] Deb, Kalyanmoy. "Investigating predator-prey algorithms for multi-objective optimization." KanGAL Report Number 2005010, IIT Kanpur, India. 2005.

[15] Büsing, Christina, et al. "Reference Point Methods and Approximation in Multicriteria Optimization." *arXiv preprint arXiv:1207.3165* (2012).

[16] DeBruyne, S. and Kaur, D. "Grey Wolf Multi-Objective Optimizer (GWMO) based on A-Priori and Evolutionary Strategies." (2016)

[17] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." *Advances in Engineering Software* 69 (2014): 46-61.

[18] Rechenberg, I. "Cybernetic Solution Path of an Experimental Problem" *Roy.Aircr. Establ. Libr. Transl.,* 1122, Farnborough, Hants, UK.1995

[19] Schwefel, Hans-Paul Paul. *Evolution and optimum seeking: the sixth generation*. John Wiley & Sons, Inc., 1993.

Solving the Subgraph Isomorphism Problem Using Simulated Annealing and Evolutionary Algorithms

Zuqing Li, Bernard Chen, and Dongsheng Che*

Abstract— **The tremendous explosion of Big Data such as social network data has made it urgent to mine important information embedded in the huge graphs. One of the mining tasks is to discover the most frequent graph in a large graph. This task is equivalently to the subgraph isomorphism problem, which is to discover the subgraph of a large graph that is isomporphic to a small input graph. The subgraph isomorphism problem is inherently a NP-hard problem, thus an exact algorithm is impossible when graphs are huge, which are the cases these days. In this study, we reported the implementation of three heuristic optimization algorithms: simulated annealing (SA), (1+1) evolutionary algorithm, and genetic algorithm (GA). We applied these algorithms on randomly generated graphs for identifying the isomorphic subgraphs, and compared the performance of these three algorithms. Our experimental results have shown that GA performed better than the other two algorithms overall.**

I. Introduction

Graph models have been applied in many fields such as Bioinformatics, Chemoinformatics, Computer Science, and Socialology. Graphs can be represented to study protein-protein interaction networks [1], chemical components [2], computer networks [3], social networks [4], semantic web [5] and XML data [6]. For instance, the availability of huge amounts of social media data has made it possible for researchers to analyze social networks using graph models, and lead new discoveries and insights through studying graph structures.

While there are many problems have been solved by using graph models in the past decades, this is not the end. Instead, the explosive growth of data will put more challenges to researchers, especially in the era of Big Data. For instance, finding most frequent subgraphs in a large graph will pose a great challenge as the graphs get larger and larger. In the *frequent subgraph* problem, frequent subgraph candidates are first generated, and the frequency of each candidate will be checked [7]. The checking of the candidates within a large graph is is known as the *subgraph isomorphism* problem, which is a NP-complete problem [8].

The algorithms of finding most frequent subgraphs in graph datasets can be roughly classifed into three categories: (1) Mathematical graph-theory based approaches, which include FSG [9], MoFa/Moss [10], Gaston [11], gSpan [12], CloseGraph [13], and gPrune [14]; (2) Greedy search based approaches, including SUBDUE [15] and GBI [16]; and (3) Evolutionary algorithms based approaches such as EP-Subdue [17] and MOSubdue [18].

The approaches for solving the related subgraph isomorphism problem have also been proposed and developed. Some earlier work of using recursive backtracking approach was proposed to solve the graphs with small graph size [19]. Recently, several efficient approaches have been used, including massive parallel computing [20], optimized backtracking search [21], graph neural network [22], and well-founded total order indexing [23]. However, the nature of exponential time complexity of this algorithm makes it impossible to solve this NP-hard problem when the graphs become large. Simulated annealing or evolutionary algorithms, such as genetic algorithm (GA), have been widely used in various NP-hard problems, such as travelling-salesman problem [24], motif-finding problems [25], haplotype problem [26]. GA can guarantee to find sub-optimal solution in a polynomial time, thus keeping the balance between execution time and the quality of solutions.

GA has been applied in the subgraph isomorphism problem in recent years. For instance, Brown et al. [27] used GAs to match two-dimensional chemical graphs. Kim and Moon [28] proposed a two-vertex exchange heuristic GA for the subgraph isomorphism problem, which was used for the malware detection process. The number of different edges between two graphs was used to measure the fitness of individauls. Later, Choi et al. [29] made an improvement by developing a multi-objective GA, which incorporated both the edge information, as well as vertex degree information when comparing the small input graph with the subgraph of the large input graph. The results showed that the multi-objective GA improved the performance.

In this study, we propose to implement various heuritic optimization algorithm for the subgraph isomorphism problem, and compare their performance of the algorithms. Specifically, we will apply the simulated annealing algorithm (SA) [31], the (1+1) evolutionary algorithm (EA) [32], and the genetic algorithm (GA) [33]. All three algorithms measure the quality of the suboptimal solution based on two fitness functions introduced in Choi et al. [29]. The first fitness function measures the number of different eges between the small input graph and the candidate subgraph of the large input graph. The second fitness function is based on the numbers of in-degree and out-degree of the small graph and the candidate subgraph. We will use standard random graph models [30] to generate graph datasets, and

Zuqing Li is with the Department of Computer Science, East Stroudsburg University, East Stroudsburg, PA 18301, USA (email: lzuqing@live.esu.edu).

Bernard Chen is with the Computer Science Department, University of Central Arkansas, Conway, AR 72034, USA (email:bchen@uca.edu)

Dongsheng Che* (Corresponding author) is with the Department of Computer Science, East Stroudsburg University, East Stroudsburg, PA 18301, USA (phone: 570-422-2731; fax: 570-422-3490; email: dche@esu.edu).

apply these algorithms on the generated datasets. Finally, the performance, in terms of the quality of solutions, of these three algorithms will be compared.

The remainder of this paper is organized as follows. Section 2 describes the subgraph isomorphism problem and the fitness function for the subgraph isomorphism problem. Section 3 outlines the simulated annealing algorithm, (1+1) evolutionary algorithm, and genetic algorithm. Section 4 describes the dataset generation, and the experimental results of these three optimization algorithms on generated graph datasets. In Section 5, we will conclude the paper, with discussion and future work.

II. Subgraph Isomorphism Problem

A. Definition

Below we introduce several formal definitions related to the subgraph isomorphism problem.

Definition 1: Given two sets X and Y, a function $f : X \to Y$ is ***bijective*** *if only if every $x \in X$ is paired with exactly one $y \in Y$, and every $y \in Y$ is paired with exactly one $x \in X$.*

Definition 2: A graph $G_1 = (V_1, E_1)$ is ***isomorphic*** *to a graph $G_2 = (V_2, E_2)$, if there is a bijection between the vertex sets V_1 and V_2 $f : V_1 \to V_2$ such that, any pair $(v_1, v_2) \in E_1$ if only if $(f(v_1), f(v_2)) \in E_2$.*

Definition 3: Given a graph $G = (V, E)$, a graph $G_s = (V_s, E_s)$ is a ***subgraph*** *of G if $V_s \subseteq V$ and $E_s \subseteq E$.*

There are two versions of the subgraph isomorphism problem: the *decision* version and the *optimization* one. For two given graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ such that $|V_1| < |V_2|$, the task for the decision version is to determine whether G_2 contains a subgraph that is isomorphic to G_1. For the optimization version, we assume that there exists a subgraph in G_2 that is isomorphic to G_1, so that the task of the optimization version is to discover a subgraph of G_2 that is isomorphic to G_1. In this paper, we only consider the optimization version of subgraph isomorphism problem, and compare the performance of three heuristic optimization algorithms.

B. Objective Functions

As we mentioned before, the subgraph isomorphism problem is NP-hard problem, thus the exact algorithm for finding the optimal solutions will be exponential time. This makes it impossible to find the solutions for large graph datasets. The approximation algorithms cannot guarantee to find the global optimal solutions, thus the design the objective functions for evaluating the quality of the candidate solutions become crucial.

In this study, we use two objective functions for evaluating the candidate solutions: The first function is edge based, which is strictly based on the definition the subgraph isomorphism problem. The second function is vertex degree based, which is reported to make improvements of performance when the first fitness function cannot differentiate the candidate subgraphs, *i.e.*, with the same values. Below, we describe the two functions in detail.

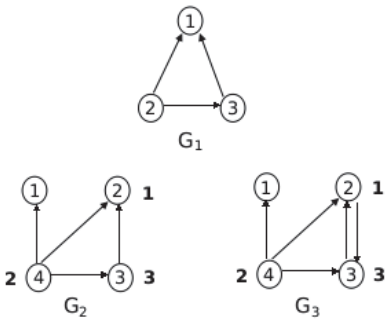


Fig. 1. Example of subgraph isomorphism and nonisomorphism

1) Edge-based Fitness Function: Let $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, where $|V_1| < |V_2|$. Let $G_s = (V_s, E_s)$ is a subgraph of G_2 such that for any $v \in V_1$, there is $f(v) \in V_s \subseteq V_2$. The mapping function f is as follows:

$$f : V_1 \to V_2$$

According to the definition of subgraph isomorphism, for any pair $(v_1, v_2) \in E_1$ if only if there is a $(f(v_1), f(v_2)) \in E_2$. Unfortunately, when we start to search our solutions, we can easily find a mapped vertex set in G_2, but not necessary meeting the edge constraints. To that end, an edge-based fitness function to measure the closeness of a candidate subgraph for the subgraph isomorphism problem is defined as follows:

$$Fit_{edge} = \sum_{(v_1,v_2)\in E_1} |(f(v_1), f(v_2)) \notin E_2| + \sum_{(f(v_1),f(v_2))\in E_2} |(v_1, v_2) \notin E_1| \quad (1)$$

The above fitness function consists of two parts: the first one counts the number of edges in G_1 but not in G_s, while second one counts the number of edges in G_s but not in G_1. Thus, if G_1 is isomorphic to G_s, then the fitness value should be 0. On the other hand, the fitness value will be greater than 0 if G_1 is not isomorphic to G_s. Figure 1 shows an example of isomorphism between G_1 and subgraph $G_2(s)$, with the mapping between $\{v_1, v_2, v_3\} \subseteq V_1$ and $\{v_2, v_4, v_3\} \subseteq V_2$, and an example of non-isomorphism between G_1 and subgraph $G_3(s)$, with the mapping between $\{v_1, v_2, v_3\} \subseteq V_1$ and $\{v_2, v_4, v_3\} \subseteq V_3$. Table 1 shows the dissection of the edge mapping information corresponding to Figure 1.

2) Vertex degree-based Fitness Function: One of the properties for the subgraph isomorphism problem is that: For any two graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, where $|V_1| < |V_2|$, vertex $v_i \in V_1$ cannot be mapped to vertex $v_j \in V_2$ if $deg(v_i) > deg(v_j)$. Thus, the degree constraint information can also be used to measure the quality of the candidate

TABLE II
AN EXAMPLE OF VERTEX DEGREE-BASED FITNESS FUNCTION

G_1	d_{in}	d_{out}	$G_2(s)$	d_{in}	d_{out}	unq	$G_3(s)$	d_{in}	d_{out}	unq
v_1	2	0	v_1	1	0	$\times$	v_1	1	0	$\times$
v_2	0	2	v_4	0	3		v_4	0	3	
v_3	1	1	v_2	2	0	$\times$	v_2	2	1	
Fitness						2				1

TABLE I
THE CALCULATION OF EDGE-BASED FITNESS FUNCTION

Parts	(G_1, G_2)	(G_1, G_3)
$(v_1, v_2) \in E_1 \setminus (f(v_1), f(v_2)) \in E_2$	$\emptyset$	$\emptyset$
$(f(v_1), f(v_2)) \in E_2 \setminus (v_1, v_2) \in E_1$	$\emptyset$	$\{(v_3, v_2)\}$
Total fitness value	0	1

solutions. The incorporation of the degree constraint has been shown to improve the optimization process [29], thus we also include it in our optimization algorithms.

$$Fit_{degree} \quad = \quad |\hat{V_s}| \tag{2}$$

where $|\hat{V_s}|$ is the number of unqualified vertices in G_s based on the degree constraints. A vertex $f(v) \in E_s$ is unqualified if either of the following conditions holds:

$$d_{in}(v \in E_1) \quad > \quad d_{in}(f(v) \in E_s) \tag{3}$$

$$d_{out}(v \in E_1) \quad > \quad d_{out}(f(v) \in E_s) \tag{4}$$

where $d_{in}(v)$ is the degree of incoming edges to vertex v, while $d_{out}(v)$ is the degree of outcoming edges from vertex v. Figure 1 shows two examples of non isomorphic graphs, between G_1 and subgraph $G_2(s)$, with the mapping between $\{v_1, v_2, v_3\} \subseteq V_1$ and $\{v_1, v_4, v_2\} \subseteq V_2$, and between G_1 and subgraph $G_3(s)$, with the mapping between $\{v_1, v_2, v_3\} \subseteq V_1$ and $\{v_1, v_4, v_2\} \subseteq V_3$. Both of the non-isomorphic graphs have the same number of mis-mapping edges (*i.e.*, 1), but they have the different vertex degree values. As shown in Table II, $G_2(s)$ has two unqualified vertices, v_1 and v_2 while $G_3(s)$ has one unqualified vertex, v_1.

3) Weighted Fitness Function: In our optimization algorithms, we evaluate the quality of candidate subgraph solutions using both edge-based and vertex degree-based functions. Thus, a weighed function is employed, and given as follows:

$$Fit(w_e, w_d) \quad = \quad w_e Fit_{edge} + w_d Fit_{degree} \tag{5}$$

where w_e is the weight for the edge-based fitness function, while w_d is the weight for the vertex degree-based fitness function. Since the edge-based fitness function is be main criteria to estimate the quality of candidate solutions, we set the weight range for $w_e = 0.9$.

Algorithm 1 : Simulated Annealing

```
T ← T_0
generate an initial State S_0
compute C(S_0) using its cost function
C(S_cur) ← C(S_0)
repeat
    generate a neighboring state S_new
    ΔC ← C(S_new) − C(S_cur)
    if ΔC < 0 then
        S_cur ← S_new
    else
        p ← Random(0, 1)
        if p < exp^(−ΔC/T) then
            S_cur ← S_new
        end if
    end if
    T ← β × T // β is a constant
until T reaches to a frozen temperature or C(S_cur) = 0
return the best state S_cur
```

III. HEURISTIC OPTIMIZATION ALGORITHMS

In this section, we outline three searching optimization algorithms: (1) the simulated annealing algorithm (SA), (2) the (1+1) evolutionary algorithm (EA), and (3) the genetic algorithm (GA). We use the same weighted fitness function described in Section 2. Different optimization algorithms use different terms when evaluating solutions, we use objective function, fitness function, or cost function interchangeably throughout the text.

A. Simulated Annealing

Simulated annealing (SA) is a stochastic optimization algorithm based on neighborhood search. A greedy neighborhood search always moves the current solution to the best neighboring solution, thus it can be easily trapped into a local optimum, and consequently the global optimal solution will not found. SA is a strategy that occasionally takes a neighboring solution worse than the current solution, aiming to jump from the local optimum.

SA starts with an initial temperature T_0 and an arbitrary solution. In the case of subgraph isomorphism problem, it is the permutation of the vertices of the larger graph. SA repeatedly generates neighboring solutions when the temperature cools down. The acceptance of a new neighboring

solution is dependent on the quality of the new solution. The neighboring solution is always taken if the new neighboring solution is better than the current solution. If the new solution is worse than the current solution, then the probability of taking this new solution depends on the current temperature, and the quality difference between the new solution and the current solution, measured by the cost function. The SA terminates when the temperature reaches to a frozen temperature, or the solution has been found. The pseudo-code of the simulated annealing algorithm is summarized in Algorithm 1.

B. (1+1) Evolutionary Algorithm

An (1+1) evolutionary algorithm (EA) is a simple genetic algorithm, where there is only one member in the population, and with no crossover. (1+1) EA is also similar to simulated annealing, except that it only takes the better individual in terms of fitness value in each iteration.

(1+1) EA starts with the generation of one individual. The individual is represented as a permutation of the vertices of the larger graph, and the first n vertices of the individual is used to map to the smaller graph, so that the fitness function can be computed. The (1+1) EA repeats a predetermined number of iterations, where each iteration the individual is mutated, and a fitness value of a mutated child is computed and compared to the parent one. The algorithm takes the mutated individual if the mutated one has better fitness value. The algorithm stops if the number of iterations has reached or the solution is found. The pseud-code of the (1+1) EA is summarized in Algorithm 2.

Algorithm 2 (1+1) EA

1: generate an initial individual P_0
2: compute $C(P_0)$ using its cost function
3: $C(P_{cur}) \leftarrow C(P_0)$
4: **repeat**
5: $child \leftarrow$mutate(P_{cur})
6: **if** $C(child) < C(P_{cur})$ **then**
7: $P_{cur} \leftarrow child$
8: **end if**
9: **until** a specified number of iterations have elapsed, or $C(P_{cur}) = 0$
10: **return** the individual P_{cur}

C. Genetic Algorithm

In the genetic algorithm (GA), the initial population consists of randomly generated individuals, where each individual is a permutation of the vertices of the larger graph. During the evolution procedure, individuals compete for the opportunity to reproduce based on fitness values, evolves over generations, and terminates when the stopping condition is met. The best individual is returned as the output of the GA. The outline of the GA for subgraph isomorphism problem is summarized in Algorithm 3, and the GA parameter setup is listed in Table III.

Algorithm 3 : Genetic Algorithm

1: **create** an initial population P
2: **evaluate** each candidate according to fitness function
3: **repeat**
4: **select** parents from P using tournament selection
5: **crossover** parents using cut-and-crossfill crossover
6: **mutate** the resulting offspring
7: **evaluate** new candidates
8: **replace** all parent individuals (except the best one) with all offspring
9: **until** a specified number of iterations have elapsed, or the best individual's $fitness = 0$
10: **return** the best individual

TABLE III
GA PARAMETER SETUP

Representation	Permutation
Population size	100
Parent selection	Tournament selection with size of 3
Crossover	Cut-and-crossfill crossover
Crossover rate	0.7
Mutation	Swap
Mutation rate	0.5
Survivor selection	Elitism
Generation	300
Stop criteria	Solution or the number of iterations have elapsed

IV. EXPERIMENTS

In this section, we present the experimental results of three optimization algorithms on various graph datasets. We start with the introduction of graph generation, and then discuss the results on the graph datasets.

A. Graph Dataset and Experiment Setup

We used the $G(n, p)$ model [30] to generate the larger graphs G_2. We chose $n = 100$, and edge probabilities p = (0.01, 0.05, 0.1, 0.2). For each combination of G_2 graphs, we generated ten independent graphs. For each random copy of G_2, we further randomly picked a number of vertices m = (10, 30, 50, 70, 90) to form $G_1(m)$ graphs. The edges for $G_1(m)$ were determined from the edges of G_2. The procedure of choosing graphs G_1 can guarantee that there exists a subgraph G_s in G_2 that is isomorphic to G_1.

The graph generation setup has 20 combinations of ($G_1(m)$, $G_2(100, p)$), with each combination of 10 independent copies. We ran each of optimization algorithms 10 times on each of graph pair combinations. Since each combination has 10 independent graphs, our final evaluations were averaged on 100 trials for each of 20 combinations. The algorithms were implemented in Java language with JDK1.7, and executed on a 64-bit Windows 7 (SP1) ultimate, and AMD Fx(tm)-8350 Eight-Core Processor 4.00 GHz computer.

TABLE IV
PERFORMANCE COMPARISON OF THREE ALGORITHMS

	$G_2 = (100, 0.01)$			$G_2 = (100, 0.05)$			$G_2 = (100, 0.1)$			$G_2 = (100, 0.2)$		
G_1	**GA**	**SA**	**EA**	**GA**	**SA**	**EA**	**GA**	**SA**	**EA**	**GA**	**SA**	**EA**
10	0.34	0.89	0.87	2.18	3.06	2.94	6.10	7.35	6.71	14.13	14.67	13.62
30	8.11	11.16	10.88	53.90	61.06	61.46	112.34	119.58	119.02	213.35	216.13	215.07
50	30.51	38.17	37.61	172.60	186.00	187.22	353.07	367.87	364.01	641.94	651.73	647.01
70	68.18	77.39	77.29	365.09	380.56	381.32	714.36	730.51	728.68	1299.80	1309.62	1305.29
90	123.96	129.30	129.28	637.66	647.50	646.51	1232.57	1240.60	1239.43	2207.44	2212.24	2210.42

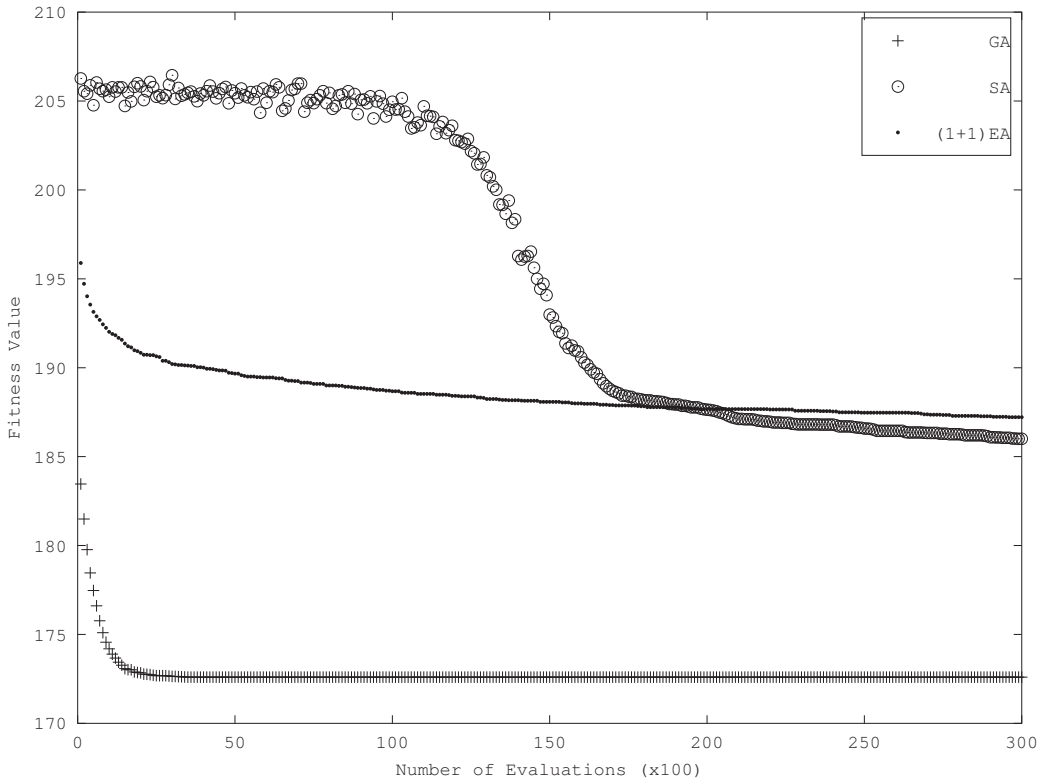


Fig. 2. The fitness values of each step on graphs $(100, 0.05, 50)$

B. Experimental Results

We ran SA, (1+1) EA, and GA on generated datasets as described above. In order to make it fair to compare the performance of these three algorithms, we compared the fitness values after each algorithm reached to the same number of fitness evaluations. For GA, we set up the population size of 100 and the number of generation of 300, so the total number of evaluations was 30,000 after 300 generations. We used the best solution after 300 generations (equivalently 30,000 evaluations). Using this as the baseline, we set 30,000 steps (iterations) in both SA and (1+1) EA, and used the solutions after 30,000 iterations.

Table IV shows the average fitness values (on 100 trials) of the solutions found by each of three algorithms. As we can see, for the same setting of graph G_2, the fitness values increase when G_1 becomes larger. This can be explained by the increase of edge-based fitness values and vertex edge-based fitness values when G_1 becomes larger. On the other hand, for the same setting of graph G_1, the fitness values increase when the edge density p becomes larger (from 0.01 to 0.5). For instance, for the same $G_1 = 30$, the fitness values of the best solutions found by GA were 8.11, 53.90, 112.34 and 213.35 for edge densities of 0.01, 0.05, 0.1 and 0.5. This can be explained by the increase of edge-based fitness values.

Out of 20 combinations of datasets we tested, we can see the GA consistently performs better than SA and (1+1) EA, with the exception of one combination, (100, 0.2, 10), where GA performs slightly worse than the (1+1) EA algorithm (Table IV). When comparing the performance of SA and EA, we found that there is no significant superiority of one over another (Table IV).

Figure 2 and 3 show the fitness values of each step on graphs (100, 0.05, 50), and (100, 0.2, 50). As we can

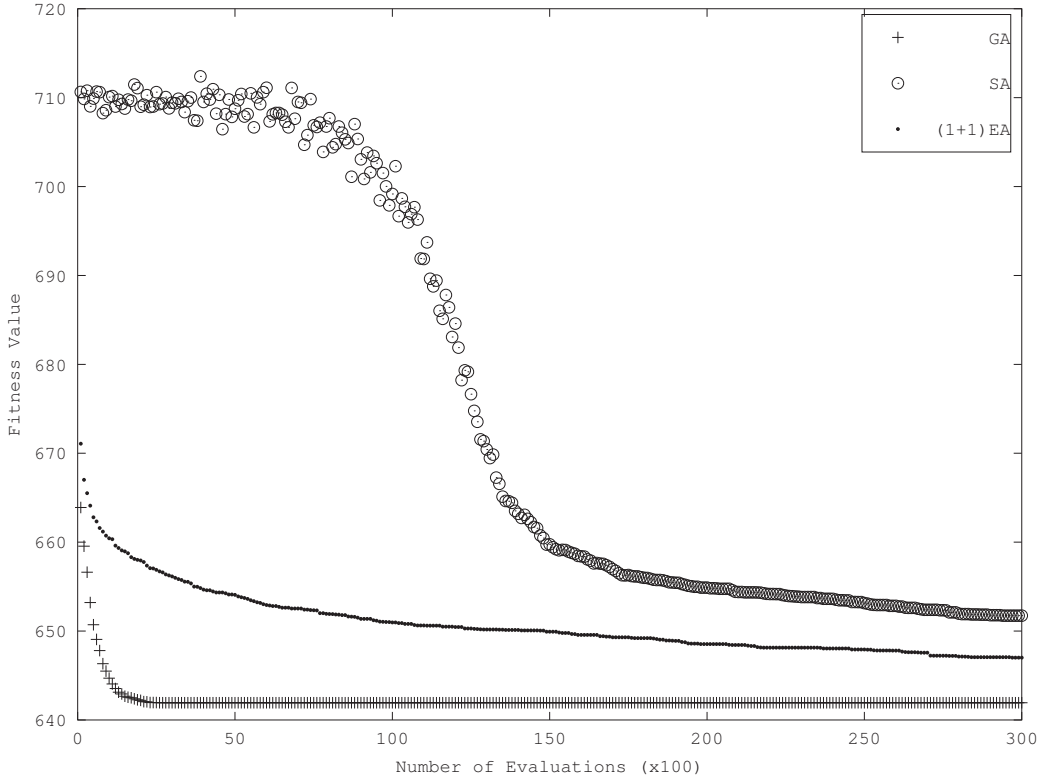


Fig. 3. The fitness values of each step on graphs $(100, 0.2, 50)$

see, the fitness values in SA stay high in the beginning of the searching process, but decrease and stabilize after the temperature cools down. Unlike SA, the fitness values of (1+1) EA monotonously decrease. From our experimental results, it is hard to judge which algorithm is better than the other.

Compared to SA and (1+1) EA, GA find the suboptimal solutions much faster and better than the other two. This can be explained by the diversity of the population, which contributes the generation of offspring with high population.

V. Conclusions

In this paper, we implemented the algorithms: SA, (1+1) EA and GA for solving the subgraph isomorphism problem. We used both edge-based and vertex-degree based objective functions to evaluate the solutions for these three algorithms. We generated 20 combinations of graph datasets for the evaluating the algorithms. Our results showed that GA performs better than SA and (1+1) EA in most of the cases. On the other hand, there is no evidence showing SA better than (1+1) EA, or (1+1) EA better than SA. The experiments indicate the the diversity of the population, and the crossover and mutation operations, together may contribute the generation of better individuals.

While GA performs better than the other two algorithms, GA itself converges very quickly, cannot generate good individuals in the later generations. The design of new fitness functions, and the use of new crossover and mutation operators are the possible directions for the future improvement of the subgraph isomorphism problem.

References

[1] Thomas, A. et al., "On the structure of proteinprotein interaction networks," *Biochem. Soc. Trans*. vol. 31, pp. 1491–1496, 2003.

[2] A. Kerber, R. Laue, M. Meringer, and C. Rcker, "Molecules in Silico: A Graph Description of Chemical Reactions," *Journal of Chemical Information and Modeling*, vol. 47, no. 3, pp. 805–817, 2007, pMID: 17532665. [Online]. Avaliable: http://dx.doi.org/10.1021/ci600470q

[3] J. M. McQuillan, "Graph theory applied to optimal connectivity in computer networks," *ACM SIGCOMM Comput. Commun. Rev.*. vol. 7, pp. 13-41, Apr.1977.

[4] M.P. Cullar, M. Delgado, M.C. Pegalajar, "Improving learning management through semantic web and social networks in e-learning environments," *Expert Systems with Applications*. vol. 38, no. 4, pp. 4181–4189, 2011.

[5] B. Berendt, A. Hotho, and G. Stumme, "Towards semantic web mining," *IN INTERNATIONAL SEMANTIC WEB CONFERENCE (ISWC)*. Springer, pp. 264–278, 2002.

[6] E. Damiani, B. Oliboni, E. Quintarelli, and L. Tanca, "Modeling semistructured data by using graph-based constraints," *On The Move to Meaningful Internet Systems 2003: OTM 2003 Workshops*. Springer, pp. 20–21, 2003.

[7] J. Han and M. Kamber, "Data Mining, Southeast Asia Edition: Concepts and Techniques," Morgan kaufmann, 2006.

[8] D. S. Johnson and M. R. Garey, "Computers and intractability: A guide to the theory of NP-completeness," *Freeman&Co, San Francisco*, vol. 31, 1979.

[9] M. Kuramochi and G. Karypis, "An efficient algorithm for discovering frequent subgraphs," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 16, no. 9, pp. 1038–1051, 2004.

[10] C. Borgelt and M. R. Berthold, "Mining molecular fragments: finding relevant substructures of molecules," *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pp. 51–58, 2002.

[11] S. Nijssen and J. N. Kok, "A quickstart in frequent structure mining can make a difference," *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 647–652, 2004.
[12] X. Yan and J. Han, "gspan: Graph-based substructure pattern mining," *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*. IEEE, pp. 721–724, 2002.
[13] X. Yan and J. Han, "CloseGraph: mining closed frequent graph patterns," *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 286–295, 2003.
[14] F. Zhu, X. Yan, J. Han, and S. Y. Philip, "gPrune: a constraint pushing framework for graph pattern mining," *Advances in Knowledge Discovery and Data Mining*. Springer, pp. 388–400, 2007.
[15] D. J. Cook and L. B. Holder, "Substructure discovery using minimum description length and background knowledge," *arXiv preprint cs/9402102*, 1994.
[16] T. Matsuda, T. Horiuchi, H. Motoda, and T. Washio, "Extension of graph-based induction for general graph structured data," *Knowledge Discovery and Data Mining. Current Issues and New Applications*. Springer, pp. 420–431, 2000.
[17] S. Bandyopadhyay, U. Maulik, D. J. Cook, L. B. Holder, and Y. Ajmerwala, "Enhancing Structure Discovery for Data Mining in Graphical Databases Using Evolutionary Programming," *FLAIRS Conference*, pp. 232–236, 2002.
[18] P. Shelokar, A. Quirin, and Ó. Cordón, "MOSubdue: a Pareto dominance-based multiobjective Subdue algorithm for frequent subgraph mining," *Knowledge and information systems*, vol. 34, no. 1, pp. 75–108, 2013.
[19] Bollob as, B, *Random graphs (2nd ed.)*, Cambridge, UK: Cambridge University Press, 2001.
[20] Z. Sun, H. Wang, H. Wang, B. Shao, and J. Li, "Efficient subgraph matching on billion node graphs." *Proceedings of the VLDB Endowment,* 5(9):788–799, 2012.
[21] L. Fu, S. Chandra, "Optimized backtracking for subgraph isomorphism," *International Journal of Database Management Systems* vol. 4. no. 6, pp: 1-10, Dec 2012.
[22] G. R. Baskararaja, M. S. Manickavasagam, "Subgraph matching using graph neural network," *Journal of Intelligent Learning Systems and Applications,* pp: 274-278, April 2012.
[23] M. Weber, M. Liwicki, A. Dengel, "Faster subgraph isomorphism detection by well-founded total order indexing," *PR Letters 33(15)*, pp: 2011-2019, 2012.
[24] K. Bryant, "Genetic algorithms and traveling salesman problem," Master's thesis, Harvey Mudd College, Claremont, United States, 2000.
[25] D Che, Y Song, and K Rasheed, "MDGA: motif discovery using a genetic algorithm," *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. ACM, pp. 447–452, 2005.
[26] D Che, H Tang, and Y Song, "Haplotype inference using a genetic algorithm," *Computational Intelligence in Bioinformatics and Computational Biology, 2009. CIBCB'09. IEEE Symposium on*. IEEE, pp. 31–37, 2009.
[27] R. Brown, G. Jones, P. Willett, and R. Glen, "Matching two-dimensional chemical graphs using genetic algorithms," *Journal of Chemical Information and Computer Sciences*, vol. 34, no.1 , pp. 63-70, 1994.
[28] K. Kim and B. Moon, "Malware detection based on dependency graph using hybrid genetic algorithm," *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*. ACM, pp. 1211-1218, 2010.
[29] J. Choi, Y. Yoon, and B.-R. Moon, "An efficient genetic algorithm for subgraph isomorphism," *Proceedings of the Fourteenth International Conference on Genetic and Evolutionary Computation Conference, GECCO12*. ACM, pp. 361-368, 2012.
[30] J. Ullmann, "An algorithm for subgraph isomorphism," *Journal of the ACM (JACM)*. vol. 23, no. 1, pp. 31-42, 1976.
[31] Bertsimas, D. and Tsitsiklis, J. , "Simulated Annealing," *Probability and Algorithms*. Washington, D.C.: National Academy Press, pp. 17–29, 1992.
[32] Droste, S., T. Jansen, and I. Wegener, "On the analysis of the (1+1) evolutionary algorithm," *Theoretical Computer Science 276*. pp. 51–81, 2002.
[33] Mitchell Melanie, *Introduction to Genetic Algorithms*, MIT Press, pp. 158, 1999.

Multi-Source Power System LFC Using the Fractional Order PID Controller Based on SSO Algorithm Including Redox Flow Batteries and SMES

H.A. Shayanfar *
Department of Elec. Engineering
College of Technical & Engineering,
South Tehran Branch,
Islamic Azad University

H. Shayeghi
Department of Electrical Engineering,
University of Mohaghegh Ardabili,
Ardabi, Iran

A. Molaee
Department of Electrical Engineering,
University of Mohaghegh Ardabili,
Ardabi, Iran

hashayanfar@gmail.com, hshayeghi@gmail.com, amolaee.a@gmail.com

Abstract- Load Frequency Control (LFC) of two-area interconnected power system using Fractional Order Proportional-Integral-Derivative (FOPID) controller based on Social Spider Optimization (SSO) algorithm is investigated in this paper. The FOPID controller is a non-integer order controller that improves the performance of the integer order PID controller. The difference between the FOPID and PID controller is fractional order of integrator and derivative in FOPID that make two degree of freedom in designing. In other word, the FOPID controller has five parameters to be tuned for exact performance. Hence, SSO algorithm is chosen to optimal tune the FOPID controller parameters in this paper. The test system is the two area multisource interconnected power system including reheat thermal, hydro, gas and diesel power plants. The proposed objective function consists of Integral of Time multiply Absolute Error (ITAE) plus sum of settling times of frequency deviation of each control area and tie-line power flow deviation that are associated with proper weighted coefficients. Also, to show the stability of FOPID controller the generation rate constraint is considered in this study and the Redox Flow Batteries (RFBs) and Super Magnetic Energy Storage (SMES) is used to improve the dynamics of the system response under the load disturbances and system nonlinearity. The simulation results reveal the good performance of the SSO based on FOPID controller in LFC problem.

Keyword: FOPID controller, Three-area power system, SSO algorithm, GRC, SMES, RFB.

1. Introduction

One of the most important issues in power system operation and protection is keeping the frequency and voltage in the nominal value or in predetermined limitation. The reliability and quality of generation power depend on balance between the power generated and power demand plus power losses in all over the power system. When this balance is disrupted causes deviation in frequency and tie-line power. Also, the loads are changed randomly. Hence using a proper control method it is necessary to prevent the power system to go to an unstable state. So far, different control method have been used to control the power system frequency in various type of power system.

Until now, some review articles in the LFC issue have been published [1]. In Ref. [2] the Proportional-Integral (PI) based on fuzzy gain scheduling considering generation Rate Constraint (GRC) has been proposed for a four-area interconnected power system. The fuzzy PID type controller for load frequency control of two-area power system has been investigated in [3]. Two-degree-of-freedom PID load frequency controller in presence of GRC at the power generation unit is performed in [4]. In [5] the robust PID controller is tuning based on maximum peak of response specification for Automatic Generation Control (AGC) in multi-machine power system. An adaptive PID has been tuned for LFC problem using Neuro Fuzzy Inference System (ANFIS) and Artificial Neural Network (ANN) based on Genetic Algorithm (GA) [6]. Recently, the researchers have used an improved form of PID controller called FOPID controller at the various research works. In [7] the FOPID controller parameters has been optimized using CNC-ABC algorithm for Automatic Generation Regulator. The FOPID controller is applied for stability analysis and compensation of reactive power in a micro-grid [8].

The FOPID controller has two extra parameters more than PID controller that makes two degree of freedom in controller design and application. Hence, the FOPID controller has been applied to load frequency control of two area multi-source interconnected power system

* Corresponding Author, H.A. Shayanfar (hashayanfar@gmail.com)

considering GRC and the Flexible Alternating Current Transmission System (FACTS) devices such as SMES, in this paper. Also, optimal tuning of controller parameters to achieve the good performance and dynamics response is necessary. The SSO algorithm is one of the heuristic algorithm that is being used to solve the optimization problem. This algorithm can balance exploration and exploitation of solutions in the search space and prevent the premature convergence. Hence, the SSO algorithm is used for optimizing the FOPID controller to obtain the suitable dynamics response. The FOPID based on SSO algorithm has been applied on the two area multi-source interconnected power system. The simulation results show the good performance of this approach to damp the frequency and tie-line deviation in the presence of load disturbances and nonlinearities. Also, the effect of SMES on improvement of disturbance reduction can be seen in the results.

2. Fractional calculus

There are different definition for Fractional calculus and the Caputo's definition [9] is chosen here and formulation is as follows:

$$ {}_{\alpha}D_t^{\mu} f(t) = \frac{1}{\Gamma(\mu - n)} \int_{\alpha}^{t} \frac{f^{(n)(\tau)}}{(t-\tau)^{\beta - n + 1}} d\tau \quad (1) $$

The above equation is the fractional Differential operator and the μ is the fractional order, m is an integer, $\Gamma(.)$ is the Euler's gamma function, $n-1 \le \mu < n$ and the other details is described in [10]. For usage of fractional operator in application study, it is necessary that the approximation in integer order transfer function form be obtained. In this paper the Oustaloup's approximation [10] is used for approximation and its formulation is as follows:

$$ s^{\mu} = K \prod_{i=-P}^{P} \frac{s + \omega_i'}{s + \omega_i} \quad (2) $$

$$ \omega_i' = \omega_l \left(\frac{\omega_h}{\omega_l}\right)^{\frac{i + P + 0.5(1-\mu)}{2P+1}} \quad (3) $$

$$ \omega_i = \omega_l \left(\frac{\omega_h}{\omega_l}\right)^{\frac{i + P + 0.5(1+\mu)}{2P+1}} \quad (4) $$

$$ K = \left(\frac{\omega_h}{\omega_l}\right)^{-\frac{\mu}{2}} \prod_{i=-P}^{P} \frac{\omega_i}{\omega_i'} \quad (5) $$

Where, μ is the fractional order of derivative, $2P$ is the number of zeros and poles, ω_i' and ω_i is the *i*th zero and pole in range of $[\omega_l, \omega_h]$, respectively.

3. FOPID Controller

Definitely, the *PID* controller is one of the most applicable controller that is used in industrial applications [11]. In the [12] an improved form of *PID* controller is proposed by converting the integer order of derivative and integrator into the fractional order. This controller is known as FOPID ($PI^{\lambda}D^{\mu}$) controller. Hence, the only difference between the PID and FOPID controller is the order of calculus operators that make more degree of freedom and flexibility in design and practice. The mathematical formulation of FOPID controller is as follows:

$$ C(s) = K_P + K_I \frac{1}{S^{\lambda}} + K_D S^{\mu} \quad (6) $$

Where, the K_P, K_I and K_D are proportional, integral and differential gains and the λ and μ are fractional order of integrator and derivative, respectively. The structure of FOPID controller is given in Fig. 1.

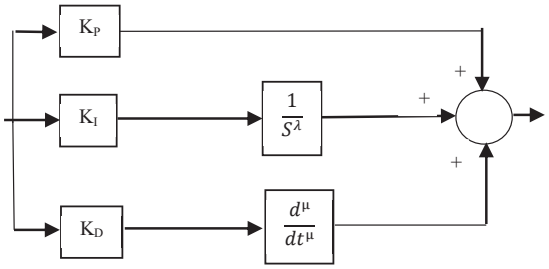


Fig. 1. FOPID controller structure

4. SSO Algorithm

The SSO algorithm is a swarm intelligence based algorithm that is proposed by Erik Cuevas et.al [13] in 2013. This algorithm simulate the cooperative behavior of the social spiders and its mechanism is based on two different search agent known as female and male spider. The SSO has a special structure to find the solutions in the search space, so that it make the balance between exploration and exploitation and also, prevents falling into the local optimum. It's assumed that the all spiders communicate with each other by the communal web that makes the search space in this algorithm. Actually, the communal web is the link that the spiders transform information through it and the information is transferred using the produced vibration by the spiders. The three different types of vibration inspired by biological principles of the social spiders are modeled and described following. Here, a brief of the algorithm is explained and in detail expressed in [13]. The vibration realized by *i*th spider as a result of the information transformed by *j*th spider and is as follows:

$$ Vib_{i,j} = w_j e^{-d_{i,j}^2} \quad (7) $$

Where, the *di,j* is the Euclidian's norm between the spiders i and j. Similarly, $Vibc_i$ is the vibration realized by *i*th spider as a result of the transformed information by the *c*th spider that has two major characteristics as following: it is the nearest spider to *i*th spider and has a higher weight in comparison to *i* ($w_c > w_i$). $Vibf_i$ is the

vibration realized by *i*th spider as a result of the transformed information by the *b*th spider that has highest weight in whole population. $Vibf_i$ is the vibration realized by *i*th spider as a result of the transformed information by the *f*th spider that is the nearest female spider to the *i*. The formulations of above definition about different type of vibration are given following:

$$Vibc_i = w_c e^{-d_{i,c}^2} \quad (8)$$

$$Vibb_i = w_b e^{-d_{i,b}^2} \quad (9)$$

$$Vibf_i = w_f e^{-d_{i,f}^2} \quad (10)$$

The flowchart of the SSO algorithm is shown in Fig. 2.

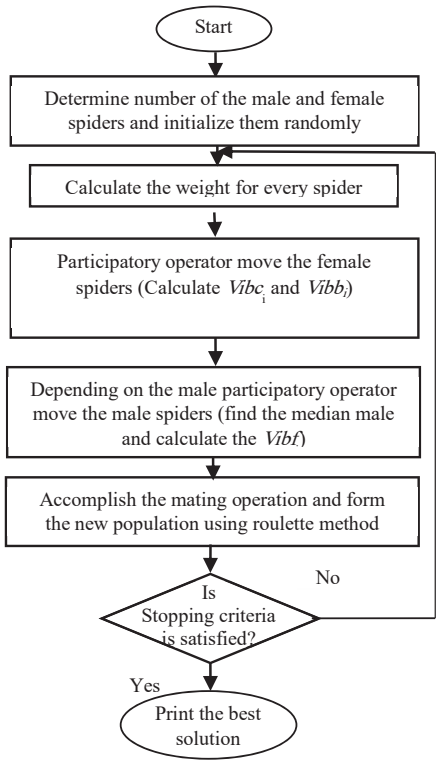


Fig. 2. Flowchart of the SSO algorithm

5. FOPID Design for LFC Problem

In this paper, the FOPID controller based on SSO algorithm has been investigated for LFC in the two area multi-source interconnected power system including three generation power in each control area that can be seen at Fig. 3. The control area-1 is consist of thermal reheat, hydro and gas power plant and control area-2 including thermal reheat, hydro and diesel power plant [14, 15]. Each plant need to be controlled by a controller, as a result, there are six FOPID controller and each controller has five parameters that should be tuned using SSO algorithm to achieve acceptable response. High performance of the proposed controller has been shown by applied on the tested system in the presence of nonlinearity and uncertainties in the power system parameters. Furthermore, in order to improve the damping of the frequency deviation the RFB and SMES are considered in simulation. To evaluate the performance of the SSO based FOPID controller it is need to discuss some error criterion such as ITEA and performance indices such as settling time and peak overshoot of power system response to load disturbances in different operation conditions. The proposed objective function is considered as follows:

$$J = a_1.ITAE + a_2.(OS) + a_3.(\mathrm{ST}) \quad (11)$$

$$ITAE = \int_o^{tsim} (|\Delta f_1| + |\Delta f_2| + |\Delta P_{Tie-line}|).tdt \quad (12)$$

$$OS = os_1 + os_2 + os_3 \quad (13)$$

$$ST = st_1 + st_2 + st_3 \quad (14)$$

Where, Δf_1 and Δf_2 are the frequency deviation in control area-1 and area-2, respectively and $\Delta P_{Tie-line}$ is tie-line power deviation. os_1, os_2 and os_3 are the peak overshoot and st_1, st_2 and st_3 are the settling time of the frequency deviation of area-1, area-2 and tie-line power deviation, respectively. The a_i coefficients are adjusted so that the all terms of Eq. (11) be in the same range.

The GRC and SMES model that have been used in this paper are shown in Fig. 4 [15, 16]. The GRC is considered 3% per min at the thermal unit in each control area and the power system parameters are given in Appendix. The SMES parameters must be optimized to improve the power system performance. The simulation is performed for three case studies as following:

Case 1. The FOPID controller has been applied on the two area multi-source interconnected power system considering GRC, only.

Case 2. The power system frequency has been controlled in the presence of GRC and considering the RFB in area-1 and SMES in area-2.

The FOPID controller parameters are optimized using SSO algorithm. The SSO and optimized controller parameters are given in Tables 1 and 2 and the SMES parameters optimized by SSO are shown in the Table 3. Also, the frequency deviation of area-1 and area-2 and tie-line power deviation are seen in Figs. 5 to 7. The ITAE and performance indices such as settling time and peak overshoot of frequency and tie-line power deviation related to both of cases are given in the Table 4.

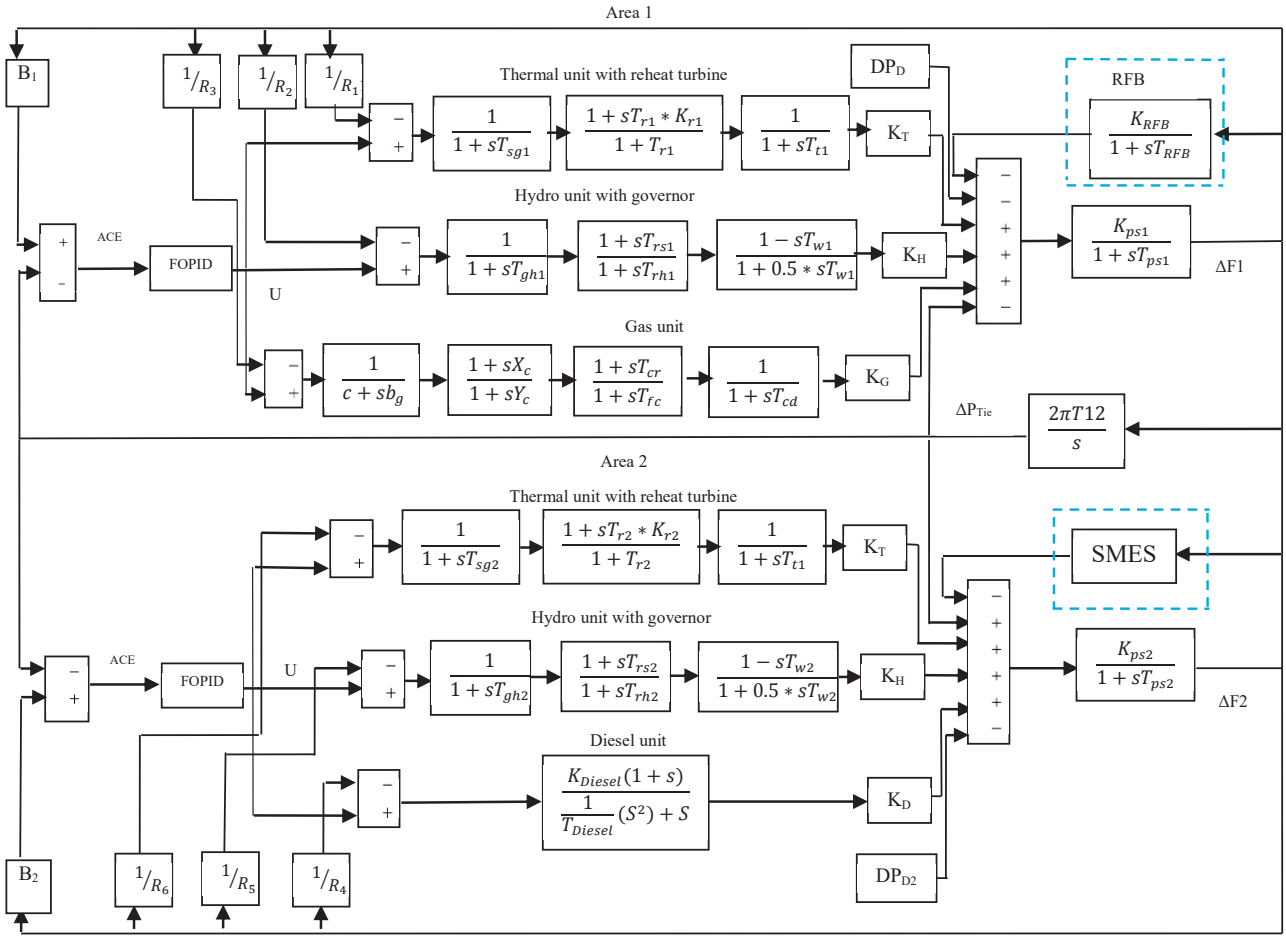


Fig. 3. Multi-source tow area interconnected power system

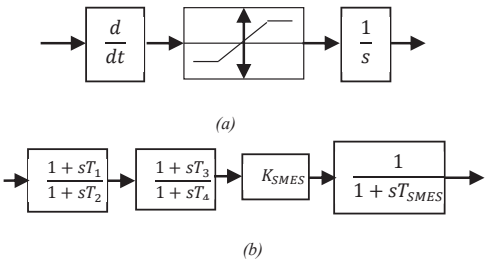


Fig.4. (a) The GRC model for thermal unit (b) The SMES model

Table 1. Value of SSO algorithm parameters for LFC problem

Parameters	Value
Population size	50
Iteration	50
Female Percent	44
Male percent	6
Upper limit of K_P, K_I and K_D	+10
Lower limit of K_P, K_I and K_D	-10
Upper limit of K_{SMES}	2
Lower limit of K_{SMES}	0
Upper limit of λ and μ	1
Lower limit of λ and μ	0
Upper limit of T_1, T_2, T_3, T_4 and T_{SMES}	1
Lower limit of T_1, T_2, T_3, T_4 and T_{SMES}	0

Table 2. Value of optimized FOPID controller parameters using SSO algorithm

		Control Parameters	Thermal	Hydro	Gas	Diesel
Model without RFB and SMES	Area 1	K_P	-8.5607	8.8461	-1.7102	
		K_I	-7.1027	-0.6398	-4.4562	
		K_D	2.3935	1.1184	-0.1753	
		λ	0.8147	0.2370	0.5135	
		μ	0.3827	0.5080	0.6390	
	Area 2	K_P	-8.5607	8.8461		3.9502
		K_I	-7.1027	-0.6398		-0.9584
		K_D	2.3935	1.1184		-8.4396
		λ	0.8147	0.2370		0.4880
		μ	0.3827	0.5080		0.0797
Model with RFB and SMES	Area 1	K_P	-4.9105	7.4214	-5.2184	
		K_I	-9.6040	-3.3650	-1.8269	
		K_D	0.9718	-0.4741	5.3892	
		λ	0.7585	0.2741	0.2723	
		μ	0.6614	0.5515	0.3467	
	Area 2	K_P	-4.9105	7.4214		1.1368
		K_I	-9.6040	-3.3650		-0.1353
		K_D	0.9718	-0.4741		-6.1911
		λ	0.7585	0.2741		0.6587
		μ	0.6614	0.5515		0.0817

Table 3. Value of SMES parameters optimized by SSO algorithm

Parameter	Value
K_{SMES}	1.9059
T_1	0.6596
T_2	0.5762
T_3	0.1250
T_4	0.2931
T_{SMES}	0.5194

Table 4. Value of ITAE and settling time and peak overshoot without RFB and SMES

Criterion	Case 1	Case 2
ITAE	0.0218	0.0194
Settling Time (0.2%)		
Δf_1	3.5262	2.2943
Δf_2	3.9261	2.7276
$\Delta P_{Tie\text{-}line}$	3.3334	2.3298
Peak Overshoot		
Δf_1	0.0193	0.0144
Δf_2	0.0157	0.0045
$\Delta P_{Tie\text{-}line}$	0.0028	0.0027

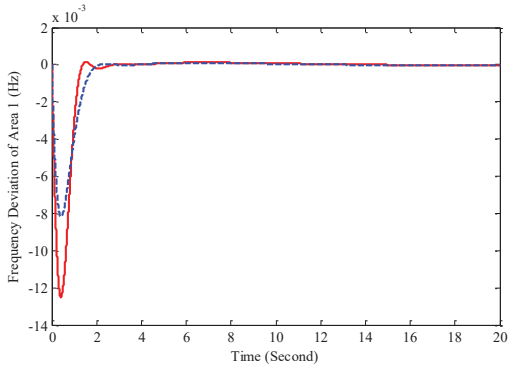


Fig. 5. Frequency deviations of area 1 for 1% step load disturbance in area 1; Solid (without FRB and SMES) and Dashed (with FRB and SMES)

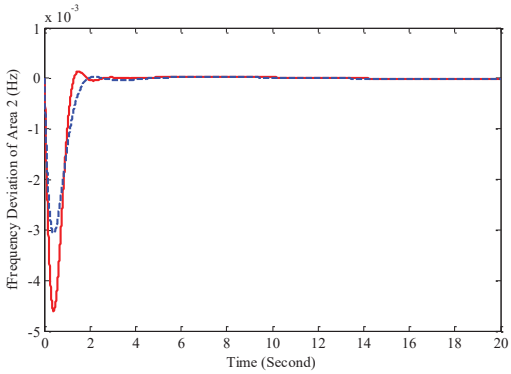


Fig. 6. Frequency deviations of area 2 for 1% step load disturbance in area 1; Solid (without FRB and SMES) and Dashed (with FRB and SMES)

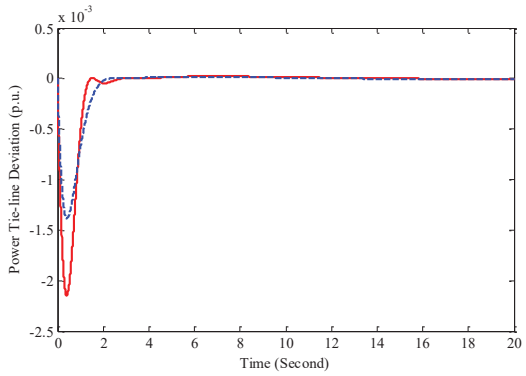


Fig. 7. Tie-line power deviations for 1% step load disturbance in area 1; Solid (without FRB and SMES) and Dashed (with FRB and SMES)

The settling time of frequency deviation of area 1 for case 1 is 3.5262 s and case 2 is 1.2943 s, same time for frequency deviation of area 2 is 3.9261 s for case 1 and is 1.7276 s and for case 2, the settling time of tie-line power deviation is 3.3334 s for case 1 and is 1.3298 s for case 2. It can be seen the high efficiency of the FOPID controller based on SSO algorithm in LFC problem and power system dynamics response has improved with considering RFB and SMES.

Case 3. To illustrate the robustness of the SSO based FOPID controller, uncertainties in the power system parameters are considered with changing in range of [-%25, +%25] from the nominal value of some parameters including T_t, T_w, T_{cd} and T_{ps}. The T_t, T_w, T_{cd} and T_{ps} is the time constant of reheat turbine, time constant of hydro turbine, gas turbine compressor discharge volume-time constant and power system time constant, respectively. The results of simulation are shown in Figs. 8 to 10.

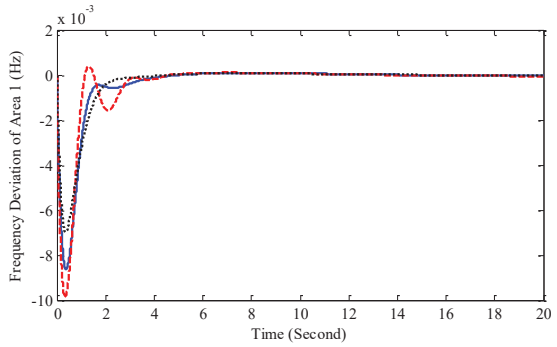


Fig. 8. Frequency deviations of area 1 for 1% step load disturbance in area 1 with RFB and SMES, Dashed (-25% of nominal value), Solid (nominal value) and Dotted (+25% of nominal value

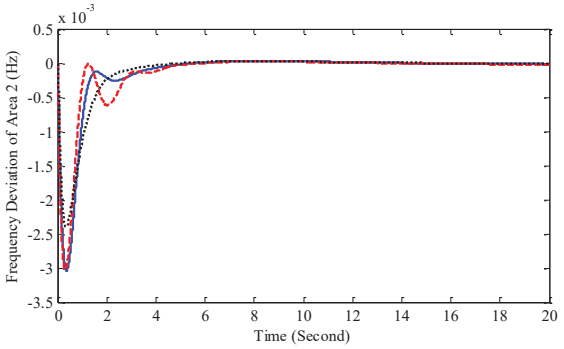


Fig. 9. Frequency deviations of area 2 for 1% step load disturbance in area 1 with RFB and SMES, Dashed (-25% of nominal value), Solid (nominal value) and Dotted (+25% of nominal value)

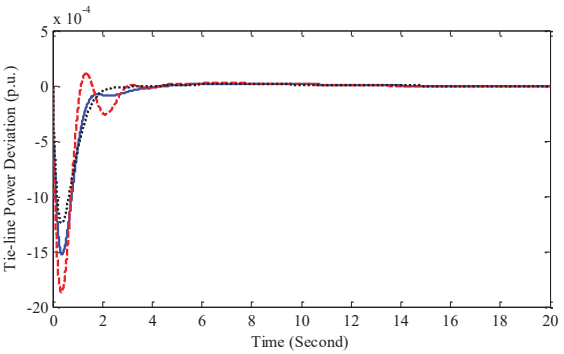


Fig. 10. Tie-line power deviations for 1% step load disturbance in area 1 with RFB and SMES, Dashed (-25% of nominal value), Solid (nominal value) and Dotted (+25% of nominal value

Table 5. Value of ITAE, settling time and peak overshoot in presence of ±25% change in system parameters

Parameters Change +25%	-25%	Nominal Value	
ITAE	0.0235	0.0194	0.0169
Δf_1 Settling Time (0.2%)	2.9222	2.2943	2.6630
Overshoot	0.0441	0.0144	0.0102
Δf_2 Settling Time (0.2%)	4.3252	2.7276	3.5196
Overshoot	0.0043	0.0045	0.0036
$\Delta P_{Tie-line}$ Settling Time (0.2%)	2.8272	2.3298	2.2067
Overshoot	0.0123	0.0027	0.0018

Based on the obtained results of Table 5 and Figs. 8-10 and comparison of ITAE index, settling time and overshoot for frequency deviations of each control area, the good performance damping of the proposed controller against various uncertainties is achieved. These outcomes show that the SSO based FOPID controller has a high ability in improvement the dynamics performance of the two area multi-source interconnected power system in presence of different operation conditions and nonlinearities such as GRC and uncertainties in the power system parameters.

6. CONCLUSION

In this paper, the FOPID controller based on SSO algorithm is proposed to solve the LFC problem in a multi-source two-area interconnected power system. The objective function is combination of ITAE and sum of settling times of the frequency deviations of each control areas and tie-line power deviations. The results demonstrates that the SSO based FOPID controller associated with RFB and SMES have a high ability in damping the frequency deviation of control areas and tie-line power deviation. The control parameters has been tuned using SSO algorithm that is capable at obtaining the optimum solutions in optimization problems. The proposed method has been applied on power system considering nonlinearity such as GRC and parameters uncertainties in power system. The results reveal high ability of the proposed control method to balance the power system frequency deviations in the presence of uncertainties, nonlinearity in power system structure and load disturbance.

APPENDIX

The value of under study system parameters in Fig. 2 are given below:

f = 60 Hz; $B_1 = B_2 = 0.4312$ pu MW/Hz; PR = 2000MW (rating), PL = 1840MW (nominal loading); $T_{r1} = T_{r2} = 10$ s; $K_{r1} = K_{r2} = 0.3$; $T_{t1} = T_{t2} = 0.3$ s; $R_1 = R_2 = R_3 = R_4 = R_5 = R_6 = 2.4$ Hz/pu MW; $T_{sg1} = T_{sg2} = 0.08$ s; $K_{T1} = K_{T2} = 0.543478$; $K_{H1} = K_{H2} = 0.326084$; $K_{G1} = K_{G2} = 0.130438$; $T_{gh1} = T_{gh2} = 0.2$ s; $T_{rh1} = T_{rh2} = 28.75$ s; $T_{rs1} = T_{rs2} = 5$ s; $T_{W1} = T_{W2} = 1$ s; $b_g = 0.5$; $c_g = 1$; $X_c = 0.6$ s; $Y_c = 1$ s; $T_{cr} = 0.01$ s; $T_{fc} = 0.23$s; $T_{cd1} = T_{cd2} = 0.2$s; $K_{Diesel} = 16.5$;

$T_{Diesel} = 1/40$; $T_{ps1} = T_{ps2} = 11.49$ s; $K_{ps1} = K_{ps2} = 68.9566$ Hz/pu MW; $T_{12} = 0.0433$ pu;

REFERENCES

[1] Shayeghi, H., H.A. Shayanfar, and A. Jalili, *Load frequency control strategies: A state-of-the-art survey for the researcher.* Energy Conversion and Management, **50**(2): pp. 344-353, 2009.

[2] Chang, C.S. and W. Fu, *Area load frequency control using fuzzy gain scheduling of PI controllers.* Electric Power Systems Research, **42**(2): pp. 145-152, 1997.

[3] Yeşil, E., M. Güzelkaya, and İ. Eksin, *Self tuning fuzzy PID type load and frequency controller.* Energy Conversion and Management, **45**(3): pp. 377-390, 2004.

[4] Tan, W., *Tuning of PID load frequency controller for power systems.* Energy Conversion and Management, **50**(6): pp. 1465-1472, 2009.

[5] Khodabakhshian, A. and R. Hooshmand, *A new PID controller design for automatic generation control of hydro power systems.* International Journal of Electrical Power & Energy Systems, **32**(5): pp. 375-382, 2010.

[6] Mosaad, M.I. and F. Salem, *LFC based adaptive PID controller using ANN and ANFIS techniques.* Journal of Electrical Systems and Information Technology, **1**(3): pp. 212-222, 2014.

[7] H. Shayeghi, H.A. Shayanfar, and A. Jalili, Multi stage fuzzy load frequency control using PSO, Energy Conversion and Management, **49**: pp. 2570-2580, 2008.

[8] Shayeghi, H., A. Jalili, and H.A. Shayanfar, *A robust mixed H_2/H_∞ based LFC of a deregulated power system including SMES.* Energy Conversion and Management, **49**(10): p. 2656-2668, 2008.

[9] Yang, X.-J., *Advanced local fractional calculus and its applications.* World Science, New York, NY, USA, 2012.

[10] Oustaloup, A., X. Moreau, and M. Nouillant, *The CRONE suspension.* Control Engineering Practice, **4**(8): pp. 1101-1108, 1996.

[11] Åström, K.J. and T. Hägglund, *Advanced PID control*. ISA-The Instrumentation, Systems, and Automation Society; Research Triangle Park, NC 27709, 2006.

[12] Podlubny, I., *Fractional-order systems and PI/sup/spl lambda//D/sup/spl mu//-controllers.* Automatic Control, IEEE Transactions on, **44**(1): pp. 208-214, 1999.

[13] Cuevas, E., et al., *A swarm optimization algorithm inspired in the behavior of the social-spider.* Expert Systems with Applications, **40**(16): pp. 6374-6384, 2013.

[14] Parmar, K.P.S., S. Majhi, and D.P. Kothari, *LFC of an interconnected power system with multi-source power generation in deregulated power environment.* International Journal of Electrical Power & Energy Systems, **57**: pp. 277-286, 2014.

[15] Sahu, R.K., T.S. Gorripotu, and S. Panda, *A hybrid DE–PS algorithm for load frequency control under deregulated power system with UPFC and RFB.* Ain Shams Engineering Journal, 6(3): pp. 893-911, 2015.

[16] Bhatt, P., S.P. Ghoshal, and R. Roy, *Coordinated control of TCPS and SMES for frequency regulation of interconnected restructured power systems with dynamic participation from DFIG based wind farm.* Renewable Energy, **40**(1): pp. 40-50, 2012.

BIOGRAPHIES

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 520 technical papers in the international journals and conferences proceedings. Dr. Shayanfar is a member of the Iranian Association of Electrical and Electronic Engineers.

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 and co-authored of 5 books in Electrical Engineering area all in Farsi, one book and two book chapters in international publishers and more than 330 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. Furthermore, he has been included in the Thomson Reuters' list of the top one percent of most-cited technical Engineering scientists in 2015 and 2016, respectively. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and senior member of IEEE.

.

Abdollah Molaee was born in Kermanshah, Iran, in 1988. He received the B.S. degree in Electrical Engineering from the Kermanshah University of Technology, Kermanshah, Iran, in 2013. He currently is M.S. Degree student in Electrical Engineering from University of Mohaghegh Ardabili, Ardabil, Iran. His areas of interest in research are the Power System Restructuring and application of heuristic optimization methods and robust control design to power system control

PID Type Stabilizer Design Using Grey Wolfe Optimization Algorithm

H. A. Shayanfar[1]* , H. Shayeghi[2], A. Younesi[2]
[1] Center of Excellence for Power System Automation and Operation, School of Electrical Engineering, Iran University of Science & Technology, Tehran, Iran
[2] Electrical Engineering Department, University of Mohaghegh Ardabili, Ardabil, Iran
hshayanfar@gmail.com, hshayeghi@gmail.com, a.younesi@ieee.org

Abstract – *In this paper Grey Wolfe (GW) optimization algorithm is proposed for optimal tuning of PID based stabilizer. The time domain based objective function is considered for the robustly PID parameters tuning problem under a wide range of operating conditions. The optimization problem is solved by the GW Algorithm (GWA) which provides robust and desirable solutions. Results evaluation on a Single Machine connected to Infinite Bus (SMIB) confirm an excellent capability of the GWA based tuned PID to stabilize power system in comparison to the invasive weed optimization based PID controller for different operating conditions. Also, overshoot, undershoot, settling time and some system performance indices are calculated for two PID based stabilizers, and results prove that the proposed GWA method is better than IWA based PID stabilizer.*

Keywords: Grey Wolfe Algorithm, Power System Stabilizer, SMIB Power System

1 Introduction

Stability of power system is one of the most important aspects in electric system operation. This arises from the fact that the power system must maintain frequency and voltage levels at the nominal values, under any disturbance, like a sudden increase in the load, loss of one generator or switching out of a transmission line during a fault [1]. Since the development of interconnected large electric power system, there have been spontaneous system oscillations at very low frequencies in the order of 0.2-3.0 Hz. Once started, they would continue for a long period of time. In some cases, they continue to grow, causing system separation if no adequate damping is available. Moreover, low frequency oscillations present limitations on the power-transfer capability. To enhance system damping, the generators are equipped with Power System Stabilizer (PSS) that provide supplementary feedback stabilizing signals in the excitation system. PSS augment the power system stability limit and extend the power-transfer capability by enhancing the system damping of low frequency oscillations associated with the electromechanical modes [2].

The conventional lead-lag compensators have been widely used as the PSSs.

* Corresponding Author: hashayanfar@gmail.com (H. A. Shayanfar)

However, the problem of PSS parameter tuning is a complex exercise. These stabilizers have previously tuned both single and multiple operation points of the power system using various methods. The approaches used to the problem of PSS parameters tuning range from modern control theory [2-8], to the more recent one using many random heuristic methods, such as like Tabu search, genetic algorithms, chaotic optimization algorithm, rule based bacteria foraging and particle swarm optimization for achieving high efficiency and search global optimal solution in the problem space [9-12].

Despite the potential of the modern control techniques with different structure, Proportional Integral Derivative (PID) type controller is still widely used for industrial applications such as power system control [13-15]. This is because it performs well for a wide class of process. Also, they give robust performance for a wide range of operating conditions and easy to implement. On the other hand, Shayeghi and Esmailnezhad [13] represented a comprehensive analysis of the effects of the different PID controller parameters on the overall dynamic performance of the PSS problem. 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 system operation. Hence, local optimization techniques, which are well elaborated upon, are not suitable for such a problem. For this reason, a new meta-heuristic algorithm, called Grey Wolfe (GW) is proposed for optimal tuning of the PID controller gains to stabilize a synchronous machine connected to an infinite bus in this paper. The GW algorithm is based on the grey wolves. In comparison with other nature-inspired algorithms it carefully mimics the leadership hierarchy and hunting mechanism of the grey wolves and uses the real world empirical studies to obtain the coefficients and only time interval parameter should be fine-tuned in the GW algorithm. Despite the other heuristic techniques, the GW algorithm has a flexible and well-balanced mechanism to enhance the global and local exploration abilities [16].

In this study, the problem of robust PSS design is formulated as an optimization problem and GW technique is used to solve it. A performance index is defined based on the system dynamics after an impulse disturbance alternately occurs in the system and used to form the objective function of the design problem. The proposed GW based tuned PID controller as a stabilizer has been applied and tested on a weekly connected power system under wide range of operating conditions to illustrate their ability to provide efficient damping of low frequency oscillations. To show the superiority of the proposed design approach, the simulations results are compared with the Invasive Weed Optimization (IWO) based designed PID controller [13] under different operating conditions through some performance indices. The results evaluation show that the proposed method achieves good robust performance for wide range of load changes in the presence of very highly disturbance and Is superior to the other stabilizers.

2 Power System Model

For the stability analysis of power system, adequate mathematical models describing the system are needed. The models must be computationally efficient and able to represent the essential dynamics of the power system. A schematic diagram for the test system is shown in Fig. 1. The generator is equipped with the excitation system and a power system stabilizer. System data are given in the Appendix.

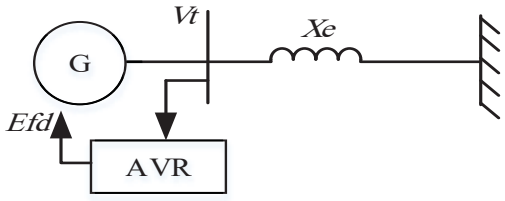


Fig. 1. SMIB power system

The synchronous generator is represented by model 1.1, i.e. with field circuit and one equivalent damper winding on the q axis (4^{th} – order model).

A linear dynamic model by linearizing the nonlinear model around an operating condition can be expressed as follows [15]:

$$\begin{aligned}
\Delta\dot{\delta} &= \omega_0 \Delta\omega \\
\Delta\dot{\omega} &= (-K_1\Delta\delta - D\Delta\omega - K_2\Delta E'_q)/M \\
\Delta\dot{E}'_q &= (-K_4\Delta\delta - \Delta E'_q/K_3 + \Delta E_{fd})/T'_{do} \\
\Delta\dot{E}_{fd} &= (-K_A K_5\Delta\delta - K_A K_5\Delta E'_q - \Delta E_{fd} + K_A V_S)/T_A
\end{aligned} \tag{1}$$

Where, $K1$, $K2$,...$K6$, are Heffron-Phillips constants and are given in [14]. Equation (1) can be rewritten in the state space form as follows:

$$\dot{x} = Ax + Bu \tag{2}$$

Where, A and B are the system and input matrices. The block diagram of the linearized dynamic model of the SMIB power system is shown in Fig. 2.

2.1 PSS Model

The structure of the PSS, to modulate the excitation voltage is shown in Fig . 2 (hatch blocks). The structure consists of a signal washout block and a PID controller as opposed to the traditional lead-lag controller. The input signal of the proposed method is the speed deviation ($\Delta\omega$) and the output is the stabilizing signal V_S which is added to the reference excitation system voltage. The signal washout block serves as a high-pass filter, with the time constant T_W, high enough to allow signals associated with the oscillations in input signal to pass unchanged. From the viewpoint of the washout function, the value of T_W is not critical and may be in the range of 1 to 20 seconds [12]. The PID controller block (gains K_p, K_i and K_d) provides the appropriate phase-lead characteristics to compensate for the phase lag between input and the output signals.

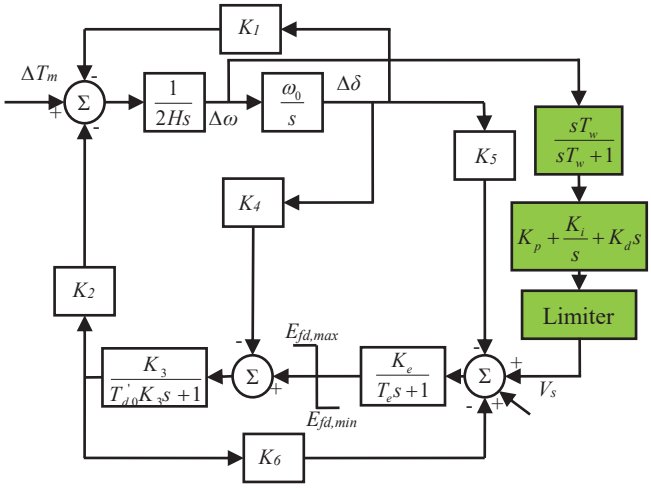


Fig. 2. Heffron-Phillips transfer function model

The augmented A matrix with incorporating the structure of the PID power system stabilizer in the system matrix as given in (2) can be expressed as follows:

$$A = \begin{bmatrix}
0 & \omega_0 & 0 & 0 & 0 \\
\frac{-K_1}{M} & 0 & \frac{K_2}{M} & 0 & 0 \\
\frac{-K_4}{T'_{d0}} & 0 & \frac{-1}{K_3 T'_{d0}} & \frac{1}{T'_{d0}} & 0 \\
\frac{MK_e(-K_5 + M^{K_i}/\omega_0 - K_d K_1)}{MT_e} & \frac{K_e K_p}{T_e} & \frac{-MK_6 K_e - K_2 K_e K_d}{MT_e} & \frac{-1}{T_e} & \frac{K_e}{T_e} \\
\frac{M^{K_i}/\omega_0 - K_d K_1}{MT_W} & \frac{K_p}{T_W} & \frac{-K_2 K_d}{MT_W} & 0 & \frac{-1}{T_W}
\end{bmatrix} \tag{3}$$

3 Grey Wolfe Optimization Algorithm

3.1 Inspiration

Grey wolf belongs to canidae family. Grey wolves mostly prefer to live in a pack. The group size is 5-12 on average. They are divided in four groups. The leaders are a male and a female, called alphas. The alpha is mostly responsible for making decisions about hunting, sleeping, place, time to wake, and so on. The alpha's decisions are dictated to the pack. The second level in the hierarchy of grey wolves is beta. The betas are subordinate wolves that help the alpha in decision making or other pack activities. The beta wolf can be either male or female, and he/she is probably the best candidate to be the alpha in case one of the alpha wolves passes away or become very old. The lowest ranking grey wolf is omega. The omega plays the role of scapegoat. Omega wolves always have to submit to all the other dominant wolves. They are the last wolves that are allowed to eat. If a wolf is not an alpha, beta, or omega, he/she is called delta. Delta wolves have to submit to alphas and betas, but they dominate the omega. In addition to the social hierarchy of wolves, group hunting is another interesting social behavior of grey wolves. Grey wolf hunting are as three stages as follows:

- Tracking, chasing, and approaching the prey.
- Pursuing, encircling, and harassing the prey until it stops moving.
- Attack towards the prey.

In the GW algorithm hunting technique and the social hierarchy of grey wolves are mathematically modeled.

3.2 Mathematical model and algorithm

In GW algorithm, the fittest solution considered as alpha (α). Consequently, the second and third best solutions are named beta (β) and delta (δ), respectively. The rest of the candidate solutions are assumed to be omega (ω). Also, in GW the hunting (optimization) is guided by α, β, and δ. The ω wolves follow these three wolves. Hunting procedure are modeled as follow:

$$\begin{aligned} \vec{D}_\alpha &= \left| \vec{C}_1.\vec{X}_\alpha(t) - \vec{X}(t) \right| \\ \vec{D}_\beta &= \left| \vec{C}_2.\vec{X}_\beta(t) - \vec{X}(t) \right| \\ \vec{D}_\delta &= \left| \vec{C}_3.\vec{X}_\delta(t) - \vec{X}(t) \right| \end{aligned} \quad (4)$$

$$\begin{aligned} \vec{X}_1(t+1) &= \vec{X}_\alpha(t) - \vec{A}_1.\vec{D}_\alpha \\ \vec{X}_2(t+1) &= \vec{X}_\beta(t) - \vec{A}_2.\vec{D}_\beta \\ \vec{X}_3(t+1) &= \vec{X}_\delta(t) - \vec{A}_3.\vec{D}_\delta \end{aligned} \quad (5)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1(t+1) + \vec{X}_2(t+1) + \vec{X}_3(t+1)}{3} \quad (6)$$

Where, t indicates the current iteration, A and C are coefficient vectors, X_p is the position vector of the prey, and X indicates the position of grey wolf. The vectors A and C are calculated using Equations (7) and (8).

$$\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a} \quad (7)$$

$$\vec{C} = 2.\vec{r}_2 \quad (8)$$

Where, components of a are linearly decreased from 2 to 0 over the course of iterations and r_1, r_2 are random vectors in [0,1]. The pseudo code of the GW algorithm shown in Table 1. For more details about GW algorithm see [16].

Table 1. Pseudo code of the GW algorithm

Initialize the grey wolf population X_i (i = 1,2,3,…,n)
Initialize a, A, and C
Calculate the fitness of each search agent
X_α = The best search agent
X_β = The second best search agent
X_δ = The third best search agent
***While** (t< Max number of iterations)*
  ***for** each search agent*
    Update the position of the current search agent
  end for
  Update a, A, and C
  Calculate the fitness of all search agents
  Update X_α, X_β, and X_δ
  t = t+1
End while
Best solution = X_α

4 Problem Formulation

In case of the PID structured PSS (Fig. 3), the washout time constants is usually specified. In the present study, washout time constant T_W =5 sec is used. For the optimal tuning of the PSS in Fig. 3, the PID gains are to be determined. It is worth mentioning that the PSS is designed to minimize the power system oscillations after a large disturbance so as to improve the overall system dynamic stability in a robust way under different operation and disturbances. These oscillations are reflected in the deviations in power angle, rotor speed and line power. It was shown that the system is unstable for moderately high loading conditions [14]. To increase system damping to electromechanical modes, an time simulation based objective function for the multiple operation conditions is considered as follows:

$$J = \sqrt{\sum_{i=1}^{NP} \left(1000 \int_0^{10} t \left|\Delta\omega_i\right| dt\right)^2}, i = 1,2,....,NP \quad (9)$$

Where, NP is the total number of operating points for which the optimization is carried out. In the optimization process, it is aimed to minimize J in order to damping low frequency oscillations under all operation points. The multiple operating conditions for the above system are completely defined by the values of real power, P and the reactive power, Q, at the generator terminals. P and Q are assumed to vary independently over the following ranges:

- P changes from 0.1 to 1.0 by step 0.1

- Q : -0.3 from to 1.0 by step 0.1

This encompasses almost all practically occurring operating conditions.

The design problem can be formulated as the following constrained optimization problem, where the constraints are the PID gains and PSS limiter bounds [17]:

Minimize J Subject to:

$$\begin{aligned} &0 \le K_p \le 50 \\ &0 \le K_i \le 50 \\ &0 \le K_d \le 10 \\ &0.01 \le U_{PSS,\max} \le 0.15 \\ &-0.15 \le U_{PSS,\min} \le -0.01 \end{aligned} \quad (10)$$

The proposed approach employs GWA to solve this optimization problem and search for an optimal or near optimal set of PID parameters. The optimization of the PID parameters is carried out by evaluating the objective cost function as given in Equation (9), which considers a multiple of operating conditions that is completely defined by the values of real power, P and the reactive power, Q, at the generator terminals. The operating conditions are considered for wide range of output power at different power factors. This encompasses almost all practically occurring operating conditions. In order to acquire better performance the GWA parameters used to tune PID gains are listed in Table 2.

Table 2. GW Parameters values for PID based PSS

Symbol	Quantity	Value
Search Agents No.	Number of grey wolves	20
Max_iter	Maximum number of iterations	200
Dim	Number of variables	5

Results of the PID parameter set values based on the objective function J, by applying a step change of 0.1 pu in the input mechanical torque of generator using the proposed GW and IWO algorithms (see Ref. [13] for more details) are given in Table 3. Fig. 4 shows the minimum fitness functions evaluating process.

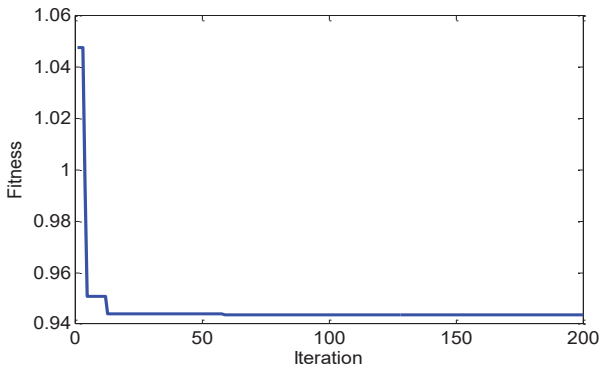


Fig. 3. Fitness Convergence

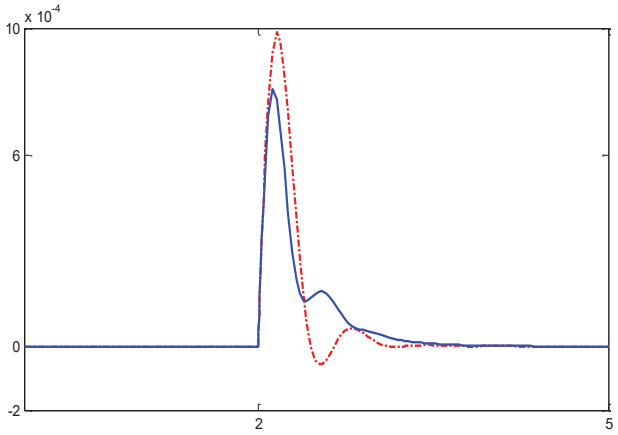
Table 3. Optimal PID gains

Algorithm	K_p	Ki	K_d	U_{pss_max}	U_{pss_min}
GW	49.5972	9.5890	9.8996	0.1492	-0.1211
IW	35.2940	8.9021	6.3197	-	-

5 Simulation Results

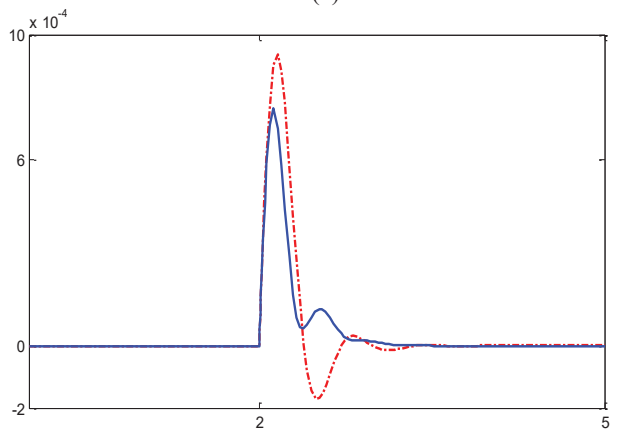
The behavior of the proposed GWO based designed PSS (GWOPSS) under transient conditions is verified by applying disturbance under different operating conditions in comparison with the IWO based tuned PSS (IWOPSS). System responses in the form of speed deviation are plotted. For disturbance a step change of 0.1 pu in the input mechanical torque are considered.

Fig. 4 shows the system response at the lagging power factor operating conditions. It can be seen that both stabilizers are able to damp the oscillations reasonably well and stabilize the system at all operating conditions and GWO based tuned



(a) P = 0.9, Q = 0.2

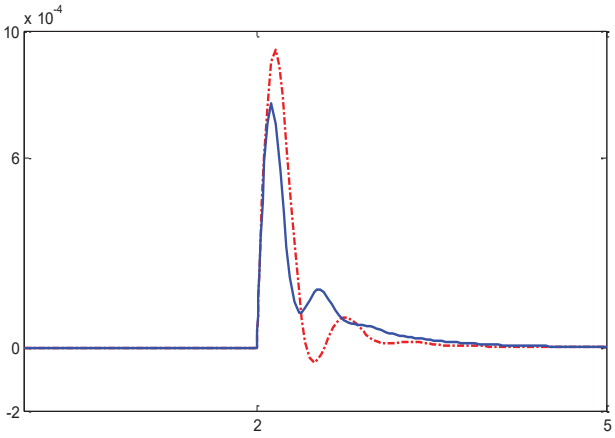
(b)



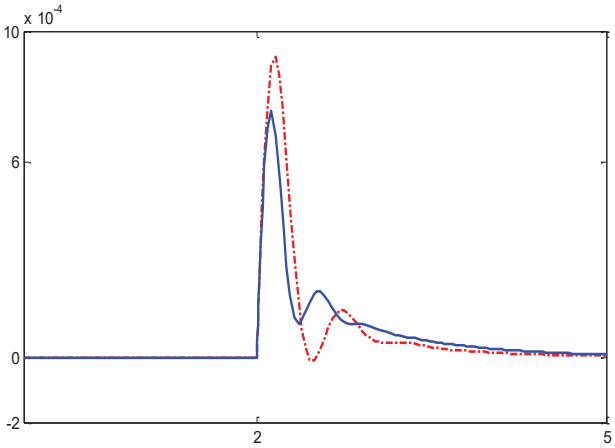
(c) P = 1.2, Q = 0.1

Fig. 4. Speed deviation at lagging power factors; GWOPSS (Solid) and IWOPSS (Dashed)

PSS shows its superiority over IWOPSS. Fig. 5 depicts the responses of the system at the leading power factor operating conditions. It can be seen that the proposed GWO based PSS



(d) P = 0.8, Q = -0.1



(e) P = 0.6, Q = -0.3

Fig. 5. Speed deviation at leading power factors; GWOPSS (Solid) and IWOPSS (Dashed)

has good performance in damping low frequency oscillations and stabilizes the system quickly.

To demonstrate performance robustness of the proposed method, four performance indices: overshoot, undershoot, the Integral of the Time multiplied Absolute value of the Error (ITAE), and ISTSE calculated and compared with IWOPSS.

ITAE and ISTSE based on the system performance characteristics are defined as [17]:

$$ITAE = 1000 \times \int_0^5 t\,|\Delta\omega| dt \tag{11}$$

$$ISTSE = 10^6 \times \int_0^5 t^2 \Delta\omega^2 dt \tag{12}$$

Numerical results of performance robustness for four operating conditions are shown in Figs. 6-9. It can be seen that the values of these system performance characteristics with the proposed GWO based tuned PSS are much smaller compared to that of IWO based designed PSS. This demonstrates that the overshoot, undershoot, and speed deviations of machine is

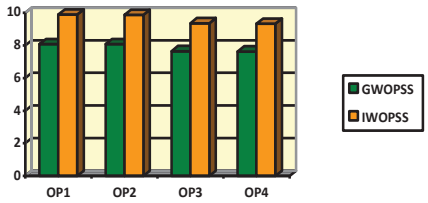


Fig. 6. OS [%]

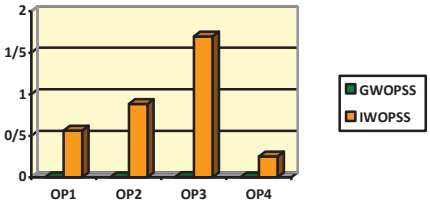


Fig. 7. US [%]

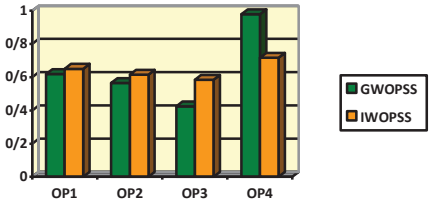


Fig. 8. ITAE

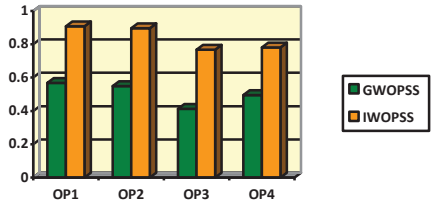


Fig. 9. ISTSE

greatly reduced by applying the proposed GWO based tuned PSS. Four operation condition are as follows:

Table 4. Different operating points data

O.P	1	2	3	4
Value	P=0.9 Q=0.2	P=1 Q=0.3	P=1.2 Q=-0.1	P=0.7 Q=-0.2

6 Conclusion

The grey wolf algorithm has been successfully applied to design of PID type stabilizer in a SMIB power system. For

this reason, the PID gains tuning problem is converted into an optimization problem based on the time domain objective function which was solved by the GW technique considering different operation conditions to improve the overall system dynamic stability in a robust way. The simulation results under wide range of operating conditions show the superiority and robustness of the GW algorithm for the PID type stabilizer design and their ability to provide efficient damping of low frequency oscillations in comparison with the IWO method. The system performance characteristics in terms of ITAE and ISTSE indices reveal that using the proposed stabilizer the dynamic performance of the system such as overshoot and undershoot are greatly reduced under various severe disturbances.

APPENDIX: SYSTEM DATA

Generator: R_a=0, x_d= 1.6, x_q = 1.55, x'_d=0.32, x'_q=0.32 , f = 50 Hz, T'_{do}=6, T'_{qo}=0.75, M=10,
Transmission line: R= 0, x_e = 0.4.
Exciter: K_A=50, T_A=0.05, E_{fdmax}=7.0, E_{fdmin}=-7.0.

References

[1] A. Safari, R. Jahani, M. Hajinasiri, A.H. Araskalaei and H.A. Shayanfar, "COA based coordinated design of FACTs and PSS output feedback controllers", International Review of Electrical Engineering, vol. 21, pp. 2803-2808, 2010.

[2] K. R. Padiyar, "Power System Dynamics, Stability and Control". Second Ed., 2008.

[3] D.K. Chaturvedi and O.P. Malik, "Experimental studies of a generalized neuron based adaptive power system stabilizer", Applied Soft Computing, vol. 11, pp. 149-155, 2007.

[4] J. Fraile-Ardanuy and P.J. Zufiria, "Design and comparison of adaptive power system stabilizers based on neural fuzzy networks and genetic algorithms", Neurocomputing , vol. 70, pp. 2902-2912, 2007.

[5] R. Segal, A. Sharma and M.L. Kothari, "A self-tuning power system stabilizer based on artificial neural network", Electrical Power and Energy Systems, vol. 26, pp. 423-30, 2004.

[6] R. Ghanizadeh, A. Jahandideh, M. Ebadian, M. Golkar, A. Ajami,"A multi-objective HBMO-based new FC-MRC compensator for damping of power system Oscillations", Journal of Operation and Automation in Power Engineering, vol.1, no.2, pp.110-123, 2013.

[7] N. Bigdeli, E. Ghanbaryan, K. Afshar,"Low frequency oscillation suppression via CPSO based damping controller", Journal of Operation and Automation in Power Engineering, vol.1, no.1, pp.22-32, 2013.

[8] K. Mazlumi, M. Darabian, M. Azari," Adaptive fuzzy synergetic PSS design to damp power system oscillations", Journal of Operation and Automation in Power Engineering, vol.1, no.1, pp.43-53, 2013.

[9] H. Shayeghi, H. A. Shayanfar, A. Akbarimajd and A. Ghasemi," PSS design for a single-machine power system using honey bee mating optimization", In: Proceedings of the International Conference on Artificial Intelligence, Las Vegas, USA, pp. 210-216, 2011.

[10] H. Shayeghi, H.A. Shayanfar and A. Ghasemi, "A robust ABC based PSS design for a SMIB power system" International Journal on Technical and Physical Problems of Engineering vol. 3, pp. 86-92, 2011.

[11] H. Shayeghi, H.A. Shayanfar, S. Jalilzadeh and A. Safari, "Multi-machine power system stabilizers design using chaotic optimization algorithm", Energy Conversion and Management, vol. 51, pp. 1572-1580, 2010.

[12] M. Eslami, H. Shareef, A. Mohamed and M. Khajehzadeh, "Damping of power system oscillations using genetic algorithm and particle swarm optimization", International Review of Electrical Engineering, vol. 5, pp. 2745-2753, 2010.

[13] H. Shayeghi and B. Esmaeilnezhad, "PID controller design for power system stabilization", Proc. of the 7th International Conference on Technical and Physical Problems of Engineering, Lefkosa, TR Northern Cyprus, pp. 252-256, July 2011.

[14] S. Duman and A. Öztürk , "Robust design of PID controller for power system stabilization by using real coded genetic algorithm", International Review of Electrical Engineering, vol. 5, pp. 2159-2170, 2010.

[15] H. Shayeghi and A. Ghasemi, "Market based LFC design using artificial bee colony", International Journal on Technical and Physical Problems of Engineering, vol. 3, pp. 1-10, 2011.

[16] S. A. Mirjalali, S. M. Mirjalali, A. Lewis, Grey Wolf Optimizer, Advances in Engineering Software, vol. 69, pp. 46-61, 2014.

[17] H. Shayeghi, A. Ghasemi, H.A. Shayanfar, PID type stabilizer design for multi machine power system using IPSO procedure, Computer Science and Engineering, vol. 1, pp. 36-42, 2011.

Biographies

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, reactive pricing in deregulated power systems and smart grids. He has published more than 520 technical papers in the international journals and conferences proceedings. Dr. Shayanfar is a member of the Iranian Association of Electrical and Electronic Engineers.

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 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, power system restructuring and smart grids. He has authored and co-authored of 5 books in Electrical Engineering area all in Farsi, one book and two book chapters in international publishers and more than 330 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. Furthermore, he has been included in the Thomson Reuters' list of the top one percent of most-cited technical Engineering scientists in 2015 and 2016, respectively. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and senior member of IEEE.

Abdollah Younesi received B.S and M.S.E degrees both in Electrical Engineering from Faculty of Technical Eng. Department of the Mohaghegh Ardabili University, Ardabil, Iran in 2012 and 2015, respectively. Currently He is a PhD. 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.

Optimization of 22 nm Logic Gates for Power-and-Noise-Margin and Energy-and-Noise-Margin

Azam Beg[1], **Rashad Ramzan**[2], **and Amr Elchouemi**[3]
[1]College of Information Technology, United Arab Emirates University, Al-Ain, United Arab Emirates
[2]College of Engineering, United Arab Emirates University, Al-Ain, United Arab Emirates
[3]Walden University, USA

Abstract— *In this paper, we propose a technique for concurrent optimization of CMOS logic gates for power-and-noise-margin and energy-and-noise-margin. The role of progressive sizing for performance enhancement of different gates has been expanded to cover other figures of merit, such as reliability, power, and energy. By using the examples of three- and four-input logic gates, we have demonstrated how multiple, yet conflicting design goals can be achieved. For example, one of our high-performance gates exhibited power savings of more than 30% while reducing the gate area by 39%. An important step of balancing the rise- and fall-times of output was also incorporated into the optimization setup. Our proposed methodology is scalable and can be used for optimizing larger logic blocks.*

Keywords: CMOS technology, logic circuits, combinational circuits, genetic algorithms, multi-objective optimization, transistor sizing

1. Introduction

With the explosive growth of mobile electronic devices in the recent years, the integrated circuit designers must pay special attention to power and energy efficiency, rather than just the performance (clock frequency, f_{clk}). Although continuous downsizing of transistors has helped the chip manufacturers maintain Moore's Law, the dynamic and the leakage powers have not been able to keep up with the pace of transistor-scaling. Yet another important challenge for the modern devices is their reliable operation.

This paper for the first time proposes the use of *genetic algorithms* (GAs) for *multi-objective* optimization of CMOS logic gates based on 22 nm technology. The technique aims to help a circuit designer find transistor sizes that meet a logic gate's design criteria for power/energy dissipation, static noise margin (SNM) (one way of representing reliability), and balanced rise- and fall-times (τ_{rise} and τ_{fall}), etc.

This paper is organized as follows: Section 2 covers basic information about the logic gates' metrics and optimization. Section 3 covers the previous work related to this paper, and Section 4 explains our experimental setup. Results and analysis are presented in Section 5. Finally, Section 6 concludes the paper.

2. Preliminaries

2.1 The Metrics for CMOS Logic Gates

The power dissipation of a CMOS gate constitutes two parts, *dynamic* and *static*. The static power consumption can be represented with a simple equation: $P_{static} = I_{off}V_{DD}$, where I_{off} is the non-ideal and undesired current flowing between the supply rails (V_{DD} and ground) even when the gate is not switching. With a high f_{clk}, most of the dynamic power consumption (P_D) is due to the charging and discharging of the load capacitance (C_L): $\alpha_{0\to1}f_{clk}C_LV_{DD}^2$, where $\alpha_{0\to1}$ is the probability of 0→1 transition. The second component of P_D occurs while both the N- and P-transistors are on. For example, in a matched inverter, the peak current flows when $v_{in} = v_{out} = V_{DD}/2$. As power and delay can be adjusted separately, it is common to consider them together by using the *power-delay product* (PDP or energy) (at a given f_{clk}) as a figure-of-merit (FOM) [1].

An obvious method for reducing P_D is to lower either or both the f_{clk} and the V_{DD}. However, lowering power consumption while ignoring f_{clk} may not be desirable. This is because reduced frequency increases the time taken to complete a task. Therefore, for an energy-efficient design, it is not enough to minimize just the power; what matters more is the saving in energy per cycle of operation [1].

The ability of a gate to withstand noise has been traditionally defined as the *noise margin* (NM). The allowed voltage range for logic-*low* is called the low NM and the range for the logic-*high* is high NM. The lower of the two NMs is referred to as SNM. Properly balancing the transistor sizes in the N- and P-stacks of a CMOS gate can help improve the SNM [2]. In this paper, SNM is used as a measure of reliability.

2.2 Circuit Design and Optimization

In modern nanometeric-device-based circuits, the design FOMs include not only the power, energy, and delay, but also the reliability. Such characterization generally involves determining proper sizes for the transistors. In most cases, a designer has to find the sizes that balance conflicting *objectives*, such as power, delay, reliability, etc [3]. Therefore, we can consider the sizing optimization to be *multi-objective*

in nature; the GAs are just the right tool to handle such optimization [4], [5].

The GAs are based on the principle of *natural selection* that is derived from the concept of biological evolution. A GA begins the optimization process using a randomly generated *population* of *parents*; the GA then uses the principles of random *crossover* and *mutation* to find the next *generation* of 'fitter' *children* (or *offsprings*). This is in contrast to a classical algorithm that would select the *next* points in the search space deterministically. A GA relies on the search of a wide but finite solution space. A GA's input-set includes the *fitness function* (also called *cost function*), while the output is considered the *cost of fitness*. A candidate solution to a GA problem is a *chromosome*. The chromosomes are made up of *genes*; the genes represent different design parameters as strings of 0's and 1's [4], [5].

The process of multi-objective optimization using GAs often yields multiple *optimal* solutions which can be quite different from each other, but they are all deemed 'as good as possible.' Dealing with multiple design objectives means that a designer has a choice to pick the *best* optimal solution based on his own intuition and/or experience.

3. Related Work

Power consumption and reliability are listed among the 'design difficult challenges' in the current International Technology Roadmap for Semiconductors [6]. The main techniques for power reduction, in the past, have included: transistor optimization [7], near-V_{TH} or sub-V_{TH} region operation [8], use of MOS current mode logic [9], dynamic voltage scaling [10], using thin-body fully-depleted SOI [7], auto-substrate biasing [11], and transistor stacking [12].

The transistor sizing methods have been revisited, from time to time, with the purpose of addressing not just the power consumption but also energy dissipation, performance, and/or reliability [3], [13]. The traditional sizing techniques used analytical models [14], [15], [16], [17] or circuit simulations, while the non-conventional methods have included the use of constraint-based optimization [18], geometric programming [19] and *single*-objective GAs [20], [21]. Use of set-oriented numerical method for *multi*-objective sizing of 90 nm gates operating at nominal V_{DD} using GAs was proposed in [22]. Sub-V_{TH} operation of a few basic 65 nm gates (INV, NAND2, NOR2) was considered in [23] and Minority-3 function in [24]. The modern nodes such as 22 nm are much more prone to failure due to variations than the earlier ones, and have to be investigated anew.

One of the methods of sizing the transistors in a CMOS gate is *progressive sizing* [1] and it entails widening the transistors that are closer to the supply rails. By doing so, both the resistances and capacitances of the transistor stacks decrease, and hence the delay of large fan-in gates. To the best of our knowledge, we are the first ones to investigate multi-objective GAs for progressively sizing the 22 nm transistors for logic gates, with the purpose of reducing not just the delay and power/energy consumption but also for improving the SNM (or reliability).

4. Experimental Setup

In this paper, we investigate multi-objective GA-based optimization of progressively-sized transistors for four different logic gates: NAND3, NAND4, NOR3, and NOR4 (see Figs. 1 and 2). The lower and the upper bounds for all transistor widths (*w*'s) were set as $44\ nm \leq w_i \leq 1000\ nm$, and the lengths (*w*'s) were fixed at $L_{min} = 22\ nm$. In order to ensure that the parallel transistors have the same widths during optimization, we used the *equality constraints*. Progressive sizing of series transistors was enforced using the *inquality constraints*. The two constraint sets are shown in Table 1.

We set up experiments to minimize these two-objective (fitness) functions: (1) power and the difference between τ_{rise} and τ_{fall}; and (2) PDP and the difference between τ_{rise} and τ_{fall}. We aimed to match τ_{rise} and τ_{fall} as practical as possible, in order to improve the SNM and hence the operational reliability. The optimization process results in a *Pareto-front* that represents multiple optimum solutions, from which a designer can select the one that meets yet another criterion, such as the SNM or the area.

In our optimization problem, the genes represent the widths (w_i) of the transistors of *units-under-test* (UUTs). The chromosomes made from the genes are utilized in two separate SPICE circuit files created automatically by a Perl script. One circuit is for measuring the current (and power),

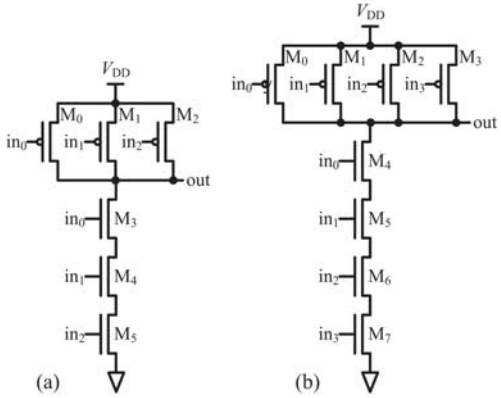


Fig. 1: Gate schematics: (a) NAND3, and (b) NAND4.

Table 1: GA constraints for transistor M_i's (w_i = width of M_i, as shown in Figs. 1 and 2)

Gate Type	Equalities	Inequalities
NAND3	$w_0 = w_1 = w_2$	$w_3 \leq w_4 \leq w_5$
NAND4	$w_0 = w_1 = w_2 = w_3$	$w_4 \leq w_5 \leq w_6 \leq w_7$
NOR3	$w_3 = w_4 = w_5$	$w_0 \geq w_1 \geq w_2$
NOR4	$w_4 = w_5 = w_6 = w_7$	$w_0 \geq w_1 \geq w_1 \geq w_3$

delay, and τ_{rise}/τ_{fall}, and the other circuit for SNM. It is worth mentioning that the difference between τ_{rise} and τ_{fall} is highly dependent on the input vectors. Ref. [25] includes an example of set of input vectors for a large fan-in gate. As it would not be practical to perfectly match the two τ's for all (C) different vectors, we defined an FOM called *average error* (ϵ_{avg}):

$$\epsilon_{avg} = \left| \sum_{i=1}^{C} \left(\frac{V_{DD}}{2} - v_{out_i} \right) / C \right|, \quad (1)$$

where v_{out_i} is the gate output voltage in response to the ith input vector.

For all gate types, we measured total power, delay, ϵ_{avg}, and SNM. We also recorded the gate area which was defined as $L_{min} \times \sum_{i=0}^{M-1} w_i$; here M = number of gate transistors, and $L_{min} = 22\ nm$. The gates were built from 22 nm PTM HP v2.1 (high-k/metal gate and stress effect) MOS transistor models [26] and BSIM4v4.7 level 54 [27]. The circuits operated at 2 GHz. As we focused only on the high performance operation of the gates, we set V_{DD} to be the same as the nominal voltage, i.e., 0.8V [26]. In other words, near- or the sub-V_{TH} modes were not considered.

Our experimental setup can be represented by the flowchart of Fig. 3. The setup is made up of a set of custom Matlab [28] and Perl scripts running under Mac OS X ver. 10.10.5. The GA population size is set to $15 \times N$. The crossover fraction has a value of 0.8. The generation count is set to 500 [28]. The starting point of the design process is a randomly generated set of chromosomes used to create SPICE circuits. The circuits are simulated using NGSpice [**?**] and the log files are saved. We then parse the log files for current, delay, SNM, etc., in order to evaluate the fitness function. We continue to *evolve* offsprings for creation and simulation of circuits, until the preset fitness criteria (power-and-$|\tau_{rise}-\tau_{fall}|$, or PDP-and$|\tau_{rise}-\tau_{fall}|$) are met or when maximum number of generations have been created. Typically, it took 4–5 compute-hours to optimize a gate. From the list of the optimal solutions, we only considered the ones that had SNM $\geq 25\%$ of V_{DD} and $\epsilon_{avg} \leq 10\%$ of V_{DD} (whenever feasible). Lastly, we sorted the solution-sets on the gate areas.

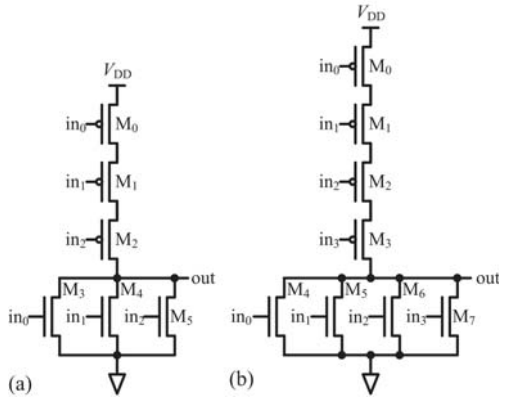


Fig. 2: Gate schematics: (a) NOR3, and (b) NOR4.

5. Results and Analysis

We performed multi-objective optimization of four gates, *viz.*, NAND3, NAND4, NOR3, and NOR4, using the setup of Fig. 3. For each gate type, we ran two sets of experiments, one for finding the Pareto-optimals for the power-and-ϵ_{avg} balance, and the other for the PDP-and-ϵ_{avg}. The optimization yielded many combinations of transistor sizes as listed in Table 2.

In the following paragraphs, we have included the data and analysis only for NAND4 (power and ϵ_{avg}, and SNM) and NOR4 (PDP, ϵ_{avg}, and SNM), for the sake of brevity (see Tables 3 and 4).

The power-ϵ_{avg}-SNM plots for the Pareto-optimal NAND4s are shown in Fig. 4, while Table 3 lists the w's and

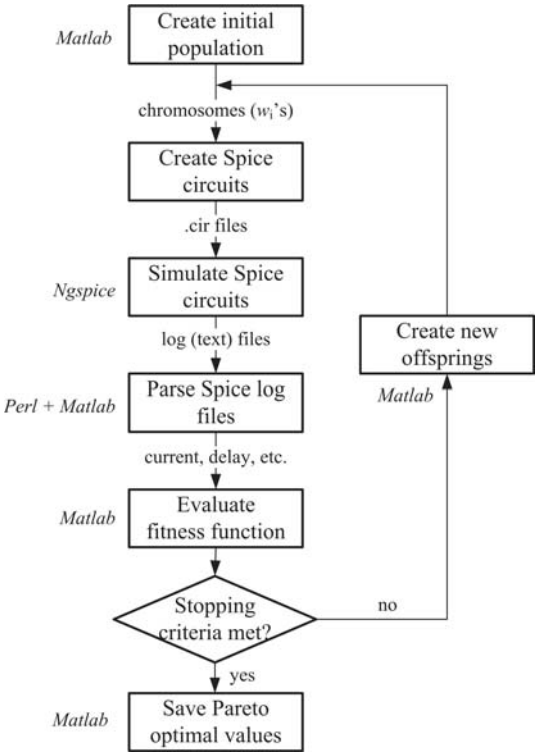


Fig. 3: Experimental setup for multi-objective GA-based transistor sizing.

Table 2: Number of optimal points in the Pareto-fronts.

Gate Type	Minimizing power and ϵ_{avg}	Minimizing PDP and ϵ_{avg}
NAND3	32	32
NAND4	42	27
NOR3	32	13
NOR4	42	42

the corresponding data. The progressively-sized gate (PSG) with the smallest area (row 2) dissipates 33% less power than the uniformly/conventionally-sized gate (USG) with an area saving of 39%. Much larger power savings are possible but they incur higher cost in terms of delay and ϵ_{avg}. The SNM is also marginally higher (1%) but it costs more in terms of ϵ_{avg}. For the PSGs, the power-consumption also shows an uptrend as the SNM increases.

The PDP-ϵ_{avg}-SNM data for the Pareto-optimal NAND4s (plots and data are omitted) reveals that the PSG gate with similar area as a USG consumes less PDP than the USG. PSGs' low gains in PDP are attributed to higher delays. Although we observed very low values of ϵ_{avg}, none of the PSGs were able to meet the SNM target (of 25%); this warrants another look at the experimental data.

The power-ϵ_{avg}-SNM data for the Pareto-optimal NOR4s demonstrates that the NOR4-PSGs have 40% power-advantage over the USG, while the SNM is 6% higher. (Due to the paucity of space, the plots and data are not included).These gains come with the added advantage of lesser gate area, i.e., 21%. The power and ϵ_{avg} show an inverse relationship (just like the PSG-NANDs did).

The PDP-ϵ_{avg}-SNM plots for the Pareto-optimal NOR4s are in Fig. 5, while Table 4 provides the w_i's and the related data. The smallest-area PSG has 28% PDP-saving and 47% smaller area than the USG; the SNMs are quite comparable, specifically PSG's SNM is 0.7% lower than USG. To gain the SNM-advantage, the gate area has to be increased, which slightly increases the PDP but it still is much lower than the USG.

6. Conclusions and Future Directions

Although progressive gate sizing scheme was originally intended to improve performance, the current work reveals many other advantages of the scheme. Using multi-objective GAs, we demonstrated that we were also able to improve power-and-reliability and energy-and-reliability; yet another benefit was the reduction in gate area. The scalability of our scheme and the encouraging results prompt us to continue our investigation with larger CMOS circuits. Looking into the effects of manufacturing-related variations, for example, in the channel length and the threshold voltage, is another venue for further research.

7. Acknowledgment

This work is partially supported by ADEC Award for Research Excellence (A^2RE) 2015.

References

[1] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolić, *Digital Integrated Circuits: A Design Perspective*, 2nd ed. Prentice Hall, NJ, USA, 2003.

[2] A. Beg, "Designing array-based CMOS logic gates by using a feedback control system," in *2014 IEEE Int. Conf. Syst. Man, Cybern.* San Diego, CA, USA: IEEE, 2014, pp. 935–939.

[3] A. Beg and A. Elchouemi, "Enhancing static noise margin while reducing power consumption," in *2013 IEEE 56th Int. Midwest Symp. Circuits Syst.* Columbus, OH, USA: IEEE, 2013, pp. 348–351.

[4] J. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: University of Michigan Press, 1975.

[5] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, 2nd ed. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2004.

[6] "International Technology Roadmap for Semiconductors (ITRS)," Austin, TX, USA, 2013. [Online]. Available: http://www.itrs.net/Links/2013ITRS/Home2013.htm

[7] S. A. Vitale, P. W. Wyatt, N. Checka, J. Kedzierski, and C. L. Keast, "FDSOI process technology for ultralow-power electronics," *Proc. IEEE*, vol. 98, no. 2, pp. 333–342, 2010.

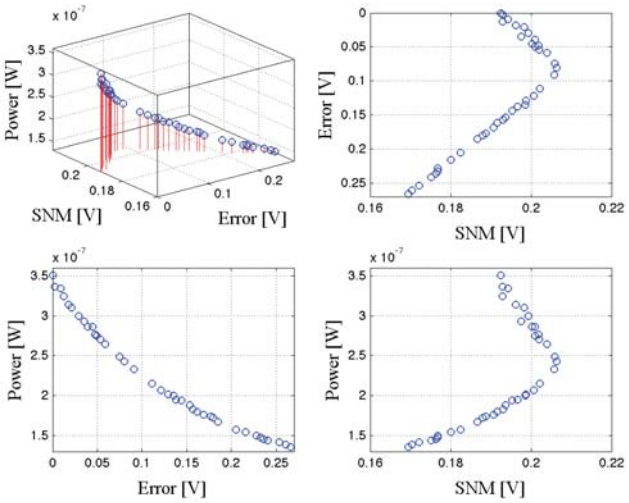


Fig. 4: Pareto-optimal NAND4s' power vs. SNM and error (ϵ_{avg}).

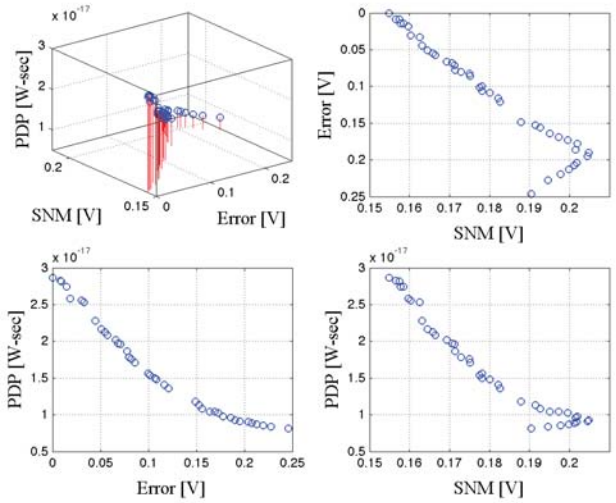


Fig. 5: Pareto-optimal NOR4s' PDP vs. SNM and error (ϵ_{avg}).

Table 3: Comparing uniformly-sized NAND4 gate with a few of 'power-ϵ_{avg}'-optimum NAND4 gates (w_i = width of M_i, as shown in Figs. 1 and 2).

Sizing Scheme	$w_0 - w_3$ [nm]	$w_4 - w_7$ [nm]	Gate Area [nm^2]	Power [W]	SNM [V]
Uniform	88	176, 176, 176, 176	23232	3.10E-07	0.198
Progressive	44	66, 90, 124, 190	14212	2.07E-07	0.201
Progressive	44	66, 103, 126, 209	14960	2.15E-07	0.202
Progressive	44	72, 103, 164, 237	16544	2.33E-07	0.206

Table 4: Comparing uniformly-sized NOR4 gates with a few of 'PDP-ϵ_{avg}'-optimum NOR4 gates (w_i = width of M_i, as shown in Figs. 1 and 2).

Sizing Scheme	$w_0 - w_3$ [nm]	$w_4 - w_7$ [nm]	Gate Area [nm^2]	PDP [W-sec]	SNM [V]
Uniform	352, 352, 352, 352	44	34848	5.50E-17	0.192
Progressive	182, 163, 163, 158	45	18612	8.11E-18	0.190
Progressive	213, 190, 189, 184	45	21032	8.39E-18	0.195
Progressive	223, 202, 202, 197	45	22088	8.51E-18	0.198

[8] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. M. Sylvester, and T. Mudge, "Near-threshold computing: reclaiming Moore's law through energy efficient integrated circuits," *Proc. IEEE*, vol. 98, no. 2, pp. 267–282, 2010.

[9] A. Tajalli, E. Vittoz, Y. Leblebici, and E. J. Brauer, "Ultra-low power subthreshold current-mode logic utilising PMOS load device," *Electron. Lett.*, vol. 43, no. 17, p. 911, 2007.

[10] A. Chandrakasan, D. Daly, D. Finchelstein, J. Kwong, Y. Ramadass, M. Sinangil, V. Sze, and N. Verma, "Technologies for Ultradynamic Voltage Scaling," *Proc. IEEE*, vol. 98, no. 2, pp. 191–214, 2010.

[11] H. Fuketa, M. Hashimoto, Y. Mitsuyama, and T. Onoye, "Transistor variability modeling and its validation with ring-oscillation frequencies for body-biased subthreshold circuits," *IEEE Trans. VLSI Syst.*, vol. 18, pp. 1118–1129, 2010.

[12] D. Bol, J. De Vos, R. Ambroisea, and J.-D. Legat, "Building ultra-low-power high-temperature digital circuits in standard high-performance SOI technology," *Solid State Electron.*, vol. 52, no. 12, pp. 1939–1945, 2008.

[13] A. Beg, "Automating the CMOS gate sizing for reduced power/energy," in *Int. Conf. Front. Inf. Technol. (FIT 2014)*, Islamabad, Pakistan, 2014, pp. 1–4.

[14] J.-m. Shyu, A. S. Jvanni-vincentelli, and A. E. Dunlop, "Optirnization-Based Transistor Sizing," *IEEE J. Solid-State Circuits*, vol. 23, no. 3, pp. 400–409, 1988.

[15] M. Kerttu, R. Drechsler, P. Lindgren, M. Kerttu, M. Thornton, and R. Drechsler, "Low power optimization technique for BDD mapped circuits," in *Asia South Pacific Des. Autom. Conf. 2001 (ASP-DAC 2001)*, Yokohama, Japan, 2001, pp. 615–621.

[16] M. L. M. Mui, K. Banerjee, and A. Mehrotra, "Power supply optimization in sub-130 nm leakage dominant technologies," in *Proc. - 5th Int. Symp. Qual. Electron. Des. ISQUED 2004*. IEEE Comput. Soc, 2004, pp. 409–414.

[17] B. C. Paul, A. Raychowdhury, and K. Roy, "Device optimization for digital subthreshold logic operation," *IEEE Trans. Electr. Dev.*, vol. 52, pp. 237–247, 2005.

[18] O. Coudert, "Gate Sizing for Constrained delay/power/area optimization," *IEEE Trans. VLSI Des.*, vol. 5, no. 4, pp. 465 – 472, 1997.

[19] P. K. Meduri and S. K. Dhali, "A Methodology for Automatic Transistor-Level Sizing of CMOS OpAmps," in *2011 24th Internatioal Conf. VLSI Des.*, Chennai, India, jan 2011, pp. 100–105.

[20] R. Rogenmoser, H. Kaeslin, and T. Blickle, "Stochastic methods for transistor size optimization of CMOS VLSI circuits," in *4th Int. Conf. Parallel Probl. Solving from Nat.*, London, UK, 1996, pp. 849–858.

[21] D. Ghaia, S. P. Mohanty, and G. Thakralc, "Fast analog design optimization using regression-based modeling and genetic algorithm: A nano-CMOS VCO case study," in *2013 14th Int. Symp. Qual. Electron. Des.*, Santa Clara, CA, USA, 2013, pp. 406–411.

[22] M. Blesken, U. Ruckert, D. Steenken, K. Witting, and M. Delintz, "Multiobjective optimization for transistor sizing of cmos logic standard cells using set-oriented numerical techniques," in *27th Norchip Conf.*, Trondheim, Norway, 2009, pp. 1–4.

[23] U. Blesken, M. and Lütkemeier, S. and Ruckert, "Multiobjective optimization for transistor sizing sub-threshold CMOS logic standard cells," in *2010 IEEE Int. Symp. Circuits Syst.*, Paris, France, 2010, pp. 1480–1483.

[24] H. K. O. Berge and S. Aunet, "Multi-objective optimization of minority-3 functions for ultra-low voltage supplies," in *2011 IEEE Int. Symp. Circuits Syst.*, Rio De Janeiro, Brazil, may 2011, pp. 2313–2316.

[25] A. Beg, "Automating the sizing of transistors in CMOS gates for low-power and high-noise margin operation," *Int. J. Circuit Theory Appl.*, pp. 1–14, 2014.

[26] "Predictive Technology Model," 2016. [Online]. Available: http://ptm.asu.edu/.

[27] "Berkeley Short-channel IGFET Model," 2013. [Online]. Available: http://www-device.eecs.berkeley.edu/bsim/?page=BSIM4_LR

[28] *Global Optimization Toolbox User's Guide*. Natick, MA, USA: The MathWorks, Inc, 2013.

SESSION

PREDICTION AND FORCASTING METHODS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Short Term Forcasting of Financial Market Using Adaptive Learning in Neural Network

Hong Li
Department of Computer Systems Technology
New York City College of Technology - The City University of New York
New York, New York, USA

Abstract - *This paper presents a short term forecasting of stock market using Feed Forward Multilayer Perceptron (FMP) with an adaptive learning algorithm. Business people often attempt to anticipate the market by interpreting external parameters, such as economic indicators, public opinion, and current political climate. However, the neural network is used in purpose of discovering trends in data that humans might not notice, and successfully use these trends in their predictions. Traditionally, the neural network training process takes trial and error for different values of learning factors. In work [1, 2], the analysis of convergence of learning process based on the Backpropagation algorithm leads to conditions that the learning factors satisfy to guarantee the convergence. In this paper, the conditions are further extended to a feasible formula that can be calculated to define an adaptive learning factor at iteration of learning process. The result of simulations using stock market data sourced from Yahoo! demonstrates that errors steadily decrease in training with the adaptive learning factor starting at different initial value and errors behave volatile with constant learning factors with different values. Once the neural network is trained, it provides predication for future market performance. The network is modified or retrained for every future data set starting with trained weights and learning factor. Performance of such network learning with adaptive learning factors are presented and demonstrated that the adaptive learning factor enhance the performance of training while avoiding oscillation phenomenon.*

Keywords: Neural network, financial market, stability analysis, adaptive learning

1. INTRODUCTION

The idea to predicate stock market is not new. It has been for many years the focus of many researchers to seek methods that achieve accurate forecasting using historical data [8, 9, 10]. Business people often attempt to anticipate the market by interpreting external parameters, such as economic indicators, public opinion, and current political climate. The question is, though, if neural networks can discover trends in data that humans might not notice, and successfully use these trends in their predictions. The Artificial Neural Networks (ANNs) have been proven in many real world applications to be useful in various tasks of modeling nonlinear systems, such as signal processing, pattern recognition, optimization, weather forecasting, to name a few. It has drawn many researchers in financial market because neural networks may be easy to use once the network is set up, but the setup and training of the network requires skill, experience, and patience. It's not all hype, though; neural networks have shown success at prediction of market trends. There are number of commercial software as well as free software that offer neural network simulation.

The ANN is a set of processing elements (neurons or perceptrons) with a specific topology of weighted interconnections between these elements and a learning law for updating the weights of interconnection between two neurons. The FMP networks have been shown to obtain successful results in system identification and control [3]. The Lyapunov function approach was used to provide stability analysis of Backpropagation training algorithm of such network in [4-7]. However, the training process can be very sensitive to initial condition such as number of neurons, number of layers, and value of weights, and learning factors which are often chosen by trial and error. This paper presents a detailed analysis of the FMP architecture and its stability. The Backpropgation algorithm is used for learning – that is, weight adjusting. This is a simple back-propagation network of three layers, and it is trained and tested on a high volume of historical market data. The challenge here is not in the network architecture itself, but instead in the choice of variables and the information used for training. In this research, historical data were chosen from previous six days open, close, high and low price as well six previous Nasdaq open, close, high and low.

The Least Square error function is defined and verified that it satisfies Lyapunov condition so that it guarantees the stability of the system. In the work [1], the analysis carries out a method that defines a range for value of learning factor at iteration which ensure the condition for stability are satisfied. In simulation, instead of selecting a learning factor by trial and error, author defines an adaptive learning factor which satisfies the convergence condition and adjust

ISBN: 1-60132-438-3, CSREA Press ©

connection weight accordingly. The simulation results are presented to demonstrate the performance.

2. Basic Principle of FMP Network

A system identification problem can be outlined as follow: a set of data is collected from the system including input data and corresponding output data observed, or measured as target output of the identification problem. The set is often called "training set". A neural network model with parameters, called weights, is designed to simulate the system. When the output from neural network is calculated, an error representing the difference between target output and calculated output from the system is generated. The learning process of neural network is to modify the network to minimize the error.

Consider a system with N inputs $X = \{X_1, \dots, X_N\}$ and M output units Y $=\{Y_1, \dots, Y_M\}$. A recurrent FMP network combines number of neurons, called nodes, feed forward to next layer of nodes, illustrated in Figure 1. Suppose N_l is number of nodes in lth layer, each output from the l-1th layer will be used as input for next layer. A system of a single layer with M outputs can be expressed in form of

$$Y_{jp} = f(Z) = f\left(\sum_{i=1}^{N} w_{ij} X_{ip} + \sum_{i=1}^{D} v_{ij} Y_{j(p-i)}\right) \quad (1)$$

where w_{ij} is called connection weight from input X_i to output Y_j;vij is called connection weight of local feedback at jth node with ith delay; $f(\,)$ is a nonlinear sigmoid function

$$f(Z) = \frac{1-e^{-\theta Z}}{1+e^{-\theta Z}} \quad (2)$$

with constant coeficient θ, called slope; p = 1, …, T, T is number of patterns, D is number of delay used in local feedback.

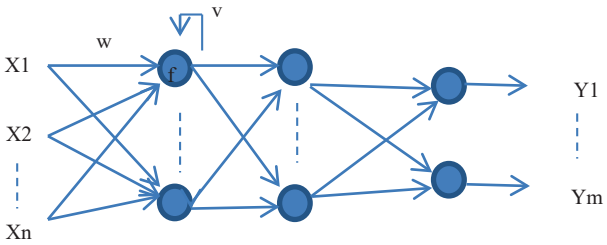


Figure 1. Feed-Forward Multi-Layer Network

The back-propagation algorithm has become the standard algorithm used for training feed-forward multilayer perceptron. It is a generalized the Least Mean Square algorithm that minimizes the mean squared error between the target output and the network output with respect to the weights. The algorithm looks for the minimum of the error function in weight space using the method of gradient descent. The combination of weights which minimizes the error function is considered to be a solution of the learning problem. A proof of the Backprpagation algorithm was presented in [10] based on a graphical approach in which the algorithm reduces to a graph labeling problem.

The total error E of the network over all training set is defined as

$$E = \frac{1}{T}\sum_{k=1}^{N_L}\sum_{p=1}^{T} e_k^2(p) \quad (3)$$

where $e_k^2(p)$ is the error associated with *pth* pattern at the *kth* node of output layer,

$$e_k^2(p) = (\, d_k(p) - Y_k^L(p))^2 \quad (4)$$

where $d_k(p)$ is the target at kth node and $Y_k^L(p)$ is the output of network at the kth node. The learning rule was chosen following gradient descent method to update the network connection weights iteratively,

$$\Delta W_j = -\mu \frac{\partial E}{\partial W_j}; j = 1, \dots, M \quad (5)$$

$$\Delta v_j = -\mu \frac{\partial E}{\partial v_j}; j = 1, \dots, D \quad (6)$$

where $W_j = (w_{1j}, \dots, w_{Nj})$ *and* $v_j = (v_{j1}, \dots, v_{Dj})$ are weight vectors in jth node; μ is a constant called learning factor.

3. Adaptive Learning Factor from Stability Analysis

There are no inclusive general concepts of stability for nonlinear systems. The behavior of a system may depend drastically on the inputs and the disturbances. However, Lyapunov theory has been used in many researches to examine the stability of nonlinear systems.

The definition of the Lyapunov function and the Lyapunov theorem are quoted below [11]:

Definition 1 (Lyapunov function): A scalar function V(x) is a Lyapunov function for the system

$$x(t+1) = f(\,x(t)), \; f(0) = 0 \quad (7)$$

if the following conditions hold:

1. $V(0) = 0$ and $V(0)$ is continuous in x
2. $V(x)$ is positive definite, that is, $V(x) \geq 0$ with $V(x) = 0$ only if $x = 0$

ISBN: 1-60132-438-3, CSREA Press ©

3. $\Delta V(x) = V(f(x(t)) - V(x(t))$ is negative definite, that is, $V(f(x(t)) - V(x(t)) \leq 0$ with $\Delta V(x) = 0$ only if $x = 0$;

Theorem 1 (Lyapunov Theorem): The solution $x(t) = 0$ for the system given by (7) is asymptotically stable if there exists a Lyapunov function of the system in x.

The stability of the learning process in an identification approach leads to better modeling and a convergent process. According to the Lyapunov theorem, determination of stability depends on selection and verification of a positive definite function. For the systems defined in (1) – (2), assume that the Backpropagation learning rule is applied and the error function and weights updating rule are defined in (5) - (6), then define

$$V(t) = \frac{1}{N_L T} \sum_{j=1}^{N_L} \sum_{p=1}^{T} e_k^2(t) \quad (8)$$

The proof is given in the following theorem that the V (t) satisfies the Lyapunov conditions.

Theorem 2: Assume that the nonlinear sigmoid function $f(\,)$ defined in equation (2) is continuous and differentiable, the network is defined in (1)-(2) with learning rule (5) - (6), then the system is stable under the condition:

$$\mu < \frac{TM}{\sum_{j=1}^{M} \sum_{p=1}^{T} \left(\left\| \frac{\partial Y_p}{\partial W_j} \right\|^2 + \left\| \frac{\partial Y_p}{\partial v_j} \right\|^2 \right)} \quad (9)$$

Proof: It is give in [1].

Theorem 3. Assume that the system with one hidden layer can be represented in the form of:

$$Y_{jp} = f(Z_{jp}^1) = f(\sum_{i=1}^{N} w_{ij}^o X_{ip}^1 + \sum_{i=1}^{D} v_i^o Y_{j(p-i)}) \quad (10)$$

$$X_{jp}^1 = f(Z_{ip}) = f(\sum_{i=1}^{N} w_{ij}^h X_{ip} + \sum_{i=1}^{D} v_i^h X_{j(p-i)}^1) \quad (11)$$

the gradient descent rule

$$\Delta W_o^j = -\mu \frac{\partial E}{\partial W_o^j}, \ j = 1, \dots, M \quad (12)$$

$$\Delta v_o^j = -\mu \frac{\partial E}{\partial v_o^j}, \ j = 1, \dots, D \quad (13)$$

$$\Delta W_h^j = -\mu \frac{\partial E}{\partial W_h^j}, \ j = 1, \dots, M \quad (14)$$

$$\Delta v_h^j = -\mu \frac{\partial E}{\partial v_h^j}, \ j = 1, \dots, D \quad (15)$$

where

$$W_o^j = (w_{o1}^j, \dots, w_{oN}^j)^T, \ v_o^j = (v_{o1}^j, \dots, v_{oD}^j)^T$$

$$W_h^j = (w_{h1}^j, \dots, w_{hH}^j)^T, \ v_h^j = (v_{h1}^j, \dots, v_{hD}^j)^T$$

are weight vectors in jth node in output layer and hidden layer respectively. H is the number of nodes in hidden layer. The system is stable when the learning factor in (18) - (21) satisfies the condition given below:

$$\mu < \frac{TM}{2\sum_{j=1}^{M} \sum_{p=1}^{T} \left[\left\| \frac{\partial Y_{pj}}{\partial W_o^j} \right\|^2 + \left\| \frac{\partial Y_{pj}}{\partial v_o^j} \right\|^2 + \left\| \frac{\partial Y_{pj}}{\partial W_h^j} \right\|^2 + \left\| \frac{\partial Y_{pj}}{\partial v_h^j} \right\|^2 \right]} \quad (16)$$

Proof: Similarly, given in [1].

In general simulation, the learning factor was predefined constant whose value was selected by trial and error. The simulation performance differs from different values of learning factor. The learning process may converge or may not reach a satisfactory threshold with different learning factors. From the result of above theorem, the convergence is guaranteed if an adaptive learning factor is selected at iteration of the learning which satisfies the stability condition. For purpose of simplifying the simulation, instead of calculating all $\frac{\partial Y_{pj}}{\partial W^l}$ and $\frac{\partial Y_{pj}}{\partial v^l}$ for l = 1, … L; j = 1, … N_L, the following corollary will provide a more restrictive but easier calculated condition.

Consider infinite norm notation for any vector $X = \{x_1, x_2, \dots, x_n\}$ that $\|X\|_\infty = \max_{1<i<n} \{x_i\}$, for simplicity, use notation $\|*\|$ in this paper representing $\|X\|_\infty$). Applying infinite norm in (16) and notation $|v_j^l| = \sum_{d=1}^{Dv} v_{jd}^l$, calculation of $\left\| \frac{\partial Y}{\partial W^o} \right\| \ and \ \left\| \frac{\partial Y}{\partial v^o} \right\|$ lead to

$$\left\| \frac{\partial Y}{\partial W^o} \right\| \leq \frac{\theta}{2} \left[1 + |v_j^o| \left\| \frac{\partial Y}{\partial v^o} \right\| \right]$$

$$\left\| \frac{\partial Y}{\partial v^o} \right\| \leq \frac{\theta}{2} \left[1 + |v_j^o| \left\| \frac{\partial Y}{\partial v^o} \right\| \right]$$

then

$$\left\| \frac{\partial Y}{\partial v^o} \right\| \leq \frac{\theta}{(2 - \theta |v_j^o|)}, \text{ and } \left\| \frac{\partial Y}{\partial v^o} \right\| \leq \frac{\theta}{(2 - \theta |v_j^o|)} \quad (17)$$

Further calculation of $\left\| \frac{\partial Y}{\partial W^h} \right\| \ and \ \left\| \frac{\partial Y}{\partial h} \right\|$ lead to

$$\left\| \frac{\partial Y}{\partial W^h} \right\| \leq \frac{\theta}{2} (\sum_{j=1}^{N} w^o \left\| \frac{\partial Y^h}{\partial W^h} \right\| + |v_j^o| \left\| \frac{\partial Y}{\partial v^h} \right\|)$$

$$\leq \frac{\theta}{2} (|w^o| \frac{\theta}{(2 - \theta |v_j^o|)} + |v_j^o| \left\| \frac{\partial Y}{\partial v^h} \right\|)$$

$$\left\| \frac{\partial Y}{\partial W^h} \right\| \leq \frac{\theta^2 |w^o|}{(2 - \theta |v_j^o|)^2}, \text{ and } \left\| \frac{\partial Y}{\partial v^h} \right\| \leq \frac{\theta^2 |w^o|}{(2 - \theta |v_j^o|)^2} \quad (18)$$

ISBN: 1-60132-438-3, CSREA Press ©

Add the (26) and (27), and result in the following Corollary.

Corollary 1: The system defined in (16) – (17) converges if the learning factor in (18) – (21) satisfies the following conditions:

$$2 - \theta\,|v^o| > 0\,, \tag{19}$$

$$.\mu < \frac{\left(2-\theta\left|v_j^o\right|\right)^2}{4\theta(2+\theta(|w^o|-\left|v_j^o\right|} \tag{20}$$

4. SIMULATION

Many researches provide analysis that what historical data are adequate for training of the neural network system as the market can be influenced by many business or political factors. However, the research of neural network learning is to build a system that learned from only previous data and required less human inputs. Human never fully understand the complicated correlation of factors that might have influenced the market a high accuracy. This simulation mainly demonstrates that the enhanced learning algorithm may avoid many trial and error for selection of learning factors.

In traditional neural network training, the initial weights are randomly selected, a learning factor is predefined. The performance of the learning can sometime very volatile due to the selection of the learning factor. To find the optimal fit, the trial and error is common practice that runs the simulation with different values of learning factors. In this research, an upper boundary of learning factors (20) is derived from the theory of convergence. At iteration of network training, the norm of weights is calculated following (17), and a learning factor is defined to satisfy the convergence condition (20).

A three layer neural network structure was selected with 24 inputs, 10 and 7 nodes in the hidden layer, and two outputs. The data are sourced from Yahoo! Finance. The daily price of stock IBM including open, high low and close price in 2015 is selected for demonstration. Two outputs are daily high and low. 24 inputs are selected as follows: high and low of the IBM stock from previous six days; high and low of Nasdaq from previous six days. Input and output data are normalized to range from 0 to 1.

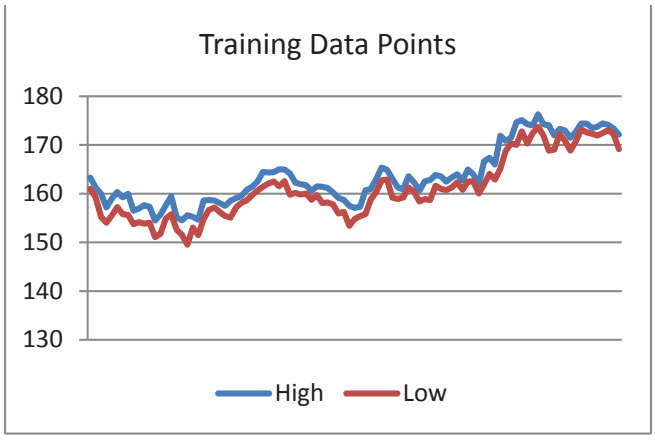


Figure 2. IBM Stock Daily High and Low Price as Training Points

With the constant learning factor, several values were used for the learning trials: 0.2, 0.15, 0.1, and 0.05. After number of attempts, with slope set as 0.7, learning factor set as constant 0.05, momentum term set as 0.1, and random generated initial weights, the system reached to absolute error 0.028 after 100000 iterations. The error behaviors are shown respectively in Figure 3 – Figure 6.

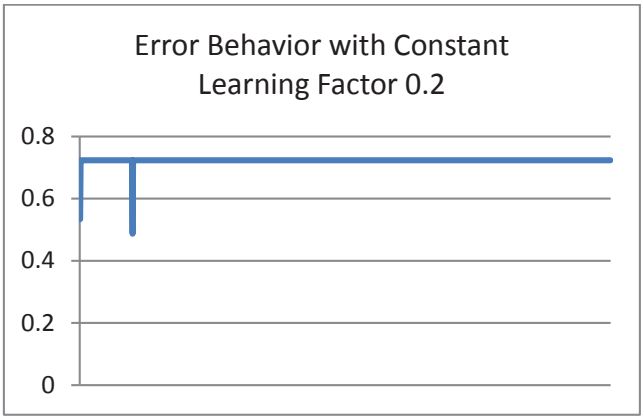


Figure 3. Error Behavior of Neural Network Training with Constant Learning Factor 0.2

ISBN: 1-60132-438-3, CSREA Press ©

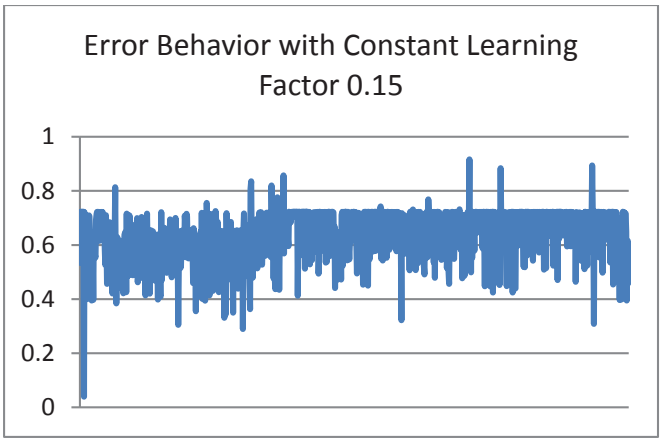


Figure 4. Error Behavior of Neural Network Training with Constant Learning Factor 0.15

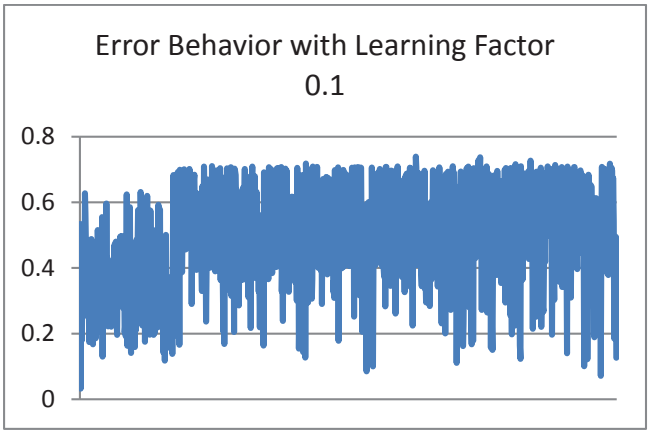


Figure 5. Error Behavior of Neural Network Training with Constant Learning Factor 0.1

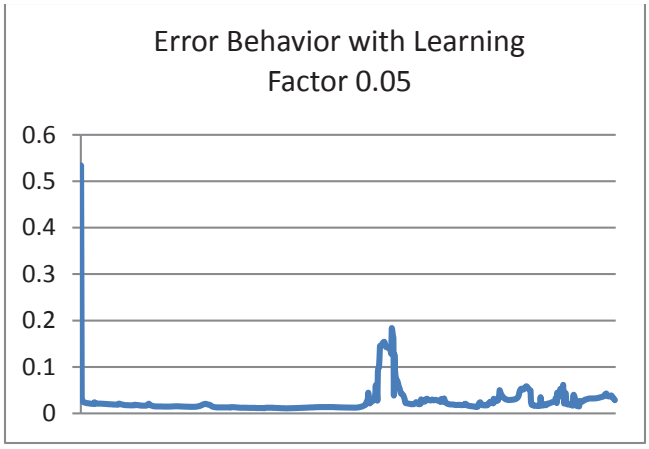


Figure 6. Error Behavior of Neural Network Training with Constant Learning Factor 0.05

With initial learning factor 0.2, 0.15, 0.1 and 0.05, the errors steadily decreases at iteration when an adaptive learning factor was applied at iteration, Figure 7 – Figure 10 demonstrate the error behaviors of learning with initial learning factors, 0.2, 0.15, 0.1 and 0.05 respectively. It is observed that error behaviors do not differ with different initial values of learning factors. However, the constant learning factor could cause volatile performance of training.

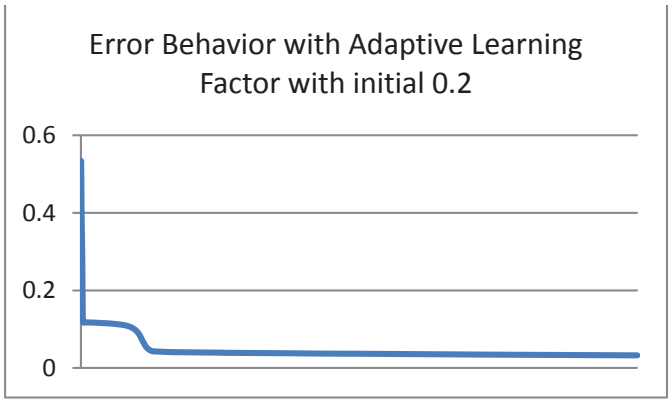


Figure 7. Error Behavior of Neural Network Training with adaptive Learning Factor with initial 0.2

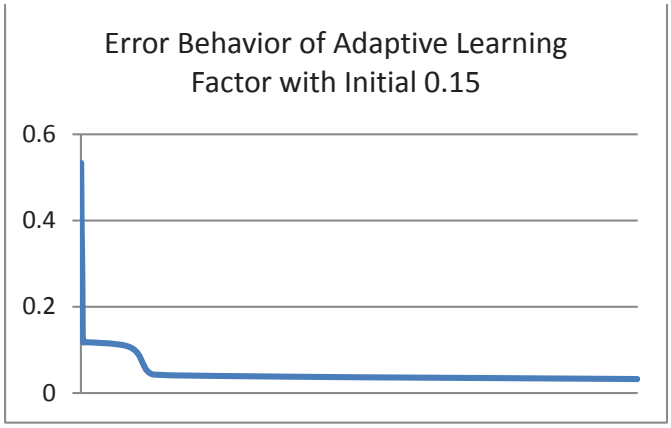


Figure 8. Error Behavior of Neural Network Training with adaptive Learning Factor with initial 0.15

The Figure 11 presents the comparison of IBM daily high and ANN model of 100 days and 9 days prediction of the trained neural network. Figure 12 presents comparison of IBM daily low and ANN model of 100 days and 9 days prediction of low from the trained neural network.

ISBN: 1-60132-438-3, CSREA Press ©

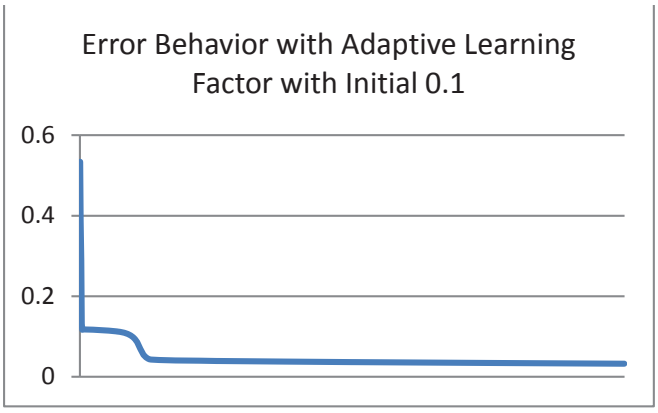


Figure 9. Error Behavior of Neural Network Training with Adaptive Learning Factor with initial 0.1

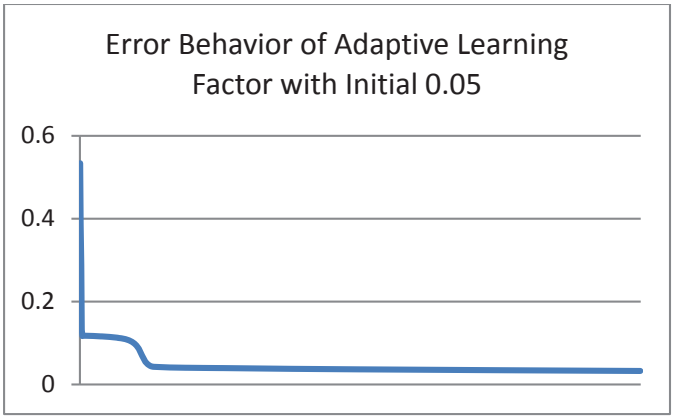


Figure 10. Error Behavior of Neural Network Training with adaptive Learning Factor with initial 0.05

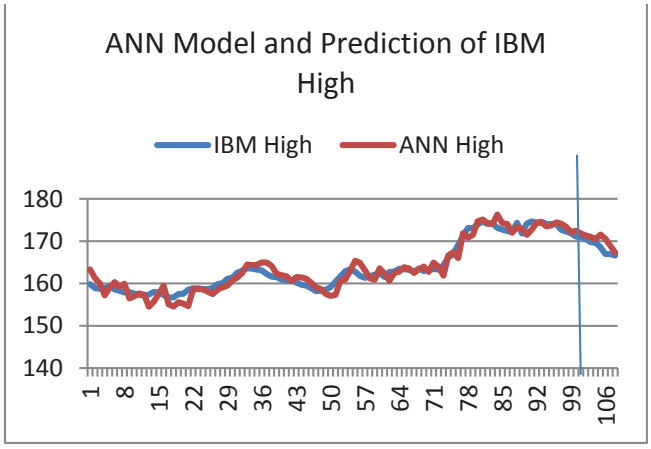


Figure 11. ANN Model and predication of IBM high

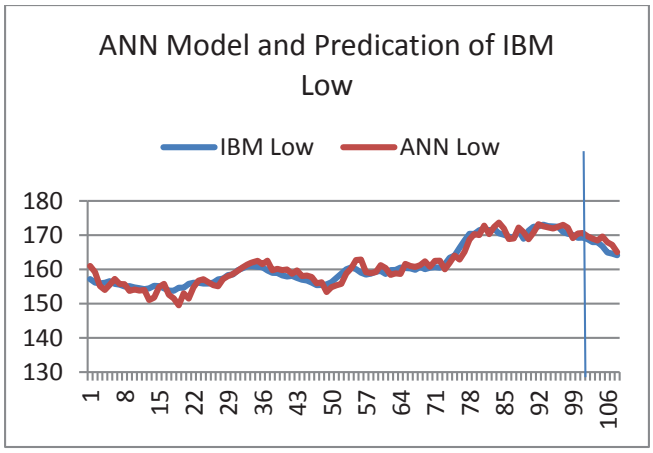


Figure 12. ANN Model and predication of IBM low

5. SUMMARY

This research focused on improvement of learning algorithm. A condition was derived from the proof of convergence of neural network system learning process using Backpropagation algorithm. The condition provides an upper boundary of the learning factor. Instead of select a constant learning factor by trial and error, an adaptive learning factor is calculated at iteration satisfying the convergence condition. Furthermore, a more simplified condition was used to provide a feasible implementation of the adaptive learning factor. At iteration of the learning process, an adaptive learning factor was selected satisfying the stability condition to avoid unstable phenomena. Simulation results of the IBM stock prices demonstrated that a learning factor arbitrarily chosen out of the predefined stability domain leads to an unstable identification of the considered system; however, an adaptive learning factor satisfying the conditions chosen for this study ensures the stability of the identification system.

6. REFERENCES

[1] Hong Li, Adaptive Learning Factor of Backpropagation in Feed Forward Neural Networks, International Journal of Modern Engineering (IJME), Vol. 14, No 2, Spring/Summer 2014, pp 47 – 53.

[2] Hong Li and Ali Setoodehnia, Convergence Analysis of Adaptive Recurrent Neural Network, International Journal of Engineering Research and Applications (IJERA), Vol. 4, Issue 6(Version 1), June 2014, pp.48-53

[3] Yu, W., Poznyak, A. S., & Li, X. (2001). Multilayer Dynamic Neural Networks for Non-linear

ISBN: 1-60132-438-3, CSREA Press ©

System Identification. International Journal of Control, 74(18), 1858-1864.

[4] Korkobi, T., Djemel, M. & Ctourou, M. (2008). Stability Analysis of Neural Netorks-based System Identification. Modeling and Simulation in Engineering, Volume 2008, Article ID 343940

[5] Zhu, E. (2010). Stability Analysis of Recurrent Neural Networks with Random Delay and Markovian Switching. Journal of Inequalities and Applications

[6] Akea, H., Alassar,R., & Covachev, V. (2005). Stability of Neural Networks with Time Varying Delays in the Presence Impulses. Advances in Dynamical Systems and Application,. 1(1), 1-15.

[7] Kolla, Sri R. (2012). Comparison of Time Delay Controllers for A class of Networked Control System, International Journal of Modern Engineering, Fall/Winter, 13(1), 51-56.

[8] Tarsauliya, A. Analysis of Artificial Neural Network for Financial Time Series Forecasting, International Journal of Computer Applications (0975 – 8887) Volume 9– No.5, November 2010.

[9] Andrew Skabar, A. Cloete, I Neural Networks, Financial Trading and the Efficient Markets Hypothesis the Twenty-Fifth Australasian Computer Science Conference (ACSC2002), Melbourne, Australia.

[10] Choon Ki Ahn, H. (2012) Stability Conditions for Fuzzy Neural Networks, Advances in Fuzzy Systems Volumn Articale ID 281821

[11] Zhao, H. (2003). Global Stability of Neural Network with Distributed Delays. Joural of Neural, Parallel & Scientific Computations, 11(3), 237-252.

[12] Liang, G. (2010). Global Asymptotically Stability of Cellular Neural Networks with Time-varying Delay, 2010 8th World Congress on Intelligent Control and Automation , (pp 5031-5036)

ISBN: 1-60132-438-3, CSREA Press ©

A Neural Network Approach for Predicting Microstructure Development in Cement

Dario Cruz[1], **Douglas A. Talbert**[2], **William Eberle**[2], **Joe Biernacki**[1]
[1]Chemical Engineering, Tennessee Tech University, Cookeville, TN, USA
[2]Computer Science, Tennessee Tech University, Cookeville, TN, USA

Abstract - *Although portland cement concrete is the most widely used construction material in the world, efficient simulations for predicting hydration and property development (hardening and aging) are yet to be developed and made widely available to researchers and engineers. Current state-of-the-art simulations rely on computationally expensive models to generate spatially resolved time sequences of chemical and physical properties (microstructures) that allow engineers to predict permeability, strength, and durability. Improving simulation accuracy or efficiency would be a significant contribution. We describe a novel application of neural networks to simulate the microstructure of hydrating tricalcium silicate (the most abundant component of ordinary portland cement). Initial model predictions correlate well with benchmark models but require only a small fraction of the computation time.*

Keywords: artificial neural networks, microstructure simulation, engineering application

1 Introduction

The properties of concrete, the most used material in the world by mass other than water, are largely due to portland cement, the binder that provides strength and durability. As such, proper understanding of cement hydration and property development (hardening and aging) in concrete is vital to infrastructure development and maintenance. Unfortunately, cement hydration is among science's enigmatic challenges, and ways to efficiently predict hydration and property development are widely sought [1].

Portland cement concrete is a complex composite consisting of aggregates (stones which provide volume, mass and stability to the concrete), cement paste (hydrated cement), unhydrated cement, and water-filled or partially water-filled pores. Furthermore, the aggregate, cement paste, and cement particles are themselves complex composites. Thus, the properties of concrete are largely dependent upon the proportions of these constituents and the chemistry and microscopic physical structure of the hydrated portland cement paste called *microstructure* (what the material physically looks like).

Predicting the progress of *hydration* and *microstructure* development is an ongoing challenge. Current modeling strategies require researchers to opt for either a *computationally intensive approach* that provides detailed microstructure, in some cases requiring weeks of run time for massively parallelized code on supercomputing platforms [2], or a *computationally efficient approach* that sacrifices detail for computational speed [3]. The goal of *this work* was to develop and evaluate a novel *application* of machine learning to streamline computations associated with microstructure development and, thus, reduce the cost of predicting cement microstructure.

After presenting some background and related work, we describe our proposed approach followed by our experimental methodology, results, discussion, and conclusions.

2 Background and related work

Type I portland cement consists mainly of *alite*, a form of tricalcium silicate (C_3S). Hydration of alite has historically been described as a process characterized by five distinct stages defined by which chemical reactions control the physical and chemical changes (*kinetics*) in that stage [4]. Hydration starts with the formation of hydrate nuclei at various sites on the alite surface, a random or stochastic process. These nuclei grow over time, fill the pore space, cause the cement to harden and properties to change, e.g. strength to develop. Model-based prediction of these kinetic behaviors is thought to be a key step in understanding how to design and optimize cements and other constituents of portland cement concrete [5].

2.1 Existing models of concrete hydration

The details of the complex process of alite hydration are vigorously debated but have been implemented in a number of existing models, including HydratiCA [6], Mi-CBM [3], CEMHYD3D [7], and μic (pronounced *mīk*) [8]. We briefly present each model below.

Developed at the National Institute of Standards and Technology (NIST), HydratiCA uses a kinetic cellular automaton-based (or kinetic CA) approach [7][9] to capture the stochastic nature of microstructure development. The CA approach models the problem as a three-dimensional (3-D) array of particles with associated properties in a space defined by a collection of *voxels* (*i.e.*, a pixel with an assigned volume and location in 3-D space). Changes to the particles and their properties over time is determined by the repeated application of rules describing the interactions among neighboring

particles. HydratiCA was developed and optimized for the simulation of mineral hydration processes and is based on an asynchronous 3-D stochastic kinetic CA. CA models such as HydratiCA can be viewed as image-based models that produce detailed microstructures that can be viewed as simulated 3-D color pictures similar to those shown in Fig. 1.

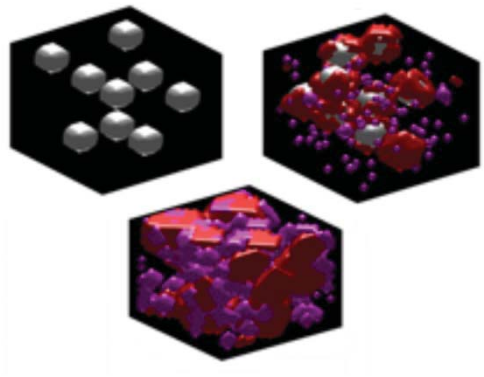


Fig. 1. Simulated Microstructure Images.

A predecessor of HydratiCA, CEMHYD3D simulates 3-D cement microstructures using carefully calibrated probabilistic rules [7]. In spite of some weaknesses, CEMHYD3D successfully generates microstructures that appear to reproduce a number of physical properties [10]-[12].

Developed by Bishnoi and Scrivener [8], µic is a user-definable modeling platform for representing microstructure and dynamics of microstructural change (kinetics). The authors define it as a tool-based "vector approach," which stores information concerning the position, orientation and size of individual microstructural features as a vector. As is the case with the other microstructure-generating models, µic suffers from a high computational expense.

Mi-CBM (Multi-ionic Continuum-Based Model) is a continuum-based model that averages properties in predefined ways. This results in properties such as *average* density at a particular radius rather than density as a function of radius, azimuth and polar angle. While less computationally expensive than HydratiCA, Mi-CBM does not provide detailed microstructural information.

2.2 Machine learning in concrete and ceramics

In the field of cement hydrations, machine learning has only been used in select cases and ways. For instance, Devaney and Hagedorn [13], applied several machine learning techniques to discover new rules for classifying hydrating plaster over multiple time periods. They trained on X-ray micro-tomography data by using unsupervised classification, decision trees, and genetic programming to classify plaster powder and applied the learned rules to data at different hydration times. This work was also extrapolated to separate and distinguish un-hydrated plaster from gypsum crystals in the microstructure.

Wang and Yang [14], modeled early-age hydration kinetics of portland cement using flexible neural trees [15] coupled with genetic algorithms to evolve the tree's structure, rules, and parameters. The results show that the simulations and rate of hydration agreed well with experimental data.

However, neither of these approaches address the issue of microstructure development. *To the best of our knowledge, nobody has attempted to apply a machine learning approach for evolving cement microstructure*. One of the issues is that traditional machine learning classifiers are deterministic, and what we are dealing with in this domain are physical processes with some randomness. Some research, however, has dealt with predicting behavior in chaotic physical systems, such as weather forecasting [16].

3 Proposed autoregressive model (ARM)

Our proposed approach aims to improve the efficiency of microstructure generation by combining a CA with an artificial neural network. Specifically, we developed a set of non-linear autoregressive artificial neural networks [17], or *autoregressive model* (ARM), to predict the next state of alite (C_3S) particles given a set of prior states for the particles and their neighbors. Once the neural networks are trained, we initialize our model using a CA-based tool to generate an initial set of particle states, then feed those initial states into the neural networks to predict the next set of states. These predicted states are added to the CA-generated states and fed back into the neural networks to predict the *next* set of states. This cycle continues until states have been predicted for the desired number of time steps.

3.1 Cement hydration and CA state generation

For the CA phase of our approach, we developed an application we call *HydratiCA-lite*, based on stochastic rules for chemical reactions. HydratiCA-lite simulates the conversion of cement grains (C_3S) to cement hydrates (C-S-H(I), C-S-H(II) and CH). Nucleation sites (the location where the first hydrates form) are picked at random from among an available set of voxels and precipitating phases are allowed to grow (deposit). Currently, the morphology used for C_3S particles is spherical, but others can also be included.

To reduce the amount of time needed to simulate hydration, HydratiCA-lite was implemented as a parallel algorithm by dividing the full lattice of particles into smaller sub-lattices, each one belonging to a single *process*. This parallel HydratiCA-lite model effectively produces microstructure, local solution, and solid phase compositions. Global properties can also be computed by forming spatial averages over the system volume.

The chemical reactions shown in Fig. 2 for C_3S hydration along with their respective kinetic rate constants (k_j^i) were adopted from Bullard and Flatt [18].These chemical reactions determine the nature of the microstructure formed; each chemical species contributes to the morphology and properties of it. The rate of formation and consumption of species is governed by the kinetic constants and associated rate laws. These play an important role in the kinetics of C_3S hydration. For more information, see reference [6].

3.2 Autoregressive model

The HydratiCA-lite output was used as input to the ARM and to generate the microstructure data to both train and validate the neural network. This HydratiCA-lite data represents the hydration and property development of cement as a function of time. Thus, this data is a series of microstructure snapshots at different times. The period of time between snapshots is defined as the *lag* [19].

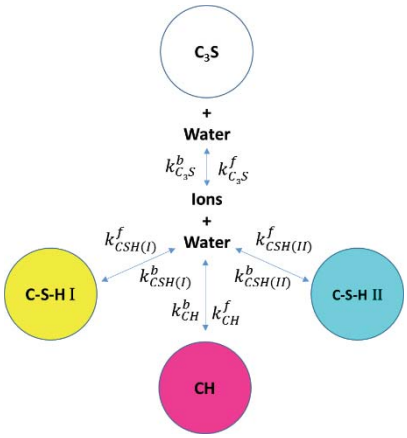


Fig. 2. Reaction Schema, Colors Map to Table I and Rate Constants to Tables II and III.

The neural network learns a function F that satisfies the autoregressive model defined by:

$$r(k) = F[r(k\text{-}1), \ldots, r(k\text{-}n_r)] + e(k) \quad (1)$$

where the value of r at lag k depends on past values of r and the noise sequence $e(k)$ and where F is a function that produces the minimum error given a chosen objective (cost function).

Intuitively, $r(k)$ is the microstructure snapshot (at *lag* k) being predicted, and $r(k - 1), \ldots, r(k - n_r)$ are prior snapshots used to inform the current prediction. The use of multiple snapshots rather than only the one immediately prior to $r(k)$ enables the model to capture the rate of change (kinetics) of the reaction. The more general form of (1) includes an additional set of external information. In our case, however, such external information was not used.

To model (1), we use a Layered Digital Dynamic Network (LDDN) [20]. LDDNs support multiple recurrent (feedback) connections. Networks with such connections have been shown to be effective for time-series data modeling [17]. The structure of LDDNs also enable a type of memory that makes it possible to learn time-varying patterns (*i.e.*, their output at any given lag will depend not only on the current network input, but on the *history* or the *past* sequence of inputs).

It is precisely this feature of LDDNs that make them of interest for our task. The recurrent (autoregressive) network receives an input signal which will propagate forward through the network's layers, while *delayed* versions of the output (the network's results) and input signals, go into the input's layer through an arrange of delays that modifies the entering signals (tapped delay lines) in order to successfully simulate time series behavior. Unlike *static* networks, the computed output of the network is a function of the current inputs, the weights and the biases of the network's layers, and the computed outputs from previous lags. More complete information regarding LDDNs can be found in [20].

3.3 Network training

The training *examples* were chosen to be three-dimensional (*3-D*) *Moore neighborhoods* of voxels. Thus, the number of voxels (3-D pixels) within a neighborhood is 27. Each of the neighborhoods contain unique voxels, thus the neighborhoods do not overlap with each other.

The software used for network simulation and training was MATLAB 2014a [21]. Training of the network used an internal mechanism to avoid over-fitting and to optimize the network outputs; this mechanism takes the data and distributes it randomly into three internal sub-categories: training, validation, and testing. The training data is used to learn network weights. The validation set is used to check for the stopping condition, and the testing set is used to evaluate different model parameters (*e.g.*, different weight arrangements).

The algorithm chosen for training uses a *Levenberg-Marquardt back-propagation* method. The cost function used during training is the *mean squared error (MSE)* of the network's outputs compared to the original target data [20].
The network is trained in two phases. First, we apply *open-loop* training that does not use the delayed feedback loops that enable "memory" in the LDDN network. Once that completes, the network starts with the final weights from the open-loop training phase and initiates the *closed-loop* training phase. This second phase makes use of the delayed feedback of the outputs in cost computations.

This dual approach seeks to enhance the performance of the neural network's recurrency by re-adjusting the weights according to feedback from previous outputs. This promises to reduce the effect of error propagation. The neural network is a standard two-layer feed-forward network having an input layer, one hidden layer and an output layer, with a sigmoidal transfer function in the hidden layer, and a linear transfer function in the output layer.

3.4 Data processing: color channel separation

Like HydratiCA, HydratiCA-lite's output can be viewed as simulated 3-D color images. To produce these simulated images, a specific color was chosen for each solid phase (*e.g.*, magenta=*CH*, yellow=*C-S-H(I)*, white=C_3S, cyan=*C-S-H(II)*). These choices were made so that each color channel (red, green, and blue) would convey different information about the chemical components in the concrete. Table I shows the mapping between chemical species (molecular entity) and color channels.

Table I. Species Assigned to Each Channel.

RED	GREEN	BLUE
C_3S	C_3S	C_3S
C-S-H (β)	C-S-H (β)	CH
CH	C-S-H (γ)	C-S-H (γ)

To fully leverage this information, we separate the output from HydratiCA-lite into three separate color channels. Thus, three different sets of inputs to the network were considered, each representing one color channel. Each channel varies continuously between 0 and 1, and each was trained and predicted separately [22], resulting in three different neural networks.

3.5 Data preprocessing: static voxel removal

Initial experiments showed that the presence of *static voxels* (voxels whose value does not change) during training significantly degraded performance of the model. To reduce the impact of such voxels, a threshold value was chosen for the number of dynamic voxels a neighborhood must have to be fed to the network. We arbitrarily chose a threshold value of nine. Thus, if a neighborhood contained fewer than 9 dynamic voxels, the neighborhood was discarded from the data. Unfortunately, this practice also discards some dynamic voxels (those in neighborhoods with fewer than 9 dynamic voxels) and results in incomplete predictions. Therefore these voxels are incorporated into a set named *excluded dynamic voxels* and are predicted separately.

4 Experimental methodology

4.1 Baseline simulations and data generation

For our experiments, we first ran HydratiCA-lite to generate data for 1200 simulated 3-D images of cement hydration under a predetermined set of reaction rates (kinetic constants). Each image represented successive 100-second intervals.

This data set served as the training data for our neural network. We then ran a second HydratiCA-lite simulation using the same set of kinetic constants. Due to the stochastic nature of the hydration process, the resulting data differed qualitatively and quantitatively (as expected and desired) from our training data. This second simulation served as test data for our experiments. These two data sets ($train_1$ and $test_1$) were used to test the time and accuracy of our proposed approach when the training kinetics match the test kinetics but nucleation sites differ.

We also wanted to evaluate the impact of varying the test kinetics. To do so, we ran a third simulation with HydratiCA-lite with a different set of kinetic constants and different nucleation sites. The resulting data ($test_2$) would be used to evaluate the ability of the trained model to adapt to different reaction rates.

Even though there is only one training set, the subscript on $train_1$ serves as a reminder that its kinetic constants match those of $test_1$. Each simulation produced 1200 3-D images. The kinetic constants used are shown in Tables II and III. All other simulation parameters were unchanged across the three simulations.

Table II. Kinetic Constants for $Train_1$ and $Test_1$

Reaction	k^f (mol/m^{2}s)	k^b (mol/m^{2}s)
C3S	5.20E-07	5.20E+10
C-S-H(I)	1.20E-06	3.97E+01
C-S-H(II)	1.00E-06	9.12E+06
CH	7.19E-06	1.14E+00
Ions	6.00E-02	9.96E-01

Table III. Kinetic Constants for $Test_2$

Reaction	k^f (mol/m^{2}s)	k^b (mol/m^{2}s)
C3S	5.20E-08	1.65E-06
C-S-H(I)	9.50E-11	6.60E+01
C-S-H(II)	8.00E-10	6.65E-03
CH	1.62E-07	3.79E-02
Ions	6.00E-02	9.96E-01

It is important to note the difference between the kinetic constants in Table II and Table III, and that even a small change in one of them can lead to a totally different behavior of the concentration of chemical species that conform the microstructure.

Table IV describes the results of applying the *color channel separation* and the *static voxel removal* (as described in Section 3.4) to the $train_1$ data set.

Table IV. Neighborhoods and Voxels per Color Channel

	RED	GREEN	BLUE
# of dynamic neighborhoods	203	489	614
Total # of static voxels	10508	8061	6578
Total # of dynamic voxels	1827	4401	5526

4.2 Experimental Simulations

For our experiments, we partitioned the data into sets of 20 simulated snapshots with 100 seconds between each snapshot. We call each 20 snapshot partition a *lag*. The training data consisted of 60 such lags. Since, in practice, our network will start with an initial set of lags generated by HydratiCA-lite, we trained each of the ARM's three (red, green, and blue) networks using only lags 30-60 via the two-phase training described in Section 3.3.

Evaluation was performed by seeding each network with the first 10 lags from the test data ($test_1$, then $test_2$). Then the

neural networks are used to predict the next lag. The data input window then slides one lag forward in time and the next lag is predicted. This repeats until all desired lags have been predicted.

In the following section, we evaluate:

- the correlation between HyradiCA-lite's simulated microstructure and the ARM's predicted microstructure,
- the comparison between HyradiCA-lite's simulated solution composition and the ARM's solution composition, and
- execution time

5 Results

Fig. 3 visually illustrates the similarities between the HydratiCA-lite-generated images and the neural network-generated images. The figure illustrates a series of 2-D cross-sections at various hydration times comparing the HydratiCA-lite outcomes for *test*$_1$ (Fig. 3a) to that for the trained model (Fig. 3b). These *snapshots* indicate good qualitative predictability of the ARM.

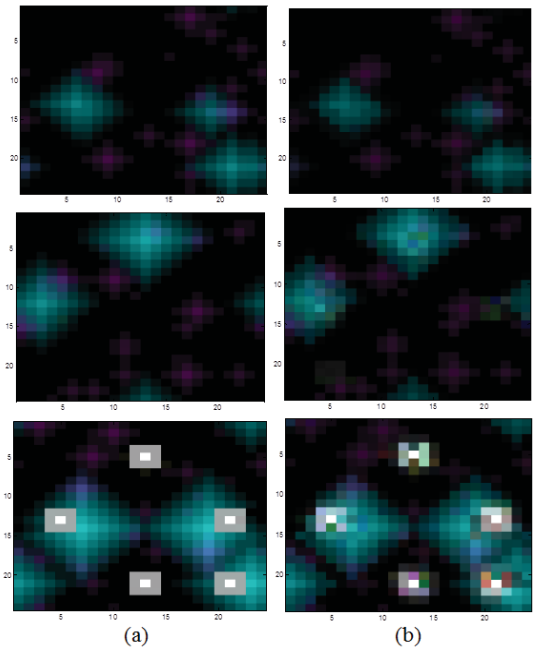


Fig. 3. Comparison of 2-D cross-sectional microstructures generated using HydratiCA-lite (a) and the neural network (b).

5.1 Microstructure correlation

Table V summarizes the correlation of the predicted voxels with the known test set voxels for each of the channels, indicating correlation between predicted microstructure and HydratiCA-lite generated microstructure.

Prediction accuracy is highly influenced by the presence of static voxels within the neighborhoods. Recall that all training neighborhoods must include at least nine dynamic voxels. Yet, some neighborhoods have fewer than nine dynamic voxels and are thus counted among the "excluded dynamic voxels." Nonetheless, these voxels must be predicted by the trained model in order to generate complete microstructures. This is error prone since the trained model does not have experience (opportunity) to learn from such neighborhoods.

Table V. R^2 Statistic for Last Step Predicted

SET	RED	GREEN	BLUE
Dynamic voxels (*Test*$_1$)	0.8053	0.8918	0.9191
Excluded dynamic voxels (*Test*$_1$)	0.7222	0.7319	0.7017
Dynamic voxels (*Test*$_2$)	0.8951	0.8346	0.79
Excluded dynamic voxels (*Test*$_2$)	0.7889	0.6754	0.789

5.2 Solution Composition

Global system profiles were generated and used to assess predictive accuracy of the trained model. Fig. 4 shows the global volume fractions of the solid phases for *test*$_1$ and *test*$_2$ compared to the network- (the trained model) predicted volume fractions. These results show that the prediction degrades immediately.

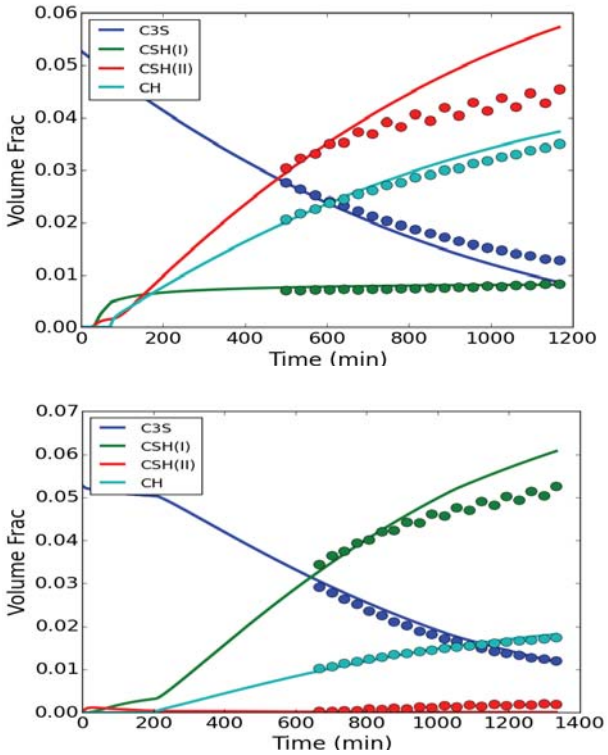


Fig. 4. Global outcomes for *test*$_1$ (a) and *test*$_2$ (b) for HydratiCA-lite (straight lines) and neural network (circles).

Regardless, the estimation error is not large and a correct evaluation and optimization of the prediction of the excluded dynamic voxels set would contribute to an enhancement of the

global prediction, as noted in Table V ($test_1$ results). The correlation between predicted values and the validation set values for the "excluded dynamic voxels" is considerably lower than for the dynamic voxels (an R^2 between 0.7 and 0.73 and between 0.8 and 0.92 respectively). Furthermore, the number of dynamic voxels also influences the prediction accuracy. In this case, the blue channel having the highest number of dynamic voxels exhibits the lowest error.

Unlike $test_1$, $test_2$ presented a larger amount of dynamic voxels in the red channel. Fig. 4b shows the global volume fractions computed by the HydratiCA-lite model and the dynamic neural network (predicted). In both cases depicted by Fig. 4a and 4b, the neural net predictions are in acceptable agreement with the results produced by the state of the art simulator (Hydratica-lite).

5.3 Execution Time

Table VI shows the CPU wall-clock time for HydratiCA-lite and the neural network, to predict microstructure for equivalent hydration times. The wall-clock time CPU speeds were measured on two different machines. HydratiCA-lite was run on an Intel Xeon 3.47 GHz multi-core processor using C++98. The autoregressive neural network was run on a Pentium 4 Xeon 2.8 GHz processor using MATLAB 2014a.

While this makes direct comparison of wall-clock times for the neural network and the CA models impossible, because MATLAB is deployed on a much slower machine but still clearly executes faster, it can be concluded that the neural network approach is computationally more efficient.

Table VI. Timing for 11.11 Hours of Hydration Simulation ($test_1$)

Simulator	CPU wall time for 11.11 hours of hydration (hours [s])	Number of processes
HydratiCA-lite	7.809 [28,112]	24
Autoregressive Neural Network	0.01556 [56]	1

6 Conclusions

These results show that the neural network (trained model) is able to make reasonable predictions for systems of unknown kinetics suggesting that there is considerable potential for the use of neural network models for fast generation of microstructure.

An autoregressive neural network with short memory was trained to generate microstructure for the hydration of tricalcium silicate (C_3S), the major component of Type I/II portland cements. This project utilized microstructures generated by a kinetic cellular automaton, a stochastic simulator. Global kinetic outcomes for the CA simulator were shown to be consistent with a continuum solution for identical kinetic and thermodynamic inputs. A single training data set was used for network learning.

The trained network was found to reproduce the training data set well, having an R^2 metric on the order of 0.98 for all channels. The trained model did a good job at predicting microstructure for different initial conditions (different number and location of C_3S particles) producing R^2 values between 0.8 and 0.92, with the outcomes improving with the number of dynamic voxels to be predicted. The trained model did a surprisingly good job even when the kinetic parameters (rates of reaction) were drastically changed along with the initial conditions, resulting in R^2 values between 0.79 and 0.9. Again, the predictive accuracy improved with increasing numbers of dynamic voxels. The neural network was able to reduce computation time by a nominal factor of 500× even when run on a slower computing platform.

This novel demonstration of the use of autoregressive neural networks for the prediction of microstructure in hydrating cement paste suggests that the approach is a viable option for dramatically reducing the computation time of kinetic microstructure generators.

7 References

[1] Thomas, J. J., Biernacki, J. J., Bullard, J. W., Bishnoi, S., Dolado, J. S., Scherer G. W. and Lutge, A., "Modeling and simulation of cement hydration kinetics and microstructure development," Cem. Concr. Res., 41(12), 1257-1278, 2011.

[2] Bullard, J., Enjolras, J., George, E., Satterfield, S. and Terrill, J. "A parallel reaction-transport model applied to cement hydration and microstructure development," Modeling Simul. Mater. Sci. Eng., vol. 18, no. 2, pp. 1-16, 2010.

[3] Biernacki, J. and Xie, T. "An advanced single particle model for C3S and alite hydration," J. Am. Ceram. Soc., vol. 94, no. 7, pp. 2037-2047, 2011.

[4] Bye, G., Portland cement, London: Thomas Telford, 1999.

[5] Biernacki, J., Bullard, J., Constantiner, D., Meininger, R., Juenger, M., Cheung, J., Hansen, W. and Hooton, R., "Paving the way for a more sustainable concrete infrastructure -a vision for developing a comprehensive description of cement hydration kinetics," National Institute of Standards and Technology Special Publication 1138, 2013.

[6] Bullard, J., "Modeling of Cement Paste Hydration and Microstructure Development," NIST Information Technology Laboratory, 2010.

[7] Bentz, D., "Three-Dimensional Computer Simulation of Portland Cement Hydration and Microstructure Development" J. Am. Ceram Soc., vol. 80, no. 1, pp. 3-21, 1997.

[8] Bishnoi S. and Scrivener K., "µic: A new platform for modelling the hydration of cements", Cem. Conc. Res., vol. 39, no. 4, pp. 266-274, 2009.

[9] Bullard, J., "A determination of hydration mechanisms for tricalcium silicate using a kinetic cellular automata model", J. Am. Ceram. Soc. vol. 91, no. 7, pp. 2088-2097, 2008.

[10] Garboczi, E. and Bentz, D., "Diffusivity-porosity relation for cement paste," in *In Proceeding from*

seminar on sulfate attack mechanisms (pp. 259-64), Quebec, Canada, 1999.

[11] Shane, J., Mason, T., Jennings, H., Garboczi, E. and Bentz, D., "Effect of the interfacial transition zone on the conductivity of Portland cement mortars," *Journal of the American Ceramic Society*, vol. 83, no. 5, pp. 1137-1144, 2000.

[12] Lawson, J., Phan, L. and Davis, F., "Mechanical properties of high performance concrete after exposure to elevated temperatures," US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2000.

[13] Devaney, J. and Hagedorn, J., "Discovery in Hydrating Plaster Using Multiple Machine Learning Methods," 2002. [Online]. Available: http://www.nist.gov/itl/math/hpcvg/upload/discov2002.pdf. [Accessed 02 May 2015]. [Garboczi and Bentz, 2000] Garboczi, E. and Bentz, D., "Percolation Aspects of Cement Paste and Concrete Properties and Durability," ACI Special Publication, 189, 2000.

[14] Wang, L., Yang, B., Chen, Y., Zhao, X., Chang, J. and Wang, H., "Modeling early-age hydration kinetics of Portland cement using flexible neural tree," *Neural Comput & Applic*, vol. 21, no. 5, pp. 877-889, 2010.

[15] Chen, Y., Yang, B., Dong, J. and Abraham, A., "Time-series forecasting using flexible neural tree model," *Information sciences*, vol. 174, no. 3, pp. 219-235, 2005.

[16] Danforth, C. and Yorke, J., "Making Forecasts for Chaotic Physical Processes," *American Physical Society, Physical Review Letters*, April 14, 2006.

[17] De Jesús, O. and Hagan, M., "Backpropagation Through Time for a General Class of Recurrent Network," International Joint Conf. on Neural Networks, Washington, DC, July 15–19, pp. 2638–2642, 2001.

[18] Bullard, J. and Flatt, R., "New Insights into the Effect of Calcium Hydroxide Precipitation on the Kinetics of Tricalcium Silicate Hydration," *Journal of the American Ceramic Society*, vol. 93, no. 7, pp. 1894-1903, 2010

[19] Perreault, E., Kirsch, R. and Acosta, A., "Multiple-input, multiple-output system identification for characterization of limb stiffness dynamics," *Biological Cybernetics*, vol. 80, no. 5, pp. 327-337, 1999.

[20] Hagan, M., "Dynamic Networks," [Online]. Available: http://hagan.ecen.ceat.okstate.edu/DynamicNetworks.pdf. [Accessed 02 May 2015].

[21] http://www.mathworks.com/products/matlab/

[22] Verma, N., "Future image frame generation using Artificial Neural Network with selected features," in IEEE Applied Imagery Pattern Recognition Workshop (AIPR), 2012 IEEE , pp.1,8, 9-11, 2012.

Modeling Wildfire Ignition Distribution and Making Prediction of Human-caused Wildfire

Weichen Ouyang[1], **Chia Yung Han**[1], **and Susanna T.Y.Tong** [2]
[1]Department of Electrical Engineering and Computing Systems, University of Cincinnati, Cincinnati, Ohio, U.S.A
[2] Department of Geography, University of Cincinnati, Cincinnati, Ohio, U.S.A
ouyangwn@mail.uc.edu, han@ucmail.uc.edu, tongty@ucmail.uc.edu

Abstract – *This paper proposes a further exploration of machine learning algorithms within the context of modelling the spatial distribution patterns of the human-caused wildfires over a Southern California landscape. In this research, the wildfire distribution problem is defined as a Binary Classification task conducted on a cellular lattice overlay the study area. A fifteen-year historical wildfire occurrence data as target variable was used, along with eight independent variables derived from anthropogenic factors such as distance measure to road-network and Wildland-Urban Interface. Meteorological factors such as temperature and humidity have also been used in the model training process. Both of the two machine learning algorithms, the Conditional Inference Tree and the Random Forest methods, combining with the Synthetic Minority Over-Sampling Technique, demonstrate a significant improvement over traditional method. And the predicted result shows that the location with high proximity with WUI and road tend to be more vulnerable towards wildfire incidence.*

Keywords: Machine learning, Wildfire, Random Forest, Ignition modeling, SMOTE

1 Introduction

Wildfire occurrence and fire intensity have been increasing in many states in western USA, especially in California, where is suffering from the record-breaking long drought climate. According to the National Interagency Fire Center (NIFC), in 2015, over 10 million acres of land have been burned in the U.S., making the year the worst in the wildfire history [1].

The deteriorating situation has brought much concern about wildfires and called for a better understanding of the cause and distribution of wildfire activity. Predictive models play a vital role in designing preventive strategies in wildfire management, whereas pure after-incidence suppression is costly and resource-limited. Therefore, comprehending driving forces of fire ignition and predicting where fires are likely to happen are becoming the research hotspot among scientific community.

Human-wildland interaction has been recognized as the dominant source of wildfire ignition over natural causes (lightning). During last decade, about 90% of wildfires in U.S. are caused by human, and for instance, in 2015, there were 58916 human-caused wildfires comparing the 9235 lightning-caused fires [1]. As the wildland–urban interface (WUI), the community built at the fringe of human developments and undeveloped vegetation expands, the proportion and the number of human-induced fires continue to rise [2]. Because the interactions between human and wildland tend to be spatially concentrated around WUI, it is important to understand how WUI and other human constructions around WUI, such as road-network, affect the wildfire frequency in a given region. It is natural then to consider using the distance measure between fire occurrence and both WUI and road-network to quantify the human influence on the frequency and spatial pattern of fire, as well as where fire risk is highest on a landscape. Several research work on the corresponding ignition-distribution models quantifying the relationship between anthropogenic factors and fire occurrence have been reported [3] [4].

Conceptually speaking, the basic approach of modelling ignition-distribution is to analyze wildfire ignition locations in relation to environmental variables that are assumed to influence the spatial distribution of ignitions. The model will estimate the response of wildfire ignitions to predictors (features) derived from these environmental variables. In the design of a model, the occurrence of an ignition event at a point in space will be encoded as a binary response (0 and 1), accompanied with absence data (points in space with minimal likelihood for fire occurrence). Then statistical or machine learning algorithms can be used to generate a classifier to classify every space unit in the study area to be either class 0 or 1, corresponding to low or high probability for a wildfire to happen at that corresponding point. However, because it is generally impossible to identify places where no ignition can occur, the definition of true absent points is not feasible. Thus, in most cases, when the fire presence locations (positive cases) are compared against negative cases for a given set of background environmental conditions [5], a severe "imbalanced dataset" problem will most certainly occur, wherein the number of data points belonging to one class, the majority class, is much more than the other, the minority class.

Over the past decade, several methods for building wildfire occurrence-distribution prediction systems or wildfire

risk assessment systems have been traditionally developed using the Binary Logistic Regression method, based on different variables, and scales. Some of which are widely accepted by the wildfire management authorities [3] [4] [6] [7]. Meanwhile, because machine learning algorithms have shown a high predictive accuracy and adaptability in Data Mining, AI and various other fields, some researcher also started to use those algorithms to model spatial distribution of wildfire occurrence or ignition. Amatulli [8] used Decision Trees to assess and generate a risk map of wildfire of Gargano Peninsula on Italy's East Coast; Artificial Neural Networks (ANN) was used to classify the space units of central Portugal into 5 class of risk ranks [9]; and more recently, an ensemble machine learning algorithm, Random Forest, has been used to model the fire distribution in a national forest in Michigan [10].

In comparison with the traditional method – Logistic Regression, the machine learning algorithms have their advantages. ANN has its merits in capturing non-linear relationship between the feature set and the response variable; Decision Trees, specifically the Conditional Inference Trees, are more adaptive to imbalance dataset and are more resistant to the overfitting problem than Logistic Regression; Random Forest, as an ensemble learning algorithm, usually outperforms other classification algorithms in terms of its predictive accuracy. The advantages of utilizing machine learning algorithms in modeling spatial distribution of wildfire over Logistic Regression (LR) will be further explored in this research. Two algorithms, namely, Random Forest (RF) and Conditional Inference Tree (CIT) are chosen in the comparison study. All three will be applied to building the predictive model of wildfire occurrence in the Santa Monica National Recreational Area, which is located at the West Shore of Los Angeles.

To be specific, the main objectives of this research are (1) identify and collect a combination of environmental and anthropogenic explanatory variables and then apply machine learning algorithms to produce predictive models for predicting patterns of the spatial distribution of human-caused wildfire over the landscape of a national forest – Santa Monica Mountains National Recreation Area; (2) build the model using the two widely-accepted classification algorithms, the Conditional Inference Tree and Random Forest, and investigate the results and compare them with the result from the model built with the Logistic Regression; and 3) explore different oversampling and under-sampling methods to find a usable algorithm to solve the "imbalance dataset" problem that exists in the related previous works.

2 Method

2.1 Study area

This research is conducted based on data collected for the region in the Santa Monica National Recreation Area plus an adjacent town, Thousand Oaks (Figure 1). This mountainous area resides on the west shore of Los Angeles, on the shore line of Pacific, across the county of Ventura and L.A. The study area includes a substantial amount of WUIs, and has experienced a series of wildfire occurrences over last decade. The whole area covers about 1 billion square meters, to facilitate the model building process, a fishnet (cellular lattice or grid) is created, consisting of approximately 1 million spatial cells of size of 30m x 30m (meter) each.

Figure 1. The Study Area – Santa Monica Mountain, CA

2.2 Problem definition

To model the spatial distribution of wildfire over this study area, a Binary Classification Problem is defined over the 1 million cells that constitute the study area. Each cell will be measured by eight features, five anthropogenic and three weather measurements (detailed in later sections), and each cell will eventually be classified by the classifier, which is generated by one of the three classifier construction algorithms (Logistic Regression, CIT and RF), to be one of the two classes: 1, representing high probability of fire occurrence and 0, the opposite. This classification result is a form of long-term risk assessment, that is, the evaluation of the proneness to wildfire incident over a period of time, based on the characteristics of climate and impact of anthropogenic factors in the near neighborhood of that specific cell.

2.3 Response variable

2.3.1 Response variable – fire ignitions

The Santa Monica ignition dataset, acquired from the USGS historical wildfire record from 2000 to 2014, contains 210 geo-coordinated points [11]. These 210 points are resampled using the given grid. By doing a careful check, there are no two points residing within a common cell. To prepare a dataset for constructing classifier, 790 random points are generated using ArcGIS at a minimal 300 meters away from each other. This is to guarantee that the 790 points are scattered across the whole region, to capture the variance in the 8 features in the study area. These random points will serve as the negative (0) cases, and together with the 210 positive cases, a 1000-data-points dataset is formed. Figure 2 shows the ignition points on this map. Also shown are the road network

and the WUI polygons whose borders are the human nature interface.

represent the max range people usually get into the remote wildland. Detail description are in Table 1.

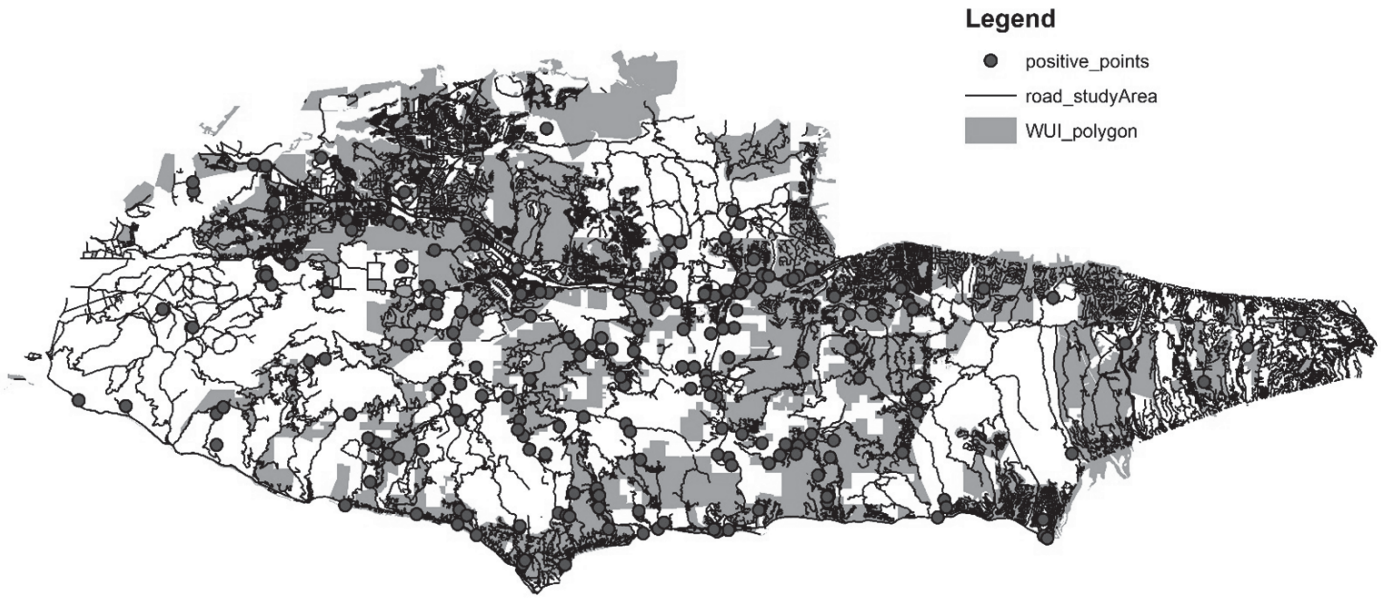


Figure 2. Ignition points (year 2000-2014), road network and WUI

2.4 Features

2.4.1 Independent variables – anthropogenic

Human-caused ignitions were concentrated close to roads, in high road density areas, and near the WUI interface [12]. As shown in Figure 2, the Ignition data in Santa Monica Mountains also demonstrate a trend that they are to be found near human developments and communities. In this research, three types of human construction artefacts are recognized: the road network, human development (man-made construction units), communities (in terms of WUI polygon). The road network data is acquired by downloading TIGER/Line files updated for "US census 2010" from the US Census Bureau. Human development data is derived from the land coverage data of the study area, provided by the National Park Service (NPS). The land coverage data classifies every 30-meter by 30-meter cell to be a kind of the coverage types, four of which are human development areas. The cells are extracted from the file and formed the map layer of human development. WUI data is provided by the Silvis Project from University of Wisconsin-Madison [2].

Accordingly, five Independent variables are derived from the above three: distance to road, distance to WUI, distance to human development, level of development (derived by constructing a kernel density estimate of human development over 1000 meters), population density (derived by constructing a kernel density estimate of population density measure of each WUI over 1000 meters). The 1000-meter threshold is chosen to

2.4.2 Independent variables – climate

Weather plays a vital role in fire ignition process, low temperature, heavy precipitation and high humidity will significantly diminish the chance of fire occurrence. Thus, to capture impact to ignition distribution from the biophysical characteristics of the study region of chosen, 30 weather stations scattered within and surrounding the area had been chosen and 3 variables had been collected and aggregated over the 15 years span to reflect the local climate (2000 to 2014): mean January temperature, mean July temperature and mean annual humidity. The 30-point data are then used to generate, using a 5-point interpolation, a continuous surface with all 3 variables being populated across the whole landscape. Details of the variables used in this study are listed in Table 1.

2.4.3 Data manipulation

Since all the cells in our grid system is of the size 30 meter by 30 meter, all of the above 8 independent variables are resampled to this 30-m resolution. Consequently, all the one million cells would now possess a feature vector of dimension eight, and then the 790 geo-coordinated negative points can be used to extract the corresponding feature vector from the cells.

Table 1. List of variables

Variable	Resolution	Source	Description
Response Variable			
Ignition points	point	USGS	N=210, from 2000 to 2014
Independent variables			
Distance to Road	30m	US Census TIGER/Line files	Nearest Perpendicular distance
Distance to WUI	30m	Silvis Project mapping data	Nearest Perpendicular distance
Distance to Development	30m	NPS data store	Nearest Perpendicular distance
Development level	1000-m	NPS data store	Kernel Density Estimate
WUI population Density	1000-m	Silvis Project WUI mapping data	Kernel Density Estimate
Mean January Temperature	30m	Mesowest weather system	5-neighbor interpolation
Mean July Temperature	30m	Mesowest weather system	5-neighbor interpolation
Mean Relative Humidity	30m	Mesowest weather system	5-neighbor interpolation

2.5 Create a balance dataset

The dataset thus generated is a typical "imbalance dataset" since the positive: negative rate is 1:4, which means that it would greatly decrease the overall sensitivity of the classifier. Because we are aiming to build a classifier that identify the vulnerable spot that is most likely encounter wildfire in this study, sensitivity is an essential performance measurement in the model evaluation process. Under the 1 to 4 ratio of positive to negative cases, the model will be most likely tuned towards recognizing every case to be negative. This is common in the modeling spatial distribution of wildfire problem. In [7], Syphard also used a dataset with 1:5.5 positive to negative rate, yielding an Area Under Curve (AUC) of 0.71 in the Receiver Operating Characteristic (ROC) test. However, in this study, the original imbalance dataset yields a 13% sensitivity classifier using CIT algorithm, basically meaning the classifier can only recognize one case in every ten positive cases, making the generated classifier useless.

To overcome this, certain over-sampling and under-sampling technique is needed to create a new "balanced" dataset as the input to our training procedure. The Synthetic Minority Over-Sampling Technique (SMOTE) is used to over-sampling the minority class [13]. The synthetic samples are created in a less application-specific manner, by operating in "feature space" rather than "data space". The minority class is over-sampled by taking each minority class sample and introducing synthetic samples along the line segments joining any or all of the k minority class nearest neighbors. That is, the samples are generated as follows. First, take the difference between the feature vector (sample) under consideration and its nearest neighbor. Multiply this difference by a random number between 0 and 1, and add it to the feature vector under consideration. This causes the selection of a random point along the line segment between two specific features. This approach effectively forces the decision region of the minority class to become more general. For our dataset, SMOTE is executed on the original 1000-data-points dataset, a new balanced dataset which contains 2730 data points, 1470 positive and 1260 negative. And the positive to negative ratio is altered to be 1.167: 1.

2.6 Model building

Before entering the model building phase, an exploratory analysis is conducted to eliminate feature variables that are highly collinearly related to other features. Features such as "development level" and "mean July temperature" were discarded. Thus, the feature dimension had been reduced to be 6. And the dataset containing 2730 data points, will be divided into training set and test set based on a ratio of 7:3.

2.6.1 Logistic regression

Logistic Regression models are statistical models which provide insights into the relationship between a qualitative dependent variable, dichotomous in the present case, and one or more independent explanatory variables, whether qualitative or quantitative. The mathematical expression of LR models is:

$$y_i = \frac{e^{(\beta_0+\beta_1 \cdot x_{1i}+\cdots+\beta_k \cdot x_{ki})}}{1+e^{(\beta_0+\beta_1 \cdot x_{1i}+\cdots+\beta_k \cdot x_{ki})}} \qquad (1)$$

In this work, the Logistic Regression model is developed using a forward stepwise procedure in which the independent variables are introduced into the model one by one. According to the result of evaluating improvement in the model by adding the variable, as measured by the Akaike Information Criterion (AIC), a decision is made to keep or discard that variable.

2.6.2 Conditional Inference Tree

Both traditional classification and regression tree (CART) algorithm and the CIT method recursively perform univariate splits of the dependent variable based on values of a set of covariates. CART and related algorithms usually employ information measures for selecting the current covariate. It uses GINI coefficient to evaluate the likelihood of a random sample chosen at this node being misclassified in the child-nodes.

According to [14], CIT avoided the following variable selection bias of the CART methods: They tend to select variables that have many possible splits or many missing

values. Unlike the others, CIT uses a significance test procedure in order to select variables, instead of selecting the variable that maximizes an information measure (e.g., Gini coefficient).

In this study, CIT is parameterized to use 5% as the maximum allowed p-value for permutation test conduct at each splitting node, a minimum node size of 20 for inner tree nodes, a minimum node size of 7 for leave nodes, and no restrictions on the tree depth.

2.6.3 Random Forest

Random Forest is an ensemble Machine Learning technique which works in a parallel fashion similar to other meta-algorithms like bootstrapping [15]. The classification result in RF is generated by averaging the predictions of many individual decision trees, each is constructed based on a subset of the training data. This algorithm is proven to be more stable and accurate than any single decision tree algorithm.

RF can be parameterized according to the number of trees averaged in the forest (*ntrees*), the number of predictor variables randomly selected at each iteration (*mtry*), and the minimum number of observations at end nodes (*nodesize*), which can decrease the length of the tree branches and thus simplifying the structure of the trees. All combinations of five *ntrees* levels (500, 1000, 1500 and 2000) and three *mtry* levels (from 1 to 3) were tested. The *nodesize* parameter was left at its default value. The values of the parameters in the final model were *mtry* = 2 and *ntrees* = 500. Models with higher values of these parameters did not improve accuracy significantly.

3 Result

3.1 Model evaluation and comparison

To compare the prediction accuracy among the three models, the area under the curve (AUC) of the receiver operating characteristic (ROC) chart is used along with three other common measures: accuracy, sensitivity, and specificity. Sensitivity refers to the proportion of ignitions that is correctly predicted, and specificity the proportion of non-ignitions correctly predicted. AUC ranges from 0.5 to 1, in general, an AUC value above 0.8 indicates a good model performance.

In summary, unlike previous works, there are two datasets used in this study, a synthetic one generated by SMOTE with 2730-data-points, and the original sample of 1000 points, the latter is a subset of the former. The performance of models on both datasets will be evaluated.

3.1.1 Performance over synthetic dataset

Both the CIT and the traditional method LR show a good classification accuracy over the synthetic dataset, however, the RF yields a result of approximately 95% of correct overall prediction in the test set, as shown in Table 2.

Table 2. Performance of all three algorithms on prediction over synthetic test set

	LR	CIT	RF
Accuracy	0.718	0.755	0.95
Sensitivity	0.819	0.754	0.962
Specificity	0.577	0.755	0.933
AUC	0.756	0.804	0.983

In terms of sensitivity, LR outperforms CIT by about 6%, which may be due to the synthetic data points' strength in the specificity of the decision plane causing the increased accuracy of linear relation. This phenomenon will increase the LR classification success rate and will decrease CIT's rate. At the other hand, RF, as an ensemble algorithm will keep the stability by average out the variance caused by high specificity issue caused by synthetic points. Thus, Random Forest has the best performance in every way for synthetic dataset.

3.1.2 Performance over original dataset

The correctness of classification has much more significance in the original dataset, since the result will be directly used to generate the predicting map for preventive strategy design and research.

Table 3. Performance of all three algorithms on prediction over original dataset

	LR	CIT	RF
Accuracy	0.576	0.753	0.936
Sensitivity	0.800	0.719	0.971
Specificity	0.516	0.762	0.927
AUC	0.717	0.802	0.988

As shown in Table 3, all three algorithms suffer some loss in the overall accuracy, while LR suffers the most, which may be due to the imbalance nature of the original dataset reflecting in terms of low linear separability. But RF shows its robustness against imbalance issue. Hence it is the best classification algorithm for use.

Furthermore, both machine learning algorithms show the greater predicting power in terms of AUC, as shown in Figures 3, 4, and 5. The AUC value of 0.756 is obtained for the LR method, while CIT's AUC values is about 0.8. And the AUC value for the RF reaches above 0.98. This measure for the three methods clearly shows that their better identification ability of the actual ignition data, which demonstrate their value in replacing the traditional LR algorithm in construct predicting model of wildland fire distribution.

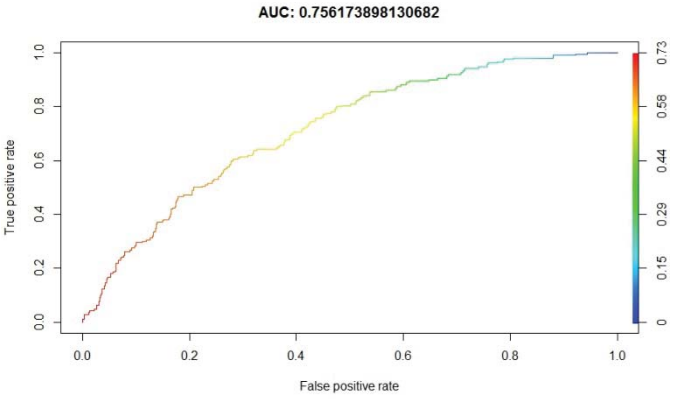


Figure 3. LR ROC plot against original dataset

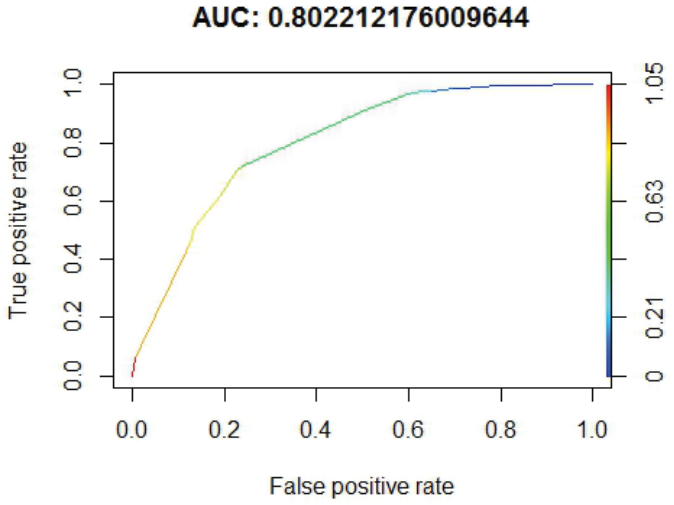


Figure 4. CIT ROC plot against original dataset

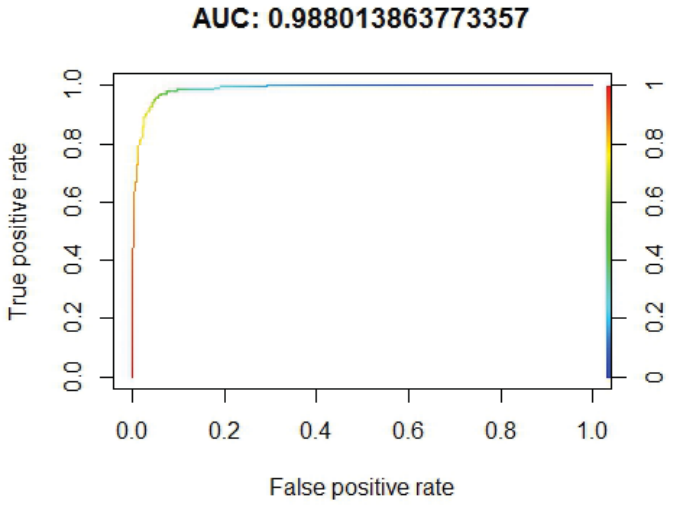


Figure 5. RF ROC plot against original dataset

3.2 Variable Importance

Figure 6 shows the variable importance graph generated by using the RF method.

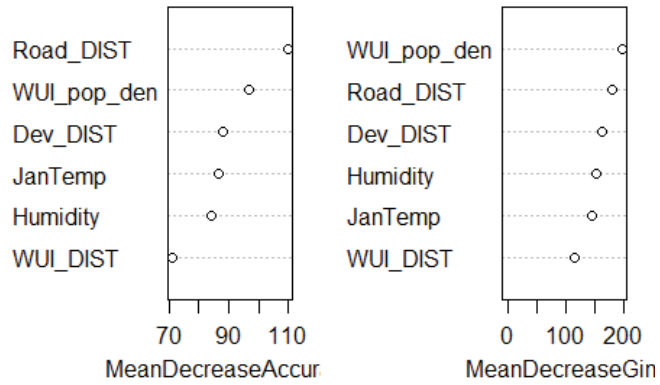


Figure 6. Variable Importance Chart in Random Forest model

The two features that show the highest importance value, indicating their predictive power in the model generated by Random Forest are Distance to Road and WUI population density kernel density estimate. The result clearly reflects the fact that human-caused wildfire tend to occur at locations within a convenient distance from roads and of higher residential human traffic.

4 Conclusion

Machine learning models improve the prediction accuracy of traditional regression methods. Either RF or CIT models yield an improvement in accuracy over LR methods for wildfire occurrence assessment, according to AUC values. More specifically, the RF algorithm seems to be the best choice not only due to its higher accuracy, but also because its calibration is easier as it involves fewer parameters. Another advantage of RF is its cartographic outputs, which seem to be more realistic than those from other models due to RF's higher spatial variability and therefore greater spatial discriminatory power. This enables RF to provide a better reflection of variability in wildfire occurrence linked to heterogeneity of landscapes and human activities.

Prediction of wildfire occurrences may benefit from using multiple approaches, yielding a range of predictions in a concurrent manner rather than relying on a single map. Regardless of the method considered, both the *Distance to Road* and the *surrounding* WUI population density have proved to be the variables most closely related to fire occurrence, and hence presenting themselves to be the most important predictors in the models. In any case, we can draw the conclusion that fire occurrence in Santa Monica Moutains is mainly related to the expansion of human settlement pressuring on wildlands and to accidents or negligence in the course of human-wildland interaction.

5 References

[1] NIFC, "NICC Wildland Fire Summary and Statistics Annual Report," 2016. [Online]. Available: http://www.predictiveservices.nifc.gov/intelligence/2015_Statssumm/annual_report_2015.pdf

[2] V. C. Radeloff, R. B. Hammer, S. I. Stewart, J. S. Fried, S. S. Holcomb and J. F. McKeefry, "THE WILDLAND–URBAN INTERFACE IN THE UNITED STATES," *Ecological Applications,* vol. 15, no. 3, pp. 799-805, 2005.

[3] J. Martínez, C. Vega-Garcia and E. Chuvieco, "Human-caused wildfire risk rating for prevention planning in Spain," *Journal of Environmental Management ,* vol. 90, no. 2, pp. 1241-1252, 2009.

[4] M. Rodrigues, J. de la Riva and S. Fotheringham, "Modeling the spatial variation of the explanatory factors of human-caused wildfires in Spain using geographically weighted logistic regression," *Applied Geography ,* vol. 48, pp. 52-63, 2014.

[5] J. Elith and J. Franklin, "Species Distribution Modeling," in *Encyclopedia of Biodiversity (Second Edition)* , Second Edition ed., S. A. Levin, Ed., Waltham, Academic Press, 2013, pp. 692-705.

[6] E. Chuvieco, I. Aguado, M. Yebra, H. Nieto, J. Salas, M. P. Martín, L. Vilar, J. Martínez, S. Martín, P. Ibarra, J. de la Riva, J. Baeza, F. Rodríguez, J. R. Molina, M. A. Herrera and R. Zamora, "Development of a framework for fire risk assessment using remote sensing and geographic information system technologies," *Ecological Modelling ,* vol. 221, no. 1, pp. 46-58, 2010.

[7] A. D. Syphard, V. C. Radeloff, N. S. Keuler, R. S. Taylor, T. J. Hawbaker, S. I. Stewart and M. K. Clayton, "Predicting spatial patterns of fire on a southern California landscape," *International Journal of Wildland Fire,* vol. 17, no. 5, pp. 602-613, 2008.

[8] G. Amatulli, M. J. Rodrigues, M. Trombetti and R. Lovreglio, "Assessing long-term fire risk at local scale by means of decision tree technique," *Journal of Geophysical Research,* vol. 111, no. G4, 2006.

[9] M. P. de Vasconcelos, S. Silva, M. Tome, M. Alvim and J. C. Pereira, "Spatial prediction of fire ignition probabilities: comparing logistic regression and neural networks," *Photogrammetric engineering and remote sensing,* vol. 67, no. 1, pp. 73-81, 2001.

[10] M. Bar, A. D. Syphard, S. I. Stewart and V. C. Radeloff, "Wildfire ignition-distribution modelling: a comparative study in the Huron–Manistee National Forest, Michigan, USA.," *International Journal of Wildland Fire,* vol. 22, pp. 174-183, 2013.

[11] USGS, "Federal Wildland Fire Occurrence Data," United States Geological Survey, 4 August 2015. [Online]. Available: http://wildfire.cr.usgs.gov/firehistory/data.html. [Accessed 15 March 2016].

[12] G. Narayanaraj and M. C. Wimberly, "Influences of forest roads on the spatial patterns of human- and lightning-caused," *Applied Geography,* vol. 32, pp. 878-888, 2012.

[13] N. Chawla, K. W. Bowyer, L. O. Hall and P. W. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research,* vol. 16, pp. 321-357, 2002.

[14] T. Hothorn, K. Hornik and A. Zeileis, "Unbiased Recursive Partitioning: A Conditional Inference Framework," *Journal of Computational and Graphical Statistics,* vol. 15, no. 3, p. 651–674, 2006.

[15] L. Breiman, "Random Forests," *Machine Learning,* vol. 45, pp. 15-32, 2001.

Noise and Error Prediction for Neural Networks

D. Vrajitoru[1] **and K. Albelihi**[2]
[1]Department of Computer and Information Sciences, Indiana University South Bend, South Bend, IN, USA

Abstract – *In this paper, we are studying the application of a neural network to a problem related to autonomous car driving. In particular, we are interested in the correlation between noise in the data and the average error that can be achieved by the neural network, and study the issue using several data collections. We propose a measure of the noise that can indicate issues in the collected data and can help improve the learning process. We are also interested in ways to use symmetry in the data for this particular problem and analyze how we can take advantage of it for a better training of the network.*

Keywords: neural networks, noise measurement, symmetry

1 Introduction

In this paper we propose a study of the correlation between noise in the data and error in artificial neural networks (NN). For this, we use a problem where a NN is applied to predict the steering angle for an autonomous car driver. This problem presents an interesting challenge for the NN because of the size of the data - about 1000 data points after filtering - and because of the complex nature of car driving rules. We started from a previous research that was published in [1] and [2], and used the data collected for the Master's thesis in the experiments.

Autonomous car driving is a very relevant topic at the moment. Significant progress is made all the time and self-driving cars are currently being tested on the road. More and more, car companies provide assistive driving technologies, such as parallel parking and accident prevention. While it may be hard for academic research to keep up with the industrial pace, this still represents an interesting and challenging problem for us and an ideal test case for learning algorithms. Our interest in the subject is in seeing how well NNs can help with the driving problem.

Noise is a fascinating subject with many applications in computer science. In computer graphics in particular, noise generation plays a big role in procedural content generation for special effects and games. One of the popular methods is that of Perlin Noise [8]. It can be used to generate realistic-looking clouds, rocks, wood, and other textures. More complex methods such as the M-Noise [10] using several noise octaves were developed from it later.

Noise was also used in conjunction with the study of NNs and other related systems. For example, [5] proposes a method using NNs to reduce background noise for a better performance of speech-recognition systems. In [6], Bayesian networks are being used to analyze noisy biological data and uncover key biological features in cellular interaction. In [9], the authors use Short-Time Fourier Transform, self-organizing maps, and NNs to analyze the integrity of audio signals and their noise composition. Some studies [11] are concerned with developing NN-based systems that are robust with respect to the noise in the data. In [12], the authors use a NN to filter Monte Carlo noise from images.

In this paper, we continue the research started in [1, 2, 4] and examine the robustness of the NN when learning the data for this problem and how the variation of the average error with respect to the structure of the NN's hidden layers. Then we examine how to take advantage of symmetry in the data. Finally, we look at ways to visualize the amount of noise in the data and how the noise measure can be used as a tool to explain the average error of the NN.

2 Autonomous Car Driving and NN

The research in this paper is a continuation of the projects described in [1], [2], and [4]. These previous papers describe a Fuzzy-Logic car pilot called Epic that was capable of learning by hill-climbing techniques. Next, a system called Gazelle improved on various aspects of Epic, added more learning components, and implemented a unit to deal with opponents.

Moreover, in Gazelle we introduced the idea of collecting driving data from the procedural pilot and training a NN with it. The experiments showed that the road performance of the NN is not necessarily better than that of the procedural pilot, but it can improve some aspects such as road stability while reducing damage.

Both Gazelle and Epic drive a car procedurally in a car race simulation environment called TORCS. This software provides several pilots available for racing in a graphical environment and allows the user to compete with them. The user has the option to either by driving a car manually, or to program their own pilot and integrate it in the system. For the latter option, TORCS provides a `CarState` class with road information such as the free distance ahead, position of the car on the road, angle with the centerline, opponents present in a 100m radius around the car, and others. The program must

respond with changes to the steering wheel, the gas or brakes, and the gear. All of them are assembled into an object of the `CarControl` class, which is returned to the TORCS race server. Figure 1 shows a racing track available in TORCS and the controlled car driving on it.

Fig. 1. The Alpine2 track (left) and a snapshot of a driving car on it (right)

The procedural pilot that we wrote for Epic and subsequently improved for Gazelle contains several control units. We start by determining the target angle for steering, which is the most important element of driving, as it establishes the trajectory. An opponent modifier unit may change this target angle based on opponents' presence. Next, a target speed is chosen so that the car can achieve a turn by the target angle, while also trying to maximize the speed based on a given limit. This speed change is translated into acceleration or braking and can also be changed by the opponent modifier unit. The gear is adjusted in the last place.

These pilots contain other units to deal with special situations. A module is taking care of the car being outside of the road, and strives to get it back on track. Another module drives the car in reverse when it is stuck facing the wrong way on the road. A learning unit can adjust the speed from one lapse of the track to another based on the amount of damage experienced so far. Another learning unit remembers trouble spots such as sharp turns in the road so that the speed can be adjusted the next time we approach them.

The current research continues our previous work in several directions. First, we investigate the best settings for the NN that can allow it to learn the function governing the steering angle. Second, we look at a better way to exploit the available data by using properties of symmetry. Third, we analyze the connection between the noise in the data and the average error of the trained NN.

In the model we used to apply the NN to this problem, we chose 5 input variables that have significant influence on the steering decisions. The output value is the steering angle. In [2], data was collected by running the procedural driver on five tracks chosen from the available ones. A filtering process was also used to ensure a balanced distribution of the data in terms of output values.

Figure 2 shows an example of the filtering process: unfiltered and unbalanced data distribution in the upper part, and more balanced data distribution in the lower part. The intervals represent a discretization of the target output value in intervals of 0.1 length, going from -0.6 radians at the bottom to 0.6 radians at the top, and the number 8 marking a special interval (-0.01, 0.01]. The filtering process was intended to insure that the NN is not over-trained on some regions of the output target value range. For example many data points of the initial collection were very close to the value 0, because those would occur on any stretches of the road that are almost straight.

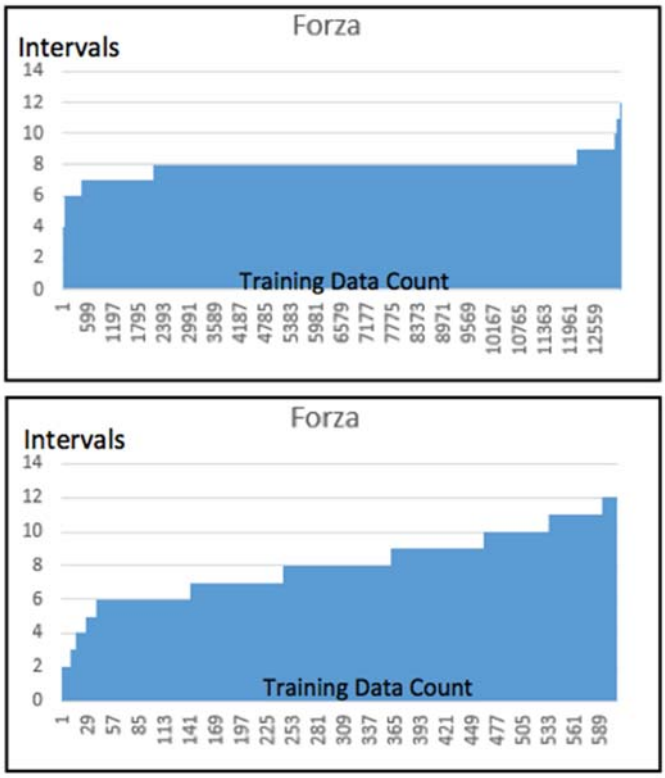


Fig. 2. Filtering the training data for the Forza track

Figure 3 shows a car with the road borders outlined and the input values that we have chosen to feed the NN, labeled from 1 to 5. The first, labeled 1, is the current angle of the car with the road centerline. Variable 2 is the lateral track position of the car and varies from -1 on the left border to 1 on the right border. Values outside of this range signify that the car has exited the road. Variable 3 represents the free distance ahead of the car. Variables 4 and 5 are the differences between the distance ahead and the free distance at 10^o angles left and right from the car direction. In this figure, the parameter 5 should have a negative value.

We selected three tracks that the data was collected for in [2] to perform our current experiments on, Alpine2, ETrack5, and Forza. We selected a fourth track, ERoad, to test the trained NN with new data that was not used for training.

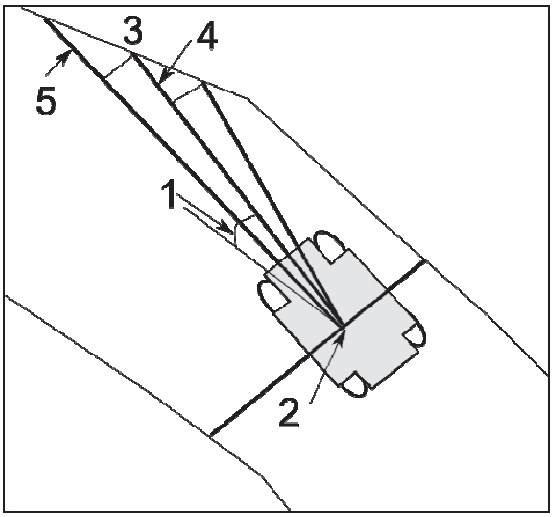


Fig. 3. Car state and five NN input variables

3 Using Problem Symmetry

Some functions by their very nature present some intrinsic symmetries. Depending on the way the data collection is done, these may or may not be present in the training data.

In our case the problem we are trying to solve is spatially symmetrical. Figure 4 shows a mirror of Figure 3 along the vertical axis. If the situation was thus reversed, the target output value for steering the car should be the negative of the target output value in Figure 3.

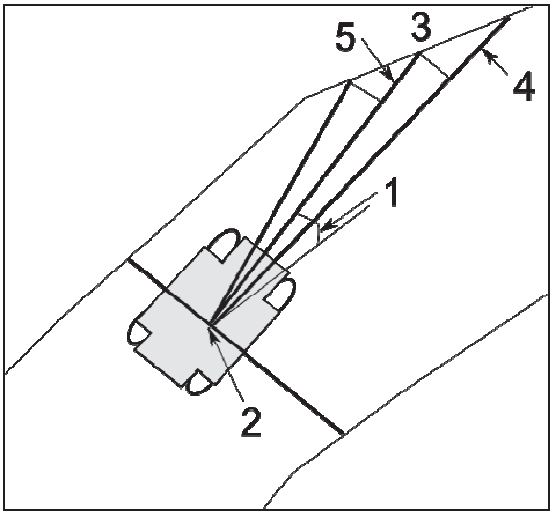


Fig. 4. Mirror situation of Figure 3

Thus, to mirror the problem, we need to multiply the first two variables and the target output value by -1. Then the values of the 4th and 5th variables must be swapped. The value of the third variable remains the same. By performing this operation on each data point that was collected, we can train the NN with data that accurately reflects the symmetrical nature of the problem, in hopes to achieve better driving performance.

4 NN Experimental Results

In this section we present the experimental results of training the NN with the data from each of the three tracks, with and without symmetry. For each track, we present the resulting average error both during the training process and for testing the trained NN on the data obtained from the fourth track, ERoad. The error is defined as the average difference between the NN's output and the target value for each data point, as follow:

$$error = \sqrt{\sum_{x\ in\ data}(output(x) - target(x))^2 / \#data}$$

Tables 1, 2, and 3 summarize the results for the tracks Alpine2, ETrack5, and Forza respectively. In all three tables, the training and testing results represent 1000 training iterations where the NN learns all the data points in the file by back-propagation. For all the tables, the testing data was obtained using data from the ERoad track. In all three tables, the NN contains four layers. The first one is the input layer and always contains 5 neurons. The number of layers in the second and third layers, denoted by HL1 and HL2 (for hidden layer), varies in the tables and is marked in the first two columns. The last layer is the output one and always contains a single neuron.

Table 1. Training and testing results for the Alpine2 data in 1000 iterations

		No Symmetry		Symmetry	
HL1	HL2	Train Error	Test Error	Train Error	Test Error
8	3	0.154	0.139	0.164	0.143
10	5	0.135	0.179	0.156	0.206
12	5	0.136	0.181	0.156	0.141
15	6	0.140	0.198	0.156	0.185

Table 2. Training and testing results for the ETrack5 data in 1000 iterations

		No Symmetry		Symmetry	
HL1	HL2	Train Error	Test Error	Train Error	Test Error
8	3	0.082	0.200	0.110	0.177
10	5	0.081	0.174	0.088	0.156
12	5	0.089	0.207	0.087	0.159
15	6	0.109	0.233	0.111	0.182

Table 3. Training and testing results for the Forza data in 1000 iterations

		No Symmetry		Symmetry	
HL1	HL2	Train Error	Test Error	Train Error	Test Error
8	3	0.046	0.149	0.042	0.145
10	5	0.044	0.154	0.045	0.149
12	5	0.044	0.161	0.049	0.142
15	6	0.057	0.165	0.046	0.143

From Table 1 we can see that for the Alpine2 track, the best configuration for the training data both with and without symmetry is that of 10 + 5 hidden neurons. For the test data, the best configuration is that of 8 + 3 hidden neurons without symmetry. The track ERoad is not a very symmetrical one in terms of turns, so for future research we will be looking for a track that has both left and right turns for more accurate testing.

Even though the ERoad track is not symmetrical, we can see that for a larger number of neurons, 12 and 15 on HL1, using data symmetry has resulted in a better average error on the test data.

From Table 2 we can see that for the ETrack5 data, the models of 10 + 5 and 12 + 5 neurons provide the best training error without and with the use of symmetry respectively. For the test data, the best result is shown by the 10 + 5 neurons configuration with the use of symmetry. In this case, using symmetry is consistently better for the test data. This can be explained by the fact that the ETrack5 track contains less turns in the road in both directions than the Alpine2 track.

It is interesting to note that the error in the test data is generally higher for the ETrack5 than it is for Alpine2, even though the error on the training data is lower. This can be explained by the fact that the Alpine2 track being more complex, it provides more opportunities for the NN to train well to calibrate its output for a larger variety of situations.

Looking at Table 3, it seems that the training error is substantially lower than for Alpine2 and for ETrack5. The best result in this case was obtained by the 8 + 3 neurons model with symmetry. For the test error the best result was obtained by the 12 + 5 neurons model, also with symmetry. Since this is a simpler track than the first two, the overall error is lower and the symmetry seems to help lower the error consistently, especially on the test data.

We can also wonder if more training iterations could have led to different results. Figures 5 and 6 show a plot of the different models we used over 5000 iterations for the ETrack5 data with symmetry for the training and testing cases respectively.

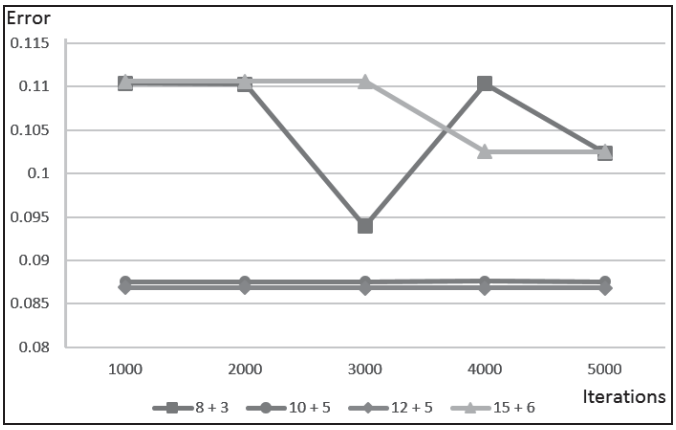


Fig. 5. Training error on the ETrack5 data with symmetry

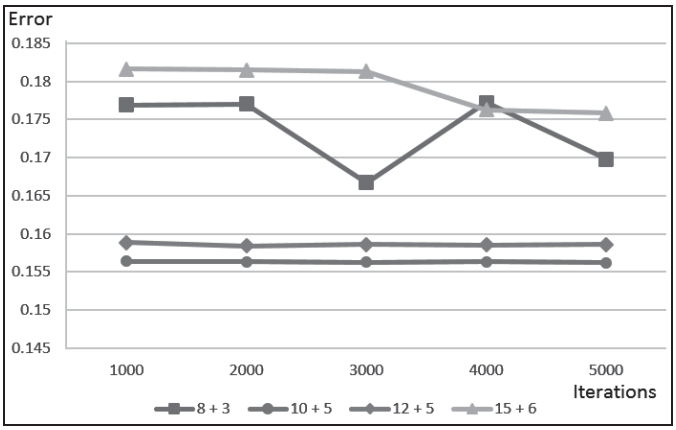


Fig. 6. Testing error on the ETrack5 data with symmetry

From these two figures we can see that for some of the models (10 + 5 and 12 + 5), training past the initial 1000 iterations led to very little change in the error both for training and for testing. For the 8 + 3 model, there is a substantial decrease of the error both in training and testing modes around 3000 iterations, followed by another increase. For the 15 + 6 model, the back-propagation seems to increase its efficiency after 3000 iterations and the error continues to decrease after that, although between 4000 and 5000 iterations the learning process slows down.

These figures show that smaller NNs are likely to learn faster, so 1000 iterations might be sufficient for the model 8 + 3 neurons. Larger NNs are likely to learn more slowly, and in this case a larger number of iterations must be tried to give the back-propagation algorithm a chance to work. However, the overall results do not suggest that larger networks always have a chance to achieve a better error than smaller ones, and experiments can help calibrate the appropriate size of the layers.

5 Noise Analysis

In this section we are analyzing the noise in the collected data for the three tracks we chose for training: Alpine2, ETrack5, and Forza.

5.1 Noise Measurement Function

Noise can be defined in a variety of ways. It is a fuzzy notion that indicates that adjacent data points in an area, mostly of visual or audio nature, differs in random, unpredictable, or unexpected ways. The idea of entropy is often connected to it, for which Boltzmann's and Gibb's equations are available as forms of measurement [3]. These equations are related to the noise measurements ideas of noise figure and noise factor used in acoustics and radio engineering. These do not directly apply to the analysis we are trying to perform. Usually logarithmical averaging methods are used for analyzing large collections of data, but those do not represent the picture we want to look at for our problem.

The idea in the noise measurement function we propose here is to be able to showcase how the amount of difference in the values of the output varies as a function of the amount of difference in the input. The kind of situation that is most interesting to us is where some input values are very close to each other, but the output values associated with them are quite different. This is likely to cause some difficulties for the NN in the training process. So we define here a function N connecting the difference in the input values to the difference in the output values.

Let x_1 and x_2 be two input data points or vectors. In our case, each of them contains 5 values for the 5 input parameters we chose for training the NN. Let $\Delta x = d(x_1, x_2)$ be the simple Euclidian distance between the two points. Let $y_1 = f(x_1)$ and $y_2=f(x_2)$ be the target output values for each of these input points, and $\Delta y = |y_1 - y_2|$. Then as a measure of noise in the data we chose to examine the shape of the two dimensional function

$$\Delta y = N(\Delta x) \qquad (1)$$

If the initial data collection contains 1000 points, then plotting the function N would result in a surface of 1,000,000 points. We decided to reduce the analysis to those points for which either Δx or Δy is less than $\varepsilon = 0.1$ and we sorted them by Δx. Figure 7 shows the function N plotted this way for the data collection Alpine2.

For a better understanding of this image, we plotted a second figure zooming in on the first part by looking at noise points for which $\Delta x <= 0.5$. Figure 8 shows this second plot.

For the ETrack5 and Forza data, the right side part of the plot is very similar to Alpine2 in Figure 7. Thus, we show here only the zoomed-in plots where $\Delta x <= 0.5$. Figure 9 shows the noise function for the ETrack5 file, while Figure 10 shows the plot for Forza.

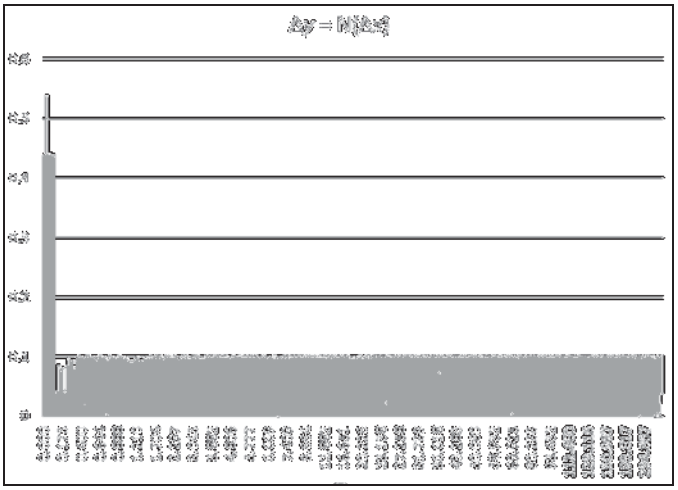


Fig. 7. Noise plot for Alpine2

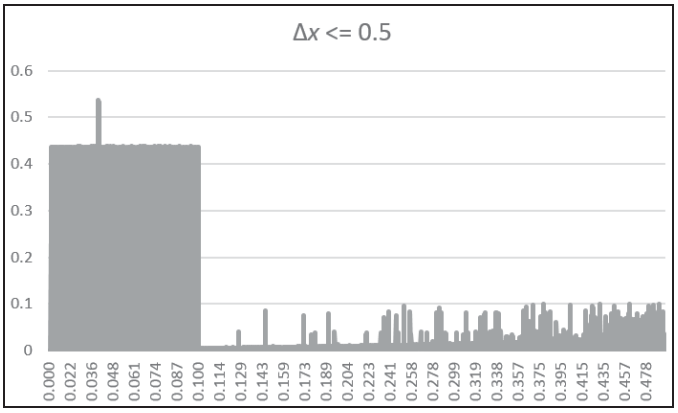


Fig. 8. Noise plot for Alpine2 with $\Delta x <= 0.5$

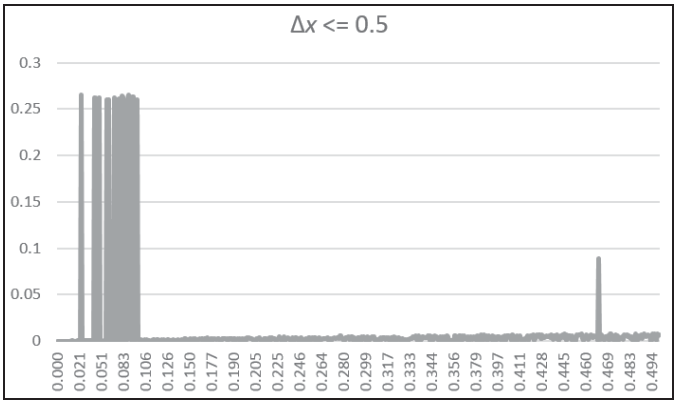


Fig. 9. Noise plot for ETrack5 with $\Delta x <= 0.5$

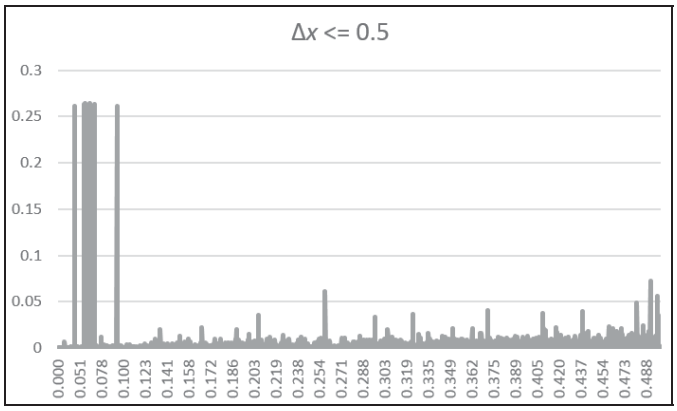


Fig. 10. Noise plot for Forza with $\Delta x <= 0.5$

5.2 Noise Analysis

Looking at Figure 7, we can distinguish two major artifacts, a visible spike on the left-hand side and a long plateau on the right-hand side. Figure 8 zooms in on the first part of the graph to show more details of this spike.

The interpretation of the spike is that we have input data points with a distance less than 0.1 for which the difference in the output is between 0.4 and 0.5. We can even spot some points closer than 0.02 with this kind of difference in the target output value. This spike will make it difficult for the NN to provide an accurate output value for all of these points, and will increase the value of the observed error.

Examining possible sources of this kind of spike, it is conceivable that the data are issued from a noisy source to begin with. If the data were collected from a human driving a car, various factors could have caused him or her to make different steering decisions in similar situations. The level of stress, fatigue, or distraction can be of influence and can cause slightly different reactions. Even when the data are collected from a procedural driver, the variety of algorithms used to address situations on the road can also cause the observed differences.

However, the spike can also mean that the data collection might have been incomplete. It is possible that external parameters that were not observed during the data collection also have an influence on the outcome and should be taken into consideration. If at all possible, the researcher can go back to the input parameters definition to see if more variables can be added. For this study, this is a possible future research track.

In our case, the noise analysis shows that the largest amount of noise, represented by the left-size spike in the three images, is present in the Alpine2 track, due to its complex nature. Looking back at the training error, this track resulted in the largest training error values of the three, between 0.13 and 0.15 in 1000 iterations. The second track in terms of noise is ETrack5, which also happens to be the median track in terms of training error, between 0.08 and 0.11. The track data collection showing the least amount of noise, Forza, is also the one with the smallest value of training error, between 0.04 and 0.06. However, in terms of test error, the best results were shown by the Alpine2 track, probably also due to its complex nature. From this analysis it seems pretty clear that there is a correlation between the amount of noise in the data, such as defined by the function N, and the average error of the trained NN. Thus, noise is a useful tool for understanding the capacity of learning of this method.

Let us examine the second artifact, the long plateau on the right side of Figure 7 for Alpine2. The same plateau is observed in the full plot for the two other tracks. This plateau means that we have data points that are increasingly distant - close to 300 for Alpine 2 - for which the difference in the target output value is less than or equal to 0.1. This occurrence in itself does not constitute a problem for the NN and is not likely to contribute to the average error. However, it can also indicate some improvements that can be made to the data collection. It could be caused by the nature of the function we are trying to learn, such as a periodical function, for example. However, another possibility is that some of the input variables chosen to feed the NN are redundant or do not contribute much to the output value. The researcher can examine the set of variables to see if some reduction of them is possible. Methods such as the ones proposed in [7] can be used for this purpose.

In our case, none of our variables are redundant. It is more likely that extra variables could provide more precision for the NN. However, some of our procedural driving algorithms have a Fuzzy Logic nature. For example, one of the rules says that if the free distance ahead is larger than a given threshold, we can consider the road to be almost straight and keep driving in the same direction. This means that in terms of this input variable, increasing its value above the threshold will not cause an observable difference in the output. This can be a possible cause for the observed plateau.

6 Conclusions

In this paper we presented an application of neural networks to a problem of autonomous car driver, in particular, to determining the steering angle. We used data collected on three tracks to train the NN and data collected from a fourth track for testing. We introduced an idea of using intrinsic symmetry of the problem to improve the NN training. We also proposed a method for visualizing noise issues in the data that can help with better understanding the results.

The experimental results presented in Section 4 show that using the problem's symmetry is beneficial to the training and testing results. The best configuration of the NN for our problem seems to be that of 10 neurons on the first hidden layer and 5 on the second. Even though the training error is generally lower for a simpler track such as Forza, the best error on the training data was obtained by using data from the Alpine2 track, which is more complex and challenging. Our

experiments also show that larger NNs do not always lead to better performance, although training them longer can improve their performance more than for smaller ones.

Section 5 introduced a noise measurement function and a method for visualizing the relevant part of it to study the noise present in the data. We have shown that data collections that present less noise are more likely to lead to lower average error for the NN. We also discussed how the analysis of the noise function can indicate better ways to collect the data for a problem. Thus, the noise analysis is an informative method for a learning algorithm.

7 References

[1] K. Albelihi, D. Vrajitoru, "An Application of Neural Networks to an Autonomous Car Driver", *The 17th International Conference on Artificial Intelligence*, July 27-30, Las Vegas, 716-722, 2015.

[2] K. Albelihi, "*The Gazelle Adaptive Racing Car Pilot*," Master's Thesis, Indiana University South Bend, 2014.

[3] E. T. Jaynes, "Gibbs vs Boltzmann entropies", *American Journal of Physics*, 33, 391-8, 1965.

[4] 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, 30–35.

[5] M. Trompf, "Building Blocks for a Neural Noise Reduction Network for Robust Speech Recognition", *Proceedings of EUSIPCA 1992*, Brussels, Belgium, Aug. 24-27, 1992.

[6] N. Friedman , M. Linial , and I. Nachman, "Using Bayesian networks to analyze expression data", *Journal of Computational Biology*, Vol. 7, 601-620, 2000.

[7] A. Hyvärinen, Survey on Independent Component Analysis, Neural Computing Surveys, Vol. 2, 94-128, 2001.

[8] K. Perlin, Ken, "An Image Synthesizer", *Proceedings of SIGGRAPH Computer Graphics*, 19 (0097-8930): 287-296, July 1985.

[9] W. Satney, A. Carrington, T. Scantlebury-Manning, and A. Als, "Determining Signal Source Integrity Using a Semi-supervised Pattern Classification System", *The 17th International Conference on Artificial Intelligence*, July 27-30, Las Vegas, 629-635, 2015.

[10] M. Olano, "Modified Noise for Evaluation on Graphics Hardware", *ACM SIGGRAPH/Eurographics Graphics Hardware*, 2006.

[11] B. Li and K. Chai Sim, "A spectral masking approach to noise-robust speech recognition using deep neural networks", *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, Volume 22 Issue 8, August 2014.

[12] N. Khademi Kalantari, S. Bako, and P. Sen, "A machine learning approach for filtering Monte Carlo noise", *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2015*, Volume 34 Issue 4, August 2015.

Autoregressive Models May Loose its Global Optimization in Recursive Multistep Ahead Forecasting

[*]Hugo Siqueira[1], Ivette Luna[2], Maurício Kaster[1] and Christiano Lyra[3]
[1]Department of Electronic Engineering, Federal University of Technology - Paraná, Ponta Grossa-PR, Brazil
[2]Institute of Economics, University of Campinas, Campinas-SP, Brazil
[3]School of Electrical and Computer Engineering, University of Campinas, Campinas-SP, Brazil

Abstract - *Among the methods applicable to time series forecasting the autoregressive models stand out. This is explained by their easy mathematical tractability allied with good results. The adjustment of the free parameters is done by an analytic solution - Yule-Walker equations -, which allows to find the global optimum of their cost function. However, the prediction of up to one step ahead in a recursive way violates the global optimization, once suboptimum points may achieve better performances. This work performs the streamflow series forecasting from hydroelectric plants, for a twelve steps ahead horizon. The calculation of the coefficients is done by the artificial immune systems and LMS algorithm. The parameters found were different from those achieved by the Yule-Walker equations, which means that the optimization processes converge to distinct solutions. The computational results indicate that other optimization methodologies should be considered to increase the performance of autoregressive models in multistep prediction.*

Keywords: Autoregressive models, artificial immune systems, LMS algorithm, seasonal streamflow series, forecasting.

1 Introduction

Probably, the most widespread linear forecasting approach from the Box & Jenkins methodology are autoregressive (AR) models, which can be seen as finite impulse response (FIR) predictors, i.e. models that do not present feedback loops. This feature allows that, under a mean-squared error (MSE)-based formulation, the optimal model parameters for one step ahead (P=1) prediction be obtained in closed form according to the Yule-Walker equations [1].

Multistep ahead forecasting is another important task, being the research focus of many forecasting works [2-3]. One of the most used methodologies to carry out the multistep prediction is the recursive approach, where the prediction is done in a way that the intermediate samples missed to obtain the response for the step P are predicted and utilized as inputs of a one-step ahead adjusted predictor. However, the forecasting capability of the models is reduced when the horizon is increased, even when there is an adequate performance in one step [4]. This behavior can be explained by the listed reasons:

i) Simplifications derived from the linearity of the model, which may not take into account factors that are determinant to the series;

ii) The insertion of previous forecasts in the input of the predictor, to form new responses for P steps ahead;

iii) When *ii)* occurs, the guarantee of optimality derived from the Yule-Walker equations ceases to exist, as they do not take into account the existence of feedback to form new forecasts, once the previous responses already have errors;

iv) As is usual in time-series studies, once the forecasting horizon is increased, the correlation between the input and output samples decreases.

As consequence of the aforementioned facts, improvements in the AR performance may be achieved with a suboptimal adjustment of the model, in the mean square error sense. The main difficult, in this case, is that the analysis of the error performance can just be done after the completion of the prediction task. The immediate consequence is counterintuitive, because better results may be found even in a worse optimization process.

To deal with this assumption, this work selected the monthly seasonal streamflow series, which are very important in countries as Brazil, where the electric power generation is predominantly carried out by hydroelectric plants. In this view, the development of efficient models to predict these flows is essential.

Monthly seasonal streamflow series prediction is a key element in the planning and mensal programming of the use of hydric and energy resources of the National Interconnected

[*] Corresponding Author

System. To cope with these tasks, the Electric System National Operator (ONS) has a group of models that employ daily, weekly and monthly streamflow series [5]. In the Monthly Operation Program coordinated by ONS, a framework widely adopted to handle the forecasting is based on time series models from the AR approach [7].

However, in respect to the nature of the energy planning problem and the need for medium term programming, the prediction of streamflow series for the wider horizon of twelve steps ahead (P=12) has to be considered. For instance, in [4] it is shown that an increase in the prediction horizon leads to a degradation in the performance of periodic AR models for streamflow time series, as expected. This behavior implies in a decrease in the potential of analyzing one year ahead the generation potential of several plants of the SIN.

A model with a good predictive power will be advantageous in terms of performance with respect to the prediction error in comparison with the long term mean (LTM), which is the upper limit of the MSE of an AR model. The LMT is the average of the streamflows of each month (January, February, etc.), estimated with all the available data.

In this sense, even small improvements in the MSE for multi-steps ahead are difficult to achieve. The standard mean unconditional variance of the series reflects the maximum expected variability corresponding to the MSE reached if the LMT were used as the prediction procedure. Hence, the standard MSE was utilized as a reference to evaluate the gains derived from the application of alternative models.

In this context, this work presents an investigation that explores a bio-inspired optimization technique to optimize the AR models: Artificial Immune Systems (IAs) [7]. This algorithm has been successfully employed in works related to streamflow series forecasting and has a *modus operandi* that presents a good balance between local and global search. The use of these algorithms in temporal signal processing tasks with similar features has led to very interesting results.

Besides, the least mean square algorithm (LSM), a robust adaptive filtering approach, which is widely applied in the context of temporal processing, was employed [1]. This method utilizes a stochastic approximation of the gradient vector, allows search estimates of the global optimum solution, with a low computational cost. It is interesting that, in the evaluation of the algorithm, their response behavior stands close of the solution, but rarely stops in the optimum point.

This work is organized as follows: Section 2 discusses the autoregressive models and Yule-Walker equations; Section 3 the LMS algorithm and the artificial immune systems, while Section 4 presents the computational results and some relevant discussions; finally, Section 5 shows the conclusions and possible future works.

2 Autoregressive Models

The prediction of future values for a time series is done, in essence, by properly combining its past values. In order to do so, a prediction model must be defined and its parameters have to be estimated based on a predetermined criterion – typically the mean-squared error (MSE) [1].

The autoregressive (AR) model is probably the most popular choice for dealing with the forecasting of stationary time series. Its mathematical expression for a p-order model, denoted as AR(p), is shown in Equation (1):

$$\hat{x}_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + ...\phi_p x_{t-p} + e_t \qquad (1)$$

where $x_{t-i}, i = 1,..,p$ are the delays of the observed series, $\hat{x}_t$ is the predicted value in time t, ϕ_i, $i=1,...,p$, are the free parameters and e_t is a not modeled random component. This model can be seen as an FIR (*finite impulse response*) predictor, which means that there is no structural feedback. This feature guarantees the existence of a closed-form solution to the problem of estimating its coefficients according to the MSE (or Wiener) criterion, which leads to the following cost function [1]:

$$\mathbf{J}_{\mathbf{w}} = E[e_t^{\,2}] \qquad (2)$$

where $\mathbf{J}_{\mathrm{w}}$ is the cost function to be minimized, $E[.]$ is the mathematical expectation operator and e_t the instantaneous sample error (which is equivalent to the unmodeled term in (1)). The difference between the desired signal d_t (which in the case described in (1) is x_t) and the predictor output $\hat{x}_t$ is

$$e_t = d_t - \hat{x}_t \qquad (3)$$

The Yule-Walker equations are derived from the standard null gradient condition, which leads to the global (and single) optimum of the MSE [8]:

$$\mathbf{\Phi} = \mathbf{R}^{-1}\mathbf{r} \qquad (4)$$

where

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1 & \dots & r_{p-1} \\ r_1 & r_0 & \dots & r_{p-2} \\ \dots & \dots & & \dots \\ r_{p-1} & r_{p-2} & \dots & r_0 \end{bmatrix}$$

is the autocorrelation matrix of the model inputs and

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \\ \dots \\ r_p \end{bmatrix}$$

is the vector containing the cross correlation between the inputs and the desired signal, which amounts to a set of values taken from the autocorrelation profile of the time series at hand. The vector $\mathbf{\Phi} = [\phi_1, \phi_2, ..., \phi_p]$ is the optimal solution to

the model presented in Equation (1). Observe that the cost function is unimodal and forms a hyperparaboloid [1,6,8].

However, time series prediction may be related to building estimates not only of the next sample, but of a sample P steps ahead, defining a more challenging forecasting horizon. The most usual strategy to solve this problem is to use a predictor adapted to minimize the MSE with $P = 1$ in a recursive process where, for each subsequent step, the input pattern of the models is built using the past predictions as the unknown observations [2]. However, in this case, the strict FIR character of the prediction model is compromised by the existence of information feedback, which questions the optimality of the design approach based on the Yule-Walker equations, which can be seen as engendering an "open-loop solution".

In the riverflow context, several studies have indicated the significant difficulty to estimate future values of monthly streamflows for a twelve-step ($P = 12$) horizon [2-3]. In general, what occurs is that, independently of the nature of the prediction model – linear or non-linear –, the response for P=12 tends to the long term mean (LMT), being the information gains small in terms of MSE. In this case, considering the tradeoff between complexity and performance, the choice is to use the LMT as the estimative of future values in several occasions.

Due to this limitation, this study aims at evaluating the impact of the adoption of a natural computing framework that deals directly with the existence of information feedback in the AR model for $P = 12$ with respect to the performance of canonical "open-loop" strategies. Aside from the direct use of Yule-Walker equations, strategies of the latter type can be interactively built using adaptive methods like the LMS algorithm, which will now be described.

It is expected that the IA and LMS converge to points in the neighborhood of the global optimum calculated via Yule-Walker equations to P=1.

3 Optimization Strategies

3.1 LMS Algorithm

The calculation of the coefficients of a linear prediction model can be interactively done by algorithms that, in essence, attempt to estimate the solution provided by the Yule-Walker equations. Among these methods, the least mean square (LMS) algorithm [9] can be highlighted as a standard solution that allies robustness to a remarkably simple operation, a consequence of the use of an stochastic approximation to the statistical averages present in the gradient of the cost function $\mathbf{J}_{\mathbf{w}}$ (Haykin, 1996).

Considering a steepest-descent method as the basis for interactive optimization, it is possible to write an update expression of the kind [9]:

$$\mathbf{\Phi}_{t+1} = \mathbf{\Phi}_t - \gamma \nabla \mathbf{J}_{\phi} \tag{5}$$

Thereby, by the aformentioned stochastic approximation, equation (6) becomes the LMS update expression:

$$\mathbf{\Phi}_{t+1} = \mathbf{\Phi}_t + \gamma e_t \mathbf{x}_t \tag{6}$$

where γ is the algorithm step-size and x_t is a vector containing the inputs.

3.2 Artificial Immune Systems

Artificial immune systems (IAs) are population-based metaheuristics whose operation is inspired in the defense mechanism of superior organisms against antigens. These methods find significant application in optimization tasks [10].

Hence, population-based algorithms become attractive candidates due to their straightforward operation, local search capability and global search potential [7]. In previous works related to other streamflow prediction tasks, as in [11], IAs presented expressive results even in comparison with other metaheuristics, as genetic algorithms [12].

This work employs a version of the CLONALG algorithm [13] with real coding. The recognition process of an antigen is the key phenomenon that underlies the computer-based representation that characterizes the CLONALG. The solution of the addressed optimization problem is a vector with real numbers that represents the antibody structure, whereas the cost function – or fitness measure – quantifies the degree of affinity between antibody and antigen.

The process of antibody adaptation is understood in terms of the clonal selection principle, which, in very simple terms, states that in the process of antigen recognition, the defense cells of the organism engender copies of themselves, i.e. clones. These clones are subject to spurious modifications (mutations) that are proportional to the degree of affinity between them and the antigen in question [10]. Therefore, from a practical standpoint, the algorithm is characterized by the use of a real-valued mutation operator over a number of clones of each antibody, which is followed by a selection process that preserves, from each set parent-offspring, the best individual. Aside from that, there is a periodic insertion of new randomly-generated antibodies, which is important to ensure population diversity and a more pronounced global search potential.

Some additional definitions are necessary to clarify the application of IAs to the focused problem. The first is the definition of the fitness function, which, in this work, is done according to (7) [13]:

$$\mathbf{J}_{\mathbf{fit}} = \frac{1}{(1 + \hat{\mathbf{J}}_{\mathbf{w}})} \tag{7}$$

where $\hat{\mathbf{J}}_{\mathbf{w}}$ is an estimate of the MSE cost function based on a time average, in analogy to Equation (2):

$$\hat{\mathbf{J}}_{\mathbf{w}} = \frac{1}{N} \sum_{t=1}^{N} (x_t - \hat{x}_t)^2 \tag{8}$$

being N the number of observations. However, to forecast with more than a single step ahead, it is possible to rewrite Equation (8), considering the variable P:

$$\hat{\mathbf{J}}_{\mathbf{w}} = \frac{1}{N}\sum_{t=1}^{N}(x_{t+P} - \hat{x}_{t+P})^2 \quad (9)$$

In the proposal described in (9), during the search process, differently from the other methodologies, the IA will select the coefficients that yield the best MSE P steps ahead. This proposal, as already stated, aims to reduce the effects produced by prediction-error propagation when the model outputs are fed back to the system input.

Algorithm 1 shows the structure of the employed version of CLONALG [11]:

Algorithm 1. CLONALG Algorithm Used

Initialization
- Choose the parameters of the algorithm and randomly initialize the antibodies of the population.
Interactive process
- While the maximum number of iterations or generations is not reached, do:
1. Calculate the fitness of all antibodies;
2. Produce Nc copies of each antibodies;
3. Apply a mutation process to all copies (clones), maintaining the original individual unchanged. The mutation is proportional to the fitness and follow these two equations:

$$c' = c + \alpha N(0,1)$$
$$\alpha = (1/\beta)exp(-f),$$

where β is a regulation parameter of the mutation amplitude and f is the value of the fitness of the clone c.
4. Determine the fitness of the new antibodies and save of each group just the best solution.
5. Each N_{it} iterations include N_{ind} solutions generated randomly and substitute for the N_{ind} individuals with lower fitness.

In steps 2-4, there is an efficient local search mechanism (directly inspired in the clonal selection principle), which, nonetheless, in view of the stochastic character of the mutation operator, can also potentially lead to the escape from suboptimal solutions. Step 5, on the other hand, accounts for the possibility of exploring very different regions of the search space, which is crucial to perform global search.

4 Case Study

4.1 Analysis of Seasonal Streamflow Series and Preprocessing

The monthly streamflow series of the hydrologic plants considered in this work are composed of three distinct and well-defined periods along their history: dry, humid and normal. In addition to that, a seasonal pattern exists with a twelve-month periodicity. The months related to the humid period are those with the largest means and variances, being therefore the most complex periods to model [3].

The case study considers three hydroelectric plants with distinct hydrological behaviors: Furnas, Sobradinho and Emborcação. The differences in the streamflow characteristics are verified by the observed means and standard deviations, as shown in Table 1:

Table 1. Mean and Standard Deviation of selected series

Series	Mean ($\hat{\mu}$)	Standard Deviation ($\hat{\sigma}$)
FURNAS	926.6177	613.1671
SOBRADINHO	2666.0229	1959.0325
EMBORCAÇÃO	486.0781	362.8067

The streamflow series are composed by observations starting in January 1931 to December 2010, available at the Electric System National Operator (ONS) website [5]. The selected test periods were: Furnas from 1967 to 1976; Furnas from 1977 to 1986; Sobradinho from 2001 to 2010 and Emborcação from 1967 to 1976. Each set contains 120 monthly samples.

It is possible to observe in Table 1 that Sobradinho plant has a historic mean significantly higher than Furnas plant, and those has means approximately twice the mean of Emborcação.

Table 2 presents the mean and standard deviations of the selected test periods. It is noticeable that the period 77/86 corresponds to an atypical moment in the affluent water volume, well above the historic mean of Furnas (around 26%). On the other hand, from 1967 to 1976, the streamflow series registered were around 90% of the historic mean associated to Furnas and Emborcação. Between 2001 and 2010, the mean registered in Sobradinho was around 80% of the historic mean.

Table 2. Mean and Standard Deviation of test sets

Period	Mean	Standard Deviation
FURNAS 67/76	830.5083	502.6076
FURNAS 77/86	1167.841	742.0656
SOBRADINHO 01/10	2154.425	1587.8901
EMBORCAÇÃO 67/76	431.8083	304.2881

These four hydrological scenarios permit a consistent evaluation of the model performance. As usual, the series are divided into two sets: training, in which the coefficients are estimated and tested, in order to verify the performance of the models. The training sets are formed by the remaining observations.

Because of the tacit assumption regarding the stationarity of the time series to be predicted in the context of the classical AR model formulation, it is necessary to preprocess the seasonal streamflow series in order to remove their seasonal component. This process is carried out in accordance to Equation (10) [3]:

$$z_{i,m} = \frac{x_{i,m} - \hat{\mu}_m}{\hat{\sigma}_m} \quad (10)$$

where $x_{i,m}$ represents the streamflow observed in the month m of the i-th year, $i = 1,2,...,N_y$, in the month $m = 1,2,...,12$, being N_y the number of the years avaliable in the historical data; $\hat{\mu}_m$ is the historical mean (or long term mean) observed to the month m, and $\hat{\sigma}_m$ the respective standard deviation. As a consequence, the original series is transformed into a new stationary series z_t, with zero mean and unit standard deviation.

The prediction models are applied to the series z_t and after this process, the seasonal component is reinserted, reverting the deseasonalization to allow the general analisys of the forecasting process and the calculation of the respective errors. Another investigation can be based on the error associated with the deseasonalized series, since it is in this domain that the AR model is adjusted. This is important because, as discussed in [3], the reversion of the deseasonalization process (or the reinsertion of the seasonal component) may cause certain distortions related to the existence of different standard deviations in Equation (10).

The choice of the order of each model was carried out with the aid of the partial autocorrelation function [1,6,8]. From this analysis, the selected AR orders were: Furnas – AR(3); Sobradinho – AR(4); Emborcação – AR(4).

4.2 Computational Results

After the selection of the orders, three methods were used to calculate the free parameters of the AR model: Yule-Walker equations, iterative estimation via LMS algorithm and the "closed loop" immune-inspired framework. It must be noted that two distinct optimization processes were performed with the IA: the first with the cost function presented in Equation (8) and another with Equation (9). Hence, in fact, four optimization processes were carried out.

The Friedman's test [14] was applied to verify if the results were significantly different, in the one step ahead forecasting, as the parameter optimization was done for this horizon. So, the AR model adjusted by Equation (9) was out of this test, since this model adjusts the parameters based on the minimization of the MSE for $P = 12$.

The analyzed error metrics were the Mean Absolute Error (MAE), the Mean Absolute Perceptual Error and the Root Mean Squared Error (RMSE). These metrics are presented in the following equations:

$$MAE = \frac{1}{N_s}\sum_{t=1}^{Ns}\left|d_t - \hat{x}_t\right| \quad (11)$$

$$MAPE = 100 * \frac{1}{N_s}\sum_{1}^{N_s}\left|\frac{d_t - \hat{x}_t}{d_t}\right| \quad (12)$$

$$RMSE = \sqrt{\frac{1}{N_s}\sum_{t=1}^{N_s}(d_t - \hat{x}_t)^2} \quad (13)$$

where d_t is the observed streamflow, $\hat{x}_t$ is the predicted streamflow and N_s is the number of observations out of the samples.

Table 3 presents the results by means of 30 independent simulations for each optimization algorithm, in the real and deseasonalized spaces, for the four test periods selected. The following labels are used to identify the series: "Furnas1" are Furnas 67/76, "Furnas2" are Furnas 77/87, "Sobrad" refers to Sobradinho 01/10 and "Emborc" is Emborcação 67/76. The labels RMSEd and MAEd are related to the results in the deseasonalized domain. "Opt" is related to the optimization method used and "RMSEd" and "MAEd" are the metric performances in the deseasonalized domain.

Table 3. Forecasting perfornamces

	Opt.	MAPE	RMSE	MAE	RMSEd	MAEd
Furnas1	IA 12	30.1699	351.905	244.501	0.9577	0.7276
	IA 1	29.3160	349.421	239.808	0.9438	0.7142
	LMS	**29.0054**	**346.920**	**237.679**	**0.9356**	**0.7067**
	Y-W	29.1397	348.113	238.205	0.9390	0.7090
Furnas2	IA 12	30.7657	488.984	348.834	1.4936	1.0334
	IA 1	30.5413	**486.830**	344.230	1.4858	1.0170
	LMS	30.4094	487.765	**343.692**	1.4856	**1.0117**
	Y-W	**30.4082**	487.802	343.978	**1.4855**	1.0126
Sobrad	IA 12	**26.7103**	970.963	**622.972**	0.7257	**0.5704**
	IA 1	27.7785	**940.656**	626.272	**0.7116**	0.5833
	LMS	28.1921	942.644	630.625	0.7143	0.5872
	Y-W	28.1994	943.091	630.865	0.7149	0.5875
Emborc	IA 12	33.9226	226.987	154.266	1.0042	0.7960
	IA 1	**33.8263**	**220.349**	**151.455**	**0.9973**	**0.7836**
	LMS	34.1777	220.557	152.075	0.9997	0.7873
	Y-W	34.1807	220.486	152.059	0.9996	0.7873

The *p-values* achieved from the Friedman's test for each period were less than 0.07. As it is possible to verify, with 10% of significance, the small values found reject the null hypothesis of no significant difference between the predicted values provided by all three algorithms.

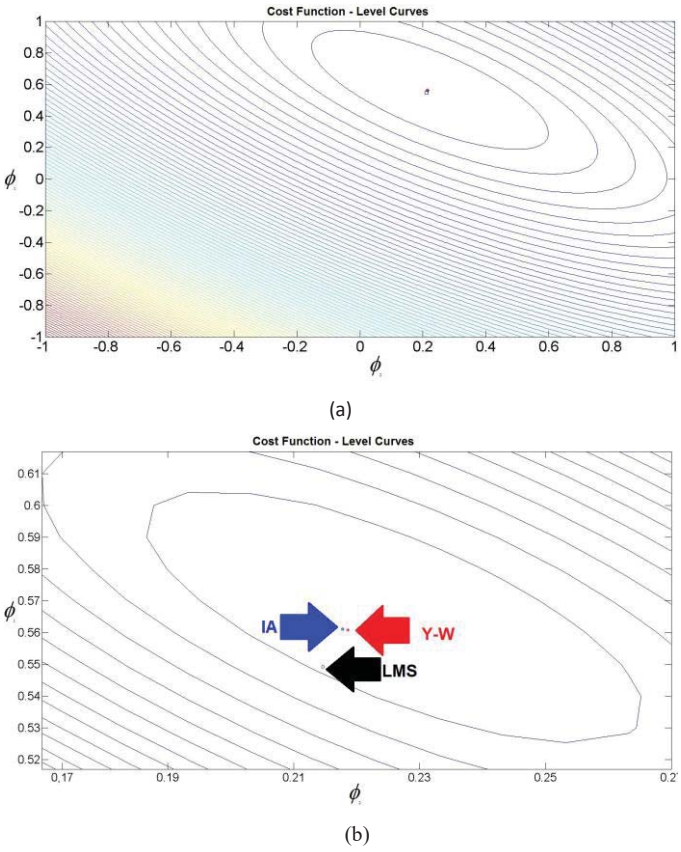
Some important conclusions can be drawn from the obtained computational results. The first is that, for the Furnas 67/76 and Sobradinho series, there is not a consensus with respect to the error metrics, the best results being different with respect to algorithms and metrics. Another fact is that, in Furnas 77/86, the best RMSE in the deseasonalized domain is generated by a method that is not the best method in the real domain. This possibility is not a surprise, as discussed in the previous section: periods with high standard deviation tend to have a greater weight during deseasonalization, which is not taken into account in the prediction domain.

The most usual methodology is to consider the RMSE in real domain as the metric to be adopted to indicate the best performance [1,3]. The results achieved for the selected test scenarios were in three of four sets favorable to the AR model trained using the IA and adjusted to $P = 1$ (see Equation (9)), while the LMS was the best in the other case. This behavior corroborates the initial assumption: Yule-Walker equations can find the global optimum to one step ahead, but to P=12, this point may not be the best option. This is a strong

indicative that the other algorithms are potentially useful to enhance the achievable performance in the context of the considered prediction problem.

Furthermore, the use of the cost function described in Equation (8) was not fruitful because of the high value of error found using all metrics. Despite the same immune algorithm was used to optimize the cost function from Equation (9), this change did not provide any benefits.

Finally, to exemplify the general behavior of the optimization procedure, an AR(2) adjustment to Furnas 67/76 was performed. The parameters found by each optimization technique were Φ=[0.5608, 0.2154] to Yule-Walker approach, Φ=[0.5611, 0.2146] to the IA and Φ=[0.5492, 0.2114] to the LMS algorithm. In Figure 1, which presents the level curves of the cost function in a 2-dimentional perspective, it is possible to observe that the parameters are close to each other, considering the accepted range in which the coefficients are in the interval [-1;+1]. In (a) it is possible to verify the convergence to close points, but in (b) the small difference achieved by each algorithm is shown.



(a)

(b)

Fig. 1. Level Curves of the AR(2) cost function: (a) coefficientes in the interval [-1;+1] and (b) a zoom perspective highlighting the optimum points found

As initiallly mentioned, the presented behavior indicates that, in multistep ahead predict horizons tasks, it is necessary to analyze other optimization strategies besides the Yule-Walker formulation.

5 Conclusions

This work presented an investigation about optimization of linear autoregressive models (AR). The widely known Yule-Walker equations guarantee the global optimization of the parameters of an AR model adjusted to one step ahead horizon. This methodology presents a simple mathematical tractability with analytic solution. However, it was found that in multi-step ahead prediction using the recursive approach, this characteristic is lost.

To deal with this assumption, the monthly seasonal streamflow series forecasting of hydroelectric plants with the horizon of twelve steps ahead have been addressed. This is an important problem once the energetic planning in a country like Brazil, where the most part of the electric power is done by hydroelectric plants, is based on these predictions.

Thus, the bio-inspired artificial immune system and the Least Mean Square algorithm (LSM) were performed to calculate the free parameters of the AR model. In the mean square error sense, these methods found suboptimal solutions to the adjustment of the parameters to one-step ahead.

However, the computational results were consistent to show that alternatives to the classical Yule-Walker equations have to be considered in an optimization process for prediction, once, in most cases, the immune algorithm and LMS presented the best results. The main indicative is that, despite the global optimization of the AR model achieved to one step ahead, in multistep tasks other methodologies must be performed, once the global optimum may shift its position.

The continuity of this work can be done by means of exploring other linear models, as the periodic autoregressive (PAR), or recursive models, like autoregressive and moving average (ARMA), once they can be used under the same egis of the application of immune algorithms, to a horizon of twelve steps ahead, with the possibility to have gains in the reduction of the error. Besides this, it is valid to test the models in another kind of series, like daily or weekly, important to the short-term operation.

Acknowledgment

This work was supported by grants from CAPES, Fundação Araucária and Secretaria de Estado da Ciência, Tecnologia e Ensino Superior do Paraná (SETI).

6 References

[1] Simon Haykin. "Adaptive Filter Theory". Prentice Hall, 1996.

[2] Anti Sorjamaa, A., Jin Hao, Nima Reyhani, Yongnan Ji, Amaury Lendasse. "Methodology for Long-Term

Prediction of Time Series." Neurocomputing, 2861–2869, October 2007.

[3] Hugo Siqueira, Levy Boccato, Romis Attux, Christiano Lyra Filho. "Unorganized Machines for Seasonal Streamflow Series Forecasting", International Journal of Neural Systems, 24, 1430009.1-1430009.16, May 2014.

[4] Marinho Andrade, Ricardo Luis dos Reis, Secundino Soares, Donato da Silva Filho. "Analysis of the Monthly Seasonal Streamflow Series Forecasting Error With Diferent Forecasting Horizons" SBA Controle & Automação, 23, 3, 294-305, June 2012. [In portuguese: Análise do erro de previsão de vazões mensais com diferentes horizontes de previsão"].

[5] ONS - Electric System National Operator - Brazil [online], 2015. Avaliable in: http://www.ons.org.br/operacao/vazoes_naturais.aspx

[6] George Box, Gwilym Jenkins, Gregory Reinsel. "Time Series Analysis: Forecasting and Control", 4th edn. Holden Day, 2008.

[7] Leandro Nunes de Castro. "Fundamentals of Natural Computing: Basic Concepts, Algorithms and Applications". Chapman & Hall, 2006.

[8] James Hamilton. "Time Series Analysis", 1st edn. Princeton University Press, 1994.

[9] John Shynk. **"**Adaptive IIR Filtering", IEEE Acoustic, Speech and Signal Processing Magazine, 6, 2, 4-21, April 1989.

[10] Leandro Nunes de Castro, Jonathan Timmis. "Artificial Immune Systems: a New Computational Intelligence Approach", 1st edn., Springer, 2002.

[11] Hugo Siqueira, Romis Attux, Christiano Lyra Filho. "Exploration of Linear Alternatives to Monthly Streamflow Series Forecasting", Mecánica Computacional, 29, 9629-9644, November 2010. [In Portuguese: "Exploração de Alternativas Lineares para Previsão de Séries de Vazões"].

[12] John Holland. "Adaptation in Natural and Artificial Systems". MIT Press, 1992.

[13] Leandro Nunes de Castro, Fernando José Von Zuben. "Learning and Optimization Using the Clonal Selection Principle", IEEE Transactions on Evolutionary Computation, 6, 3, 239-251, June 2002.

[14] Ivette Luna, Rosangela Ballin. "Top-Down Strategies Based on Adaptive Fuzzy Rule-Based Systems for Daily Time Series Forecasting". International Journal of Forecasting, , 1–17, July 2011.

Assessment Of The Admission Criteria That Predict Students' Academic Performance in Undergraduate Years in a Nigerian University

Okereke Eze Aru, Ifeyinwa E. Achumba and F.K. Opara

Department of Electrical/Electronic Engineering, Federal University of Technology-Owerri, Imo State-Nigeria

ABSTRACT:

Prior to this era, individual universities in Nigeria conducted concessional examinations for the selection of their prospective students. Other compulsory condition was the possession of five credit passes in the relevant subjects in the ordinary level certificate. The other mode of entry was the direct admission of students into the direct second year level of candidates who possessed either the national diploma certificate or the general certificate of education at the advanced level in the relevant papers. Problem arose, where students had multiple admissions in several universities and those who could not afford to write admission examinations to multiple institutions and the not too brilliant students were deprived the opportunity of admission into the very limited Universities. These reasons caused the Federal Government of Nigeria to establish the Joint admission and Matriculation Board (JAMB) in 1978. Today, studies showed that JAMB and WAEC alone can no more ascertain the best crop of students for admission as there are exam syndicates everywhere in the country to ensure that their candidates/customers make all their papers in one sitting consequently pass their JAMB and get admission as well. This research is therefore proffering an alternative admission paradigm that will go beyond considering only test scores for admission, but will consider aggregate academic records of the prospective students before admission is offered or denied. This alternative method can be actually using Artificial Neural Network techniques.

Keywords: Admission, Artificial Neural Network, Academic Performance, Prediction, Undergraduates, decision support system, etc.

I NTRODUCTION:

Irrespective of the end close strategies employed by the Nigeria Ministry of Education to ensure that educational standards are maintained at University level, students who after passing through these tedious examinations still perform poorer than expected. For instance, from the summary of Computer Engineering students' data, College of Engineering, Michael Okpara University of Agriculture, Umudike Umuahia, Abia State for 2009/2010 session, out of 87 students that were admitted into the programme, only 25 students were able to graduate within the NUC approved number of years for studying the course. None had CGPA above 4.5, 13 had CGPA between 3.5 and 4.49, 11 had CGPA between 2.4 and 3.49, 1 had CGPA between 1.5 and 2.39. At the end 58 students were recorded to have one or two carry-overs and 4 were asked to withdraw (Departmental second semester summary, 2014).

This implies that only 28.74% of the students actually had satisfactory results at the end of their stay of five academic years. This also shows that 71.26% of the students had academic challenges as undergraduate students. The high rate of poor academic achievement among undergraduate is not unconnected with the channel through which they gained entry into the University. Ebiri (2010), observed that using JAMB as a yardstick for admission of students into Nigerian universities has led to the intake of poor caliber of candidates, characterized by high failure rate, increase in examination malpractice, high spillovers and the production of poor quality output that are neither self-reliant nor able to contribute effectively in the employment world.

Ironically, the process of selecting candidates for admission into tertiary institutions has largely depended on some fixed combinations of some subjects taken by applicants in their lower level classes. However, this technique has

never been proved efficient in admitting candidates that may perform well in the chosen courses. Considering the fast growing number of candidates seeking for admission into tertiary institutions, there is a need to use past data for decision support in admitting suitable candidate for a course of study.
Universities are facing the immense and quick growth of the volume of educational data (Schönbrunn and Hilbert, 2006). Intuitively, this large amount of raw stored data contains valuable hidden knowledge, which could be used to improve the decision making process of universities (keshavamurthy et al., 2010). An analysis of the existing transaction data provides the information on students that will allow the definition of the key processes that have to be adapted in order to enhance the efficiency of studying (Mario et al., 2010). This study delves into the problem of finding data patterns in admission datasets and provides a technique to predict the performance of students in the first year in the University based on the admission combination.

2. PROBLEM STATEMENT:

The traditional admission decision paradigm in Higher Academic Institutions of Learning in Nigeria is based solely on UME and Post-UME scores, given that the student has credited the required five subjects in WAEC, NECO, and/or GCE. This traditional admission decision process may not yield the best crop of students in terms of academic performance as it is vulnerable to cheating from undeserving students who can afford to cheat during the WAEC/NECO/GCE and UME/Post-UME exams and tests, respectively. This is especially the case in this era of Organized UME/WAEC/NECO/GCE Syndicates that promise and "ensure" that their customers pass their UTME test and credit all their WAEC/NECO/GCE subjects in one sitting. Also, the freshman admission decision process in Nigeria is complicated by ethnicity, sectionalism, favoritism, influence, catchment area and quota system policies.

3. RESEARCH OBJECTIVES:

The objectives of this study include, but not limited to the following:
1) establishing some relevant factors that affect a undergraduate students' academic performance,
2) to transform these factors into forms suitable for an adaptive system coding, and
3) to model an Artificial neural network that can be used to predict a candidate's performance based some historical information and aggregate academic records.

4. A BRIEF REVIEW OF STUDENT ACADEMIC PERFORMANCE

According to Literature, in 1954, the University of New Zealand Council for Educational Research investigated the relationship between academic standards of students on entrance and their first year university work. The study found that the median correlation found among the many sets of variables representing general school performance and general university performance was indicated by a *tau* coefficient of 0.36 for the first year students undertaking their studies on a full time basis (Maidment ,1968). In 1975, Bakare summarized the factors and variables affecting students performance into the intellective and non-intellective factors, emphasizing that the intellectual abilities were the best measure (Bakare 1975). He categorized causes of poor academic performance into four major classes as follows:
1) Causes resident in society, where the society is responsible for the poor performance
2) Causes resident in school, where the fate of the students seeking for undergraduate admission is determined by what their previous school equipped them with.
3) Causes resident in the family, where the family is to be blamed for failure due to their failure to shape the student well on time
4) Causes resident in the student, where the student himself fails to work hard.

Studies such as (Lage and Tregelia, 1996) and (Dynan, 1977) looked at a more general aspects of success while Anderson et al., 1994 studied the effect of factors such as gender, student age, and students' high school scores in mathematics, English, and economics, on the level of university attainment. According to their study, students who received better scores in high school also performed better in university. Another aspect discovered was that men had better grades than women and choose to drop from school less often. Adedeji (2001) sought to find out a orrelation between students matriculation exam (UME) scores and their academic performance in Nigerian

universities, using the Faculty of Technology, University of Ibadan, Nigeria as a test case. He investigated the relationship between students' UME scores, first, second, and final year Grade Points (GP) with the use of a simple correlation and regression analysis. He concluded in his research that there exists a positive relationship between students admission scores and their undergraduate performance. However, recent trends after Adedeji's study indicates the unreliability of the WAEC and UTME/PUTME scores.

5. ARTIFICIAL NEURAL NETWORK

Artificial Neural Networks are electronic models based on the neural structure of the brain. The brain learns from experience. The most basic element of the human brain is a cell which, unlike the rest of the body, doesn't appear to regenerate. These cells are known as neurons. There are 100 billion of them these cells in the brain and each can connect with up to 200,000 other neurons. There are multiple connections between them.

The individual neurons are complicated. They have a myriad of parts, sub-systems, and control mechanisms. They convey information via a host of electrochemical pathways. Together these neurons and their connections form a process which is not binary, not stable, and not synchronous. The artificial neural networks try to replicate only the most basic elements of this complicated, versatile, and powerful organism. But for the software engineer who is trying to solve problems, neural computing was never about replicating human brains. It is about machines and a new way to solve problems.

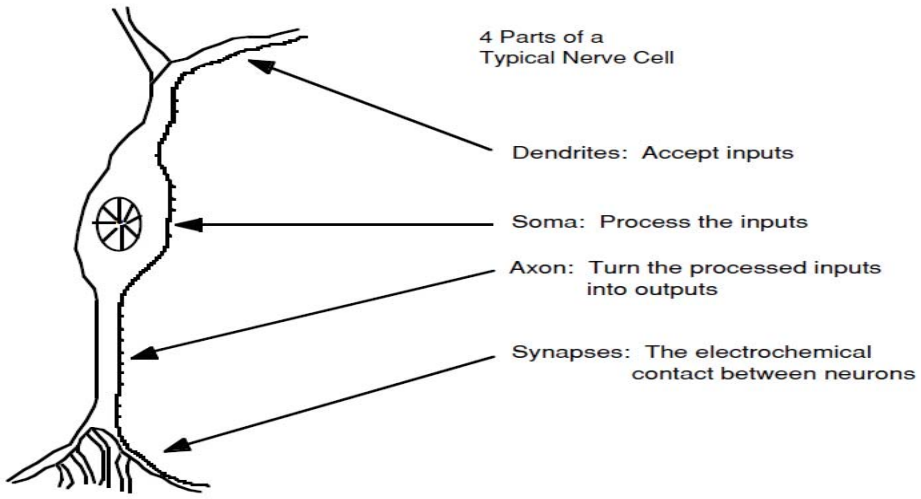


Fig 1: Model of a Biological Neuron

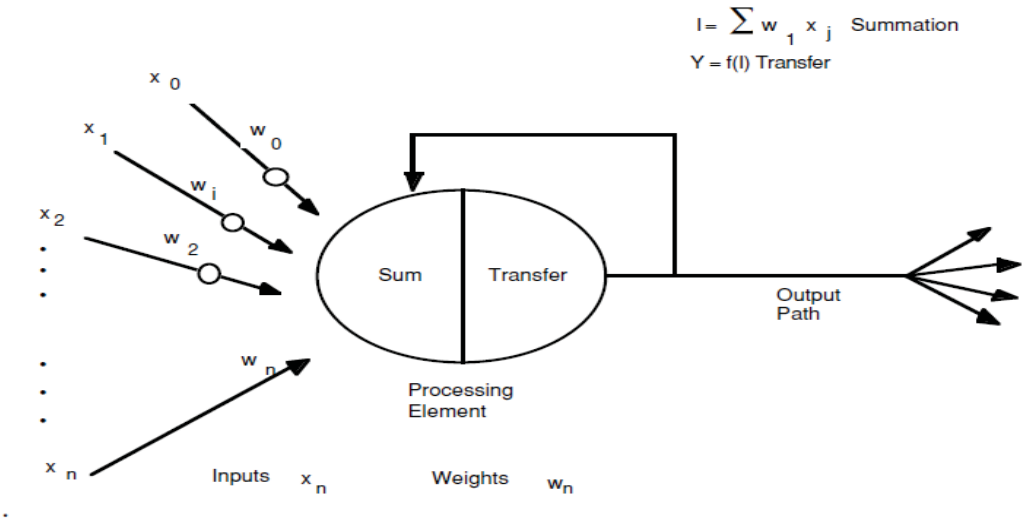


Fig 2: Model of Artificial Neuron

In Figure 2, various inputs to the network are represented by the mathematical symbol, x_n. Each of these inputs are multiplied by a connection weight. These weights are represented by w_n . In the simplest case, these products are simply summed, fed through a transfer function to generate a result, and then output.

The mathematical representation of a neuron depicted above can be described as:

$$Y = f\left(\left(\sum_{i=1}^{n} x_i . w_i\right) + b\right)$$

where x_1 , x_2 ,....x_k represent an input vector, and w_1 , w_2,....w_k represent the weights (or strengths) of the incoming synapses (or interconnections). The bias (b) performs an affine transformation of the linearly combined input signals, and the activation function (f) applies to produce the final output (Y) from the neuron.

6. METHODOLOGY

The study followed a correlation study design that is focused to explore the key variables which recommend or deny students' admission, as well as predict performance of university students. Primary school, Secondary school average mark, WAEC result scores, UTME and PUTME scores and socioeconomic variables were utilized in predicting students' academic success (first year, through fourth year CGPA).

The study was conducted at the College of Engineering and Engineering Technology, Michael Okpara University of Agriculture, Umuahia, Abia State of Nigeria. The populations in this study were 1000 students monitored from year one through four. The percentage ratio of girls to boys is 25% is to 75%. The students were admitted through the traditional admission paradigm which considers only the UTME and PUTME test scores with the WAEC grades.

The sampling method used was Multi-stage sampling specifically, stratified random sampling was used. There are a total of six departments in the College of Engineering and Engineering Technology, which were treated as strata in this study.

One of the study variables was university entrance exam score that are the basis or criteria for students' placement at different colleges and departments in the university For instance, students with relatively low university entrance exam average scores also joined the fields of business, social and applied sciences. On the other hand, students with relatively better university entrance exam average scores were assigned in the fields of studies such as medicine, technology and law based on their choices (Mulu, 2012). 500 were selected in line with (Kurtz, 1983) formula of determining the sample of participants from the target population. The sample size for each department was set proportionally using the target student population of each department.

The information and records of selected students needed for the research were obtained from the admission files of the college admin officer. In this study, information collected from the college officer was used. The data, which

included primary and secondary school information, were used. UTME, PUTME and undergraduate academic performance were used. The questionnaire was adopted from a new instrument (scale) for measuring the socioeconomic status of families (Aggarwal, Bhasin, Sharma, Chhabra, Aggarwal & Rajoura, 2005) .The survey questionnaire was divided into four parts. Part I include department, sex, and student identification number. Part II, dealt with educational background of parents; Part III dealt with items of parents' income and part IV dealt with items on parents' occupation. After the collection of data needed to answer the research questions, both descriptive and inferential statistics were used to analyze the data in this study. Descriptive statistics were computed for predictor and criterion variables. The statistical analyses carried out based on the basic research questions that the study aimed to answer. Correlation matrices were employed depending on the nature of the data and research questions.

7. RESULTS

All the predictor variables were separately checked with the criterion variables for their possible association by standard multiple regression and all of the predictor variables were entered in the stepwise multiple regression analysis to test their combined association on the criterion variable and those variables with p-value <0.05 was selected as possible predictor variables in the final model.

Issued questionnaire	**500**	**100%**
Answered questionnaire	**468**	**93.6%**
Unanswered questionnaire	**32**	**6.4%**

As shown above, the questionnaire was administered to 500 students; the analysis was made using 468 students and the remaining 32 students were unanswered questionnaire.

From the number 468 students studied, 13 persons scored between 4.5 in a scale of 5.0 as average CGPA for 3years studied. This is 2.78% of the total number.
128 scored between 3.5 to 4.49 and this make only 27.35%
117 out of this number (128) that fell within this range were admitted with an average admission score less than 60%. , and 11 scored above 60%
166 scored between 2.5 to 3.49 which is 35.47%
22 out of this number (166) that fell within this range were admitted with an average admission score above 60% and 44 scored less than 60%
139 scored between 1.5 and 2.49 which is 29.7%
123 out of this number (139) that fell within this range were admitted with an average admission score above than 60%.
22 scored below 1.5 which is 4.70%

8. DISCUSSION OF THE RESULT

The main purpose of this study was to find out the main parameters for predicting academic performance of undergraduate students. The study examined performances of students admitted into the college with their Unified Tertiary Matriculated Examination (UTME), PUTME and Senior School Certificate Examination results. It was found out from the study that 9 people out of the 13 people scoring 4.5 CGPA and above (First Class) had admission point less than 60%. And the majority of people scoring CGPA of 3.49 and below is having admission point of 60% and above. This therefore shows that basing admission on the corruptible admission examination scores will not yield the best crop of students for admission. It was also found out that some people who scored between 40% to 48% admission score and were offered admission based on quota systems and other form of favoritism still had CGPA of less than 2.5 in the scale of 5.0.
Therefore, entrance scores should be combined with other parameters such as family background, parents educational status, parental income, type of primary and secondary school attended, family size, etc in recommending or denying admission for students.

Studies have shown that students who come from high income and educated families have significantly predicted high college success than students from low socioeconomic status. Some potential explanations were parents in such settings reported lower educational expectations, less monitoring of children's school work and less overall supervision of social activities compared to students from high socioeconomic and intact families (Jacob and Harvey, 2005).

9. CONCLUSION AND RECOMMENDATIONS

Based on the findings of this study, it is possible to come up with the following conclusions.
For any admission test the validity of the test for its intended purpose should be the primary consideration for the admission decision-makers. Previous education records, parents' income and UTME/PUTME scores , parent education socio-economic background are commonly used in predicting future academic performance. Secondary School performance reflects students' performance in the SSCE results in a variety of subjects. Parents' income reflects to support their children in providing reading materials like books and dictionaries and paying school fees create
differences between children in their academic achievement. UTME are based on subjects that university faculty regard as essential prerequisites for university level learning and that has shown to be positively correlated with university outcomes. Therefore all these predictor variables examined should be considered in offering/denying admission to students.

The study also indicated that students' academic performance is influenced by the socioeconomic background of their parents; as parents that earn high income can take better responsibilities of their children's education compared to parents that earn low incomes. Students whose parents have better jobs and higher levels of educational attainment and who are exposed
to more educational and cultural resources at home tend to perform better than their counterparts without such opportunities. Thus, based on the findings and conclusions of this study, I recommend an in-depth study on the reliability and validity of the UTME/PUTME and SSCE result that is used as a major criterion to university admission in the country. I also recommend the design of a more sophisticated intelligent decision support system that will harness the strength of Artificial Neural Network and Fuzzy Logic for admission recommendation and academic prediction in Nigeria.

REFERENCES:

Zebdewos Zekarias, Nasser Aba-Milki, Fisseha Mikre(2015). PREDICTORS OF ACADEMIC ACHIEVEMENT FOR FIRST YEAR STUDENTS. THE CASE OF WOLAITA-SODDO UNIVERSITY, ETHIOPIA.European Scientific Journal October 2015 edition vol.11, No.28

Aboma Olani. (2009). Predicting first year university students' academic success. *Electronic journal of research in educational psychology,* 7(3), 1053-1072.2009. ISSN: 1696-2095.

Aggarwal, S.K.Bhasin, A.K. Sharma, P .Chhabra, K. Aggarwal, O .P. Rajoura. (2005). A new instrument (scale) for measuring the socioeconomic status of a family. *Indian Journal of Community Medicine* Vol.30, No .4.

Adeniyi, O.S., Araoye, M.A., Amali, E. O., Eru, E.U., Ojabo, C. O., and Alao, O.O. (2010). Effect of Using Combination of O'level Result with JAMB Score on Student Performance in the First Two Years of Medical School in Benue State University, Makurdi. Afr. J. Biomed. 13(3), 189 – 195.

Adewale, O., Adebiyi, A., and Solanke, O. (2007). Web-Based Neural Network Model for University Undergraduate Admission Selection and Placement.Pacific Journal of Science and Technology, 8(2), 367-384.

Adeyemi, T. O. (2010). Credit in Mathematics in Senior Secondary Certificate Examinations as a Predictor of Success in Educational Management in Universities in Ekiti and Ondo States, Nigeria. Middle-East Journal of Scientific Research, 5 (4), 235-244.

Adeyemo, A. B., and Kuye, G. (2006). Mining Students' Academic Performance using Decision Tree Algorithms. Journal of Information Technology Impact, 6(3), 161-170.

Aigbe, P.E. (2008). Design and Implementation of a Web Based University Admission and Placement Neural Network Model. Unpublished M.Sc. Thesis. Ahmadu Bello University Zaria, Kaduna State.
Ajala, O. P. (2010). Three Years of Post UME Screening: Influence on Science Education Students' Achievement. Delta State University, Abraka. International Journal Education Science, 2(1), 29-40.
Ande, E. (2006). Obasanjo accuses JAMB of corrupt practices. The Guardian, June, 13, P 7.
Antonio, B. (2013). A comparative study of crossover operators for Genetic Algorithms to solve the job shop scheduling problem. Wseas transactions on computers 4, 164 - 173
Bandyopadhyay, S., and Saha, S. (2013). Some single and multiobjective optimization techniques. Springer-Verlag, DOI 10.1007/978-3-642-32451-2_2
Barricelli, N. A. (1957). Symbiogenetic evolution processes realized by artificial methods. Methodos, 9(35-36), 143-182.
Behrouz, M.B., and William, F. P. (2003). Using Genetic Algorithms for Data Mining Optimization in an Educational Web-Based. System. Springer-Verlag Berlin Heidelberg . 27(24), 2252–2263.
Binitha, S., and Sathya, S. S. (2012). A Survey of Bio inspired Optimization Algorithms. International Journal of Soft Computing and Engineering, 2(2) 2231-2307.
Bledsoe, W. W. (1961). The use of biological concepts in the analytical study of systems. In Proceedings of the ORSA-TIMS National Meeting.

SESSION

ARTIFICIAL INTELLIGENCE AND RELATED CONCEPTS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

AI Inferences utilizing Occam Abduction

James A. Crowder, Raytheon Intelligence, Information, and Services

Abstract – *Abduction is formally defined as finding the best explanation for a set of observations, or inferring cause from effect. Here we discuss the notion of Occam Abduction, which relates to finding the simplest explanation with respect to inferring cause from effect. Occam abduction is useful in artificial intelligence in application of autonomous reasoning, knowledge assimilation, belief revision, and works well within a multi-agent AI framework. Here we present a flexible, hypothesis-driven methodology for Occam Abduction within a cognitive, artificially intelligent, system architecture.*

Keywords: Artificial Intelligence, Abduction, Occam Reasoning, Inferencing.

1. Introduction

As explained in the abstract, Abduction is formally defined as finding the best explanation for a set of observations, or inferring cause from effect. The notion of Occam Abduction relates to finding the simplest explanation with respect to inferring cause from effect. A formal definition for Artificial Occam Abduction would be:

Artificial Occam Abduction: The simplest set of consistent assumptions and hypotheses, which, together with available background knowledge, entails adequate description/explanation for a given set of observations [2].

In formal logic notation, given $\boldsymbol{B_D}$, representing current background knowledge of domain $\boldsymbol{D}$, and a set of observations $\boldsymbol{O_D}$, on the problem domain $\boldsymbol{D}$, we look for a set of Occam Hypotheses, $\boldsymbol{H_D}$, such that:

- $\boldsymbol{H_D}$ is consistent[1] w.r.t. $\boldsymbol{B_D}$, and
- It holds that $\boldsymbol{B_D}, \models \boldsymbol{H_D}, \rightarrow \boldsymbol{O_D}$

[1] If $\boldsymbol{H_D}$ contains free variables, $\exists(\boldsymbol{H_D})$ should be consistent w.r.t. $\boldsymbol{B_D}$.

Abduction consists of computing explanations (hypotheses) from observations. It is a form of non-monotonic reasoning and provides explanations that are consistent with a current state of knowledge and may become less consistent or inconsistent, when new information is gathered. The existence of multiple hypotheses (or explanations) is a general characteristic of abductive reasoning, and the selection of the preferred, or most simple, but possible, explanation is an important precept in Artificial Occam Abduction.

Abduction was originally embraced in Artificial Intelligence work as a non-monotonic reasoning paradigm to overcome inherent limitations in deductive reasoning. It is useful in Artificial Intelligence applications for natural language understanding, default reasoning, knowledge assimilation, belief revision, and very useful in multi-agent systems [3]. The Abduction form of inference, using hypotheses to explain observed phenomena, is a useful and flexible methodology of reasoning on incomplete or uncertain knowledge. Occam Abduction, by the way it is defined here, provides not only an answer, or cause, to the observations, it provides much more information, in that it describes the properties of the class of possible hypotheses in which the observations are valid, and denotes which is the simplest set of hypotheses under which this is true.

Here is where we diverge from classical Abductive Reasoning, which is generally steeped in Bayesian probabilistics. Fuzzy abduction, as opposed to Bayesian reasoning, utilizes fuzzy sets of hypotheses to explain a given set of observations. The Fuzzy Abduction utilized here

genetically derives a set of fuzzy hypotheses, using the most appropriate of the available fuzzy implications, and uses these fuzzy hypotheses to derive a truth value (how well do the hypotheses explain the observations). This process is considered abductive because it looks for information that both supports and rebuts the fuzzy hypotheses. The combination of supporting and rebutting arguments is used to determine the "possibility" that each hypothesis explains all or part of the observations. Hypotheses whose possibility is above a given threshold are sent forward either to provide explanations, or as input for the next genetically generated set of hypotheses.

2. Elementary Artificial Occam Abduction

There are several distinct types of interactions that are possible between two elementary Occam Abductive hypotheses $h_1, h_2 \in H_e$: [4]

- ***Associativity***: The inclusion of $h_1 \in H_e$ suggests the inclusion of h_2. Such an interaction may arise if there is knowledge of, for instance, mutual information (in a Renyi sense) between h_1 and h_2.
- ***Additivity***: h_1 and h_2 collaborate additively where their abductive and explanatory capabilities overlap. This may happen if h_1 and h_2 each partially explain dome datum $d \in D_0$ but collectively can explain more, if not all of D_0.
- ***Incompatibility***: h_1 and h_2 are mutually incompatible, in that if one of them is included in H_e then the other one should not be included.
- ***Cancellation***: h_1 and h_2 cancel the abductive explanatory capabilities of each other in relation to some $d \in D_0$.
  - For example, h_1 implies an increase in a value, while h_2 implies a decrease in a value. In this case, one is used to support the hypothesis and the other is used to rebut the hypothesis.

The Occam Abductive Process is:

- Nonlinear in the presence of incompatibility relations
- Non-monotonic in the presence of cancellation relations
- The general case (nonlinear and non-monotonic) Occam Abduction hypothesis investigation is NP-complete.

Consider a special version of the general problem of synthesizing an Artificial Occam abductive composite hypothesis that is linear, and, therefore, monotonic.

The synthesis is linear if:

$$\forall h_i, h_j \in H_e, \quad q(h_i) \cup q(h_j) = q(\{h_i, h_j\})$$

The synthesis is monotonic if:

$$\forall h_i, h_j \in H_e, \quad q(h_i) \cup q(h_j) \subseteq q(\{h_i, h_j\})$$

In this special version, we assume that the Occam hypotheses are non-interacting, i.e., each offers a mutually compatible explanation where their coverage provides mutual information (in a Renyi sense). We also assume that the Occam, abductive belief values found by the classification subtasks of abduction for all $h \in H_e$ are equal to 1 (i.e., true).

Under these conditions, the synthesis subtask of Artificial Occam Abduction can be represented by a bipartite graph, consisting of nodes in the set $D_0 \cup H_e$. This says there are not edges between the nodes in D_0, nor are there edges between the nodes in H_e. The edges between the nodes in D_0 and those nodes in H_e can be represented by a matrix $\boldsymbol{Q}$ where the rows correspond to $d \in D_0$ and the columns correspond to $h_t \in H_e$.

The entries in $\boldsymbol{Q}$ are denoted as $\boldsymbol{Q_{ij}}$ and indicate whether the given analyzed data are explained by a specific abductive Occam hypothesis. The entries are defined as:

$$Q_{i,j} = \begin{cases} 0 & \text{if datum } d_i \text{ is not explained by hypothesis } h_j \\ 1 & \text{if datum } d_i \text{ is explained by hypothesis } h_j \end{cases}$$

Given the matrix Q for the bipartite graph, the abductive, Occam synthesis subtask can be modeled as a set-covering problem, i.e., finding the minimum number of columns that cover all the rows. This ensures that the composite abductive,

Occam hypothesis will explain all of D_0 and therefore be parsimonious[2].

Now we look at a special linear and monotonic version of the general abductive, Occam hypothesis synthesis subtask and look at a Possibilistic Abductive Neural Networks (PANNs) for solving it [1]. The first is based on an adapted Hopfield model of computation:

$$\forall i = 1,2,\ldots,n, \quad \sum_{j=1}^{m} Q_{ij} V_j \geq 1$$

For the Occam, abductive synthesis subtask, we associate variable $\boldsymbol{V_j}$ with each Occam hypothesis $h_t \in H_e$, in order to indicate if the Occam hypothesis is included in the composite Occam, abductive hypothesis $\boldsymbol{C}$. We then minimize the cardinality of $\boldsymbol{C}$ by:

$$\sum_{j=1}^{m} V_j$$

Subject to the constraint that all data $d \in D_0$ are completely explained.

For the Occam, abductive network, the problem constraints must evaluate to zero when the constraint is satisfied, and must evaluate to a large positive value when the constraint is not satisfied, forcing the evolving solution lattice to evolve accordingly [5]. For this term, we use a term expressed as a sum of expressions, one for each datum element, $\boldsymbol{d_i}$, such that the expression evaluates to zero, when hypothesis $\boldsymbol{h_j}$ that can explain the datum $\boldsymbol{d_i}$ is in the composite hypothesis, i.e., $\boldsymbol{V_j} = 1$. Given that $\boldsymbol{Q}$ is an incidence matrix (with elements either 0 or 1), we use the expression:

$$\sum_{i=1}^{n} \prod_{j=1}^{m} \{(1 - Q_{ij}) + (1 - V_j)\}$$

And it satisfies the following conditions:

- Each sum of the product terms can never evaluate to a negative number.
- The total sum of the product terms, thus, can never evaluate to a negative number.
- Each product term evaluates to zero when a hypothesis that can explain the datum in the composite; otherwise, it evaluates to a large value.
- The total sum of the product terms, thus, evaluates to zero when a composite set of hypotheses can explain all the data.

We derive our Occam abductive reasoning function as follows:

$$E = \alpha * \sum_{j=1}^{m} V_j + \beta * \sum_{i=1}^{n} \prod_{j=1}^{m} \{(1 - Q_{ij}) + (1 - V_j)\}$$

Where α and β are positive constants, and $\beta > \alpha$. The first term represents the cardinality of the Occam hypothesis and the second term represents the penalty for a lack of complete coverage; 0 indicates complete coverage. The self-organizing algorithm for the Occam abductive network is:

Assume data set : $\overline{X} = \{X^{(1)},\ldots,X^{(p)},\ldots,X^{(P)}\}$, where P is the number of input vectors.
Vector $X^{(p)} = \{x_1^{(p)},\ldots,X_i^{(p)},\ldots,X_{n_1}^{p}\}$ represents the pth input vector to the network.
We initialize STEP and SLOPE $\in (0,0.5]$.
$IT = \beta = TD = 0.5$
For all input vetors $p \in [1\ldots P]$ do {
  For all input dimentions $i \in [1\ldots n_1]$ do {
    if there are no fuzzy clusters in the i^{th} input dimention $(J_i = 0)$
      Create a new cluster using $x_i^{(p)}$
    else do {
      find the best - fit

Figure 1 below illustrates the Occam Abduction Inference process.

[2] Note that the general set-covering problem is NP-complete.

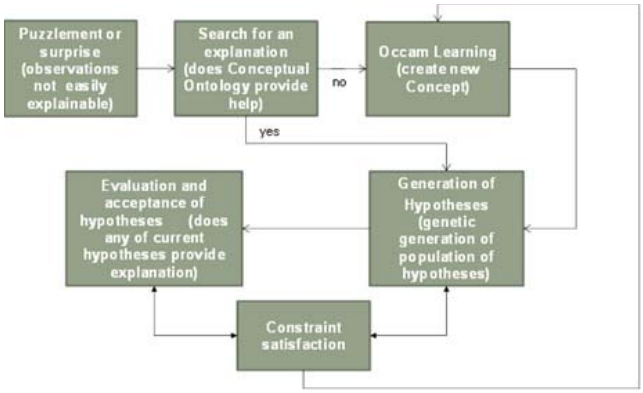


Figure 1 – Artificial Occam Abduction Process

3. Synthesis of Artificial Occam Abduction

Let $B = \{b_k | k = 1, \ldots, l\}$ be a finite set of l Occam learned possible beliefs

Let $H_e \subseteq H$ such that for each $h_j \in H_c$ can explain some non - empty subset of D_0

Let p be a map from H_c to $H_e : p : H_e \rightarrow B$. The map p is also defined from an elementary Occam hypothesis belief value. We define $p(\{h_j\})$ as $p(h_j)$ and interpret $p(h_j)$ as the prima facie Occam belief value for h_j.

The Occam abductive classification subtask takes D, H, D_0, and r as input, where r is a map from $\wp(D_0) \rightarrow \wp(H)$, and gives H_e and p as output.

The abductive hypothesis synthesis subtask may be characterized as a five - tuple (D_0, H_e, q, p, H_c), where D_0, H_e, q, and p constitute the input to the abduction task, and H_c is the output of the task.

Maximal explanitory coverage of hypothesis data :

A composite hypothesis H_1^c is a better explanation of D_0 than another abductive hypothesis H_2^c if :

$$q(H_1^c) \cap D_0 \supset q(H_2^c) \cap D_0$$

Ideally, the assembled composite abductive hypothesis, H_c, would provide adequate explanitary coverage of D_0, i.e., $q(H_c) \supseteq D_0$.

Maximal belief in abductive Occam hypothesis :

A complosite hypothesis H_1^c is a better explanation of D_0 than another abductive hypothesis H_2^c if :

$$p(H_1^c) \geq p(H_2^c)$$

This specifies that among the composite dialectic Occam hypotheses that explain the data, the one with the highest "belief" value is the "best" explanation, by abduction.

Minimal hypothesis : An composite abductive Occam hypothesis H_1^c is a better explanation of D_0 than another compoaite abductive Occam hypothesis H_2^c if :

$$|H_1^c| < |H_2^c|$$

This condition specifies that H_c should be *parsimonious*.

4. Artificial Occam Abduction Hypothesis Evaluation Logic

The following lays out the basics of the Artificial Occam Abductive Logic that will be used to do hypothesis evaluation for the Dialectic Argument Structure, Hypothesis generation and testing system [6]:

Definition 1: A triplet (Φ, Ω, e) defines a domain of Occam hypothesis assembly:

- Φ = The set of hypotheses
- Ω = The set of observations (sensor inputs)
- e = The Mapping from the subsets of Φ to the subsets of Ω
- Assumptions:
  - Computational: For every subset Φ' of Φ, e(Φ') is computable.
  - Independence: $e(\Phi_1 \cup \Phi_2) = e(\Phi_1) \cup e(\Phi_2)$; for all Φ_1 and Φ_2 that are subsets of Φ.
  - Monotonicity: If Φ_1 is a subset of Φ_2, then $e(\Phi_1)$ is a subset of $e(\Phi_2)$.
  - Accountability: α(φ) is the set of observations that cannot be explained without hypothesis φ.

This drives the four-part Occam Dialectic Argument Structure (DAS) Process:

Screening: Screening determines the acceptability of the possible hypotheses and then allocates them in a hierarchical classification system of fuzzy classifications.

Collection: Collection of Hypotheses accounting for the observations. A set of hypotheses is made by adding every hypothesis that explains all or part of the observations.

Parsimony: Parsimony narrows down the collection of hypotheses to the most applicable Occam subset. If a subset of collected hypotheses is able to explain the observations which is the new (narrowed down) hypothesis set.

Critique: Critique determines which hypotheses are the most essential, among the available ones, based on fuzzy inference metrics. Individually, every hypothesis is excluded from the set, and then the set is tested against the observations. If the observations cannot be explained without the excluded hypothesis, then the excluded hypothesis is marked essential and reintroduced into the set.

Definition 2: An Occam abduction system consists of a logical theory 'T' defined over a domain language 'L', and a set of domain syntax 'A' of 'L' that are called abducible[3] [7].

Definition 3: If a set of syntax φ is found as a result of an abductive process in searching for an explanation of ω observations, it must satisfy the following conditions:

- $T \cup \varphi$ is consistent
- $T \cup \varphi \vdash \omega$
- φ is abducible, i.e., $\varphi \in A$

Definition 4: (C, E, T) is a simple causal theory defined over a first order language 'L' where 'C' is a set of causes, 'E' is a set of effects, and 'T' is a logical theory defined over 'L'.

Definition 5: An *Occam Explanation* of a set of observations Ω, which is a subset of E, is the simplest finite set Φ such that:

- Φ is consistent with T
- $T \cup \Phi \vdash \Omega$, where Ω is the conjunction of all $\omega \in \Omega$.
- Φ is a subset-minimal.

Figure 2 illustrates the Artificial Occam Abduction Causal Framework. Figure 3 shows the Occam Abduction Fuzzy Inference high-level architecture

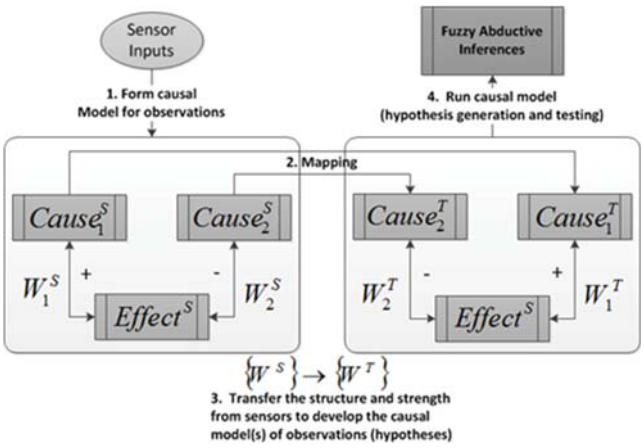


Figure 2 – The Artificial Occam Abduction Causal Framework

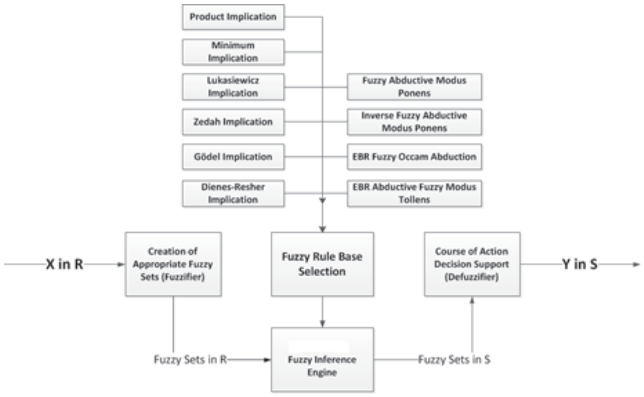


Figure 3 – Fuzzy, Abductive Inference Engine High-Level Architecture

An example of the use of Occam Abduction through the use of an inference engine structure

[3] An abducible argument is a first-order argument consisting of both positive and negative instances of abducable predicates. Abducible predicates are those defined by facts only and the inference engine required to interpret the meaning. In formal logic, abducible refers to incomplete or not completely defined predicates. Problem solving is effected by deriving hypotheses on these abducible predicates as solutions to the problem to be solved (observations to be explained).

shown in Figure 3, within an agent-driven Dialectic Search Argument (reasoning) architecture that utilizes the fuzzy, abductive inferences to find relevant information that then develops a large argument, or inference. The Dialectic Search Argument high-level architecture is illustrated in Figure 4, utilizing the fuzzy, abductive inference engine shown in Figure 3.

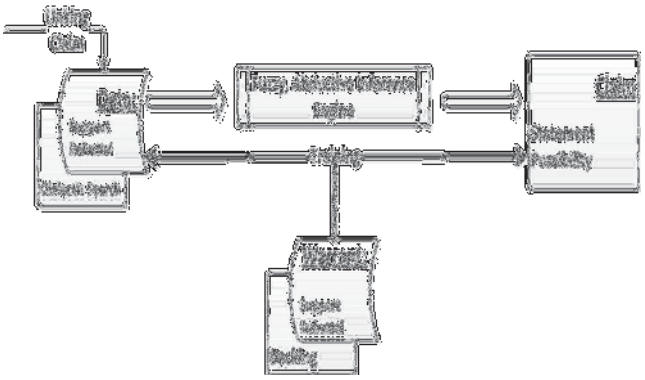


Figure 4 – The Dialectic Argument Structure

The dialectic argument serves two distinct purposes. First, it provides an effective basis for mimicking human reasoning. Second, it provides a means to glean relevant information from Fuzzy, Semantic, Self-Organizing Topical Maps (FSSOTMs) [3] and transform it into actionable intelligence (practical knowledge.) These two purposes work together to provide an intelligent system that captures the capability of a human operator to sort through diverse information and find clues.

This approach is considered dialectic in that it does not depend on deductive or inductive logic, though these may be included as part of the warrant. Instead, the DSA depends on non-analytic inferences to find new possibilities based upon warrant examples (abductive logic). The DSA is dialectic because its reasoning is based upon what is plausible; the DSA is a hypothesis fabricated from bits of information.

Once the examples have been used to train the DSA, data that fits the support and rebuttal requirements is used to instantiate a new claim. This claim is then used to invoke one or more new DSAs that perform their searches. The developing lattice forms the reasoning that renders the intelligence lead plausible and enables measurement of the possibility.

As the lattice develops, the aggregate possibility is computed using the fuzzy membership values of the support and rebuttal information. Eventually, a DSA lattice is formed that relates information with its computed possibility. The computation, based on Renyi's entropy theory, uses joint information memberships to generate a robust measure of Possibility, a process that is not achievable using Bayesian methods.

5. Discussion

Here we have laid the foundations for learning structures that will be required for real-time autonomous AI systems. We have provided a mathematical basis for these learning algorithms, based in computational mechanics. The Occam Abduction is but one of many learning constructs that must be present for an AI system to actually act autonomously and to make sense of a complex world it will find itself a part of [8]. Occam Abduction provides the ability for simple Pattern Discovery that feeds more complex memory and inference systems within an AI cognitive system to allow the autonomous system to think, reason, and evolve. We have but scratched the surface in providing constructs and methodologies required for a self-aware, thinking, reasoning, and fully autonomous real-time AI system.

6. References

1. Crowder, J. 2012. Possibilistic, Abductive Neural Networks (PANNs) for Decision Support in Autonomous Systems: The Advanced Learning, Abductive Network (ALAN). Proceedings of the 1st International Conference on Robotic Intelligence and Applications, Gwanju, Korea.
2. Crowder, J. and Carbone, J. 2011. Occam Learning through Pattern Discovery: Computational Mechanics in AI Systems.

Proceedings of the 13th annual International Conference on Artificial Intelligence, Las Vegas, NV.

3. Crowder, J., Scally, L., and Bonato, M. 2012. Applications for Intelligent Information Agents (I2As): Learning Agents for Autonomous Space Asset Management (LAASAM). Proceedings of the International Conference on Artificial Intelligence, ICAI'12, Las Vegas, NV.
4. Crowder, J., Friess, S., and Carbone, J. 2013. Artificial Cognition Architectures. Springer Publishing, New York, NY. ISBN 978-1-4614-8071-6.
5. Crowder, J. 2013. The Advanced Learning, Abductive Network (ALAN). Proceedings of the AIAA Space 2013 Conference, San Diego, CA.
6. Dimopoulos, Y. and A. Kakas: 1996. Abduction and Inductive Learning. In: L. De Raedt (ed.): Advances in Inductive Logic Programming. Amsterdam: IOS Press, pp. 144{171.
7. Levesque, H. J.: 1989, `A Knowledge-Level Account of Abduction'. In: Proceedings of the Eleventh International Joint Conference on Artificial Intelligence. Detroit, MI, pp. 1061{1067.
8. O'Rorke, P. 1994. Abduction and Explanation-Based Learning: Case Studies in Diverse Domains. Computational Intelligence, 10:295-330.

Schematization for Machine-Oriented Aesthetics

Sabah Al-Fedaghi
Computer Engineering Department
Kuwait University
Kuwait
sabah.alfedaghi@ku.edu.kw

***Abstract*— From the perspective of machine-oriented aesthetics, and according to Levi R. Bryant, the *being of an object* lies not in whatever qualities it happens to manifest, but rather in what an object is capable of *doing*. Objects can be understood as machines, a view that, in general, is true of *art objects* as well. Viewing artworks as *being* not assemblages of signs but machines significantly transforms our approach to them. Whereas an assemblage of signs refers to something beyond itself, a machine, rather, functions and *produces something*. This paper contributes to this line of thought by proposing a methodology to diagram machines that produces engineering-like schemata, thus facilitating a means for analysis and exploration in aesthetics.**

Keywords—machine-oriented aesthetics; schematization; artworks being; diagram; flow

I. INTRODUCTION

Harman [1] interprets Heidegger's [2] modes of being by marking two irreducible aspects of every object:

- Present-at-hand refers to conceiving *being* in terms of tangible "objects of some sort of discussion or perception, and is recognized by a specific shape and color and texture" [1].
- "Ready-to-hand is the mode of beings in themselves: an invisible thing radiating its being is dissolved into the world. It means that prior to any list of properties that can be drawn up, an object is *what he is*. The world would be a different place if it did not *exist*: different for his vast network of other objects" [1]. This mode of being is the focus of this paper.

In this Ready-to-hand mode of beings, "The only way to do justice to objects is to consider that their reality is free of all relation, deeper than all reciprocity. The object is a dark crystal veiled in a private vacuum: irreducible to its own pieces, and equally irreducible to its outward relations with other things" [3].

Accordingly, objects have autonomous existence. Objects flow or move, and they change or break relationships with other objects. Their identity of being remains the same throughout shifting relations and qualitative changes [4]. "The being of an object lies not in whatever qualities it happens to manifest or actualize, but rather in what an object is capable of *doing*; its effects" [4; italics added].

Drawing on Deleuze and Guattari [5], Bryant [4] declares, "all objects can be understood as machines" and what is true of objects in general is true of art objects as well."

> The shift from artworks as assemblages of signs to machines significantly transforms how we approach works and the activity of criticism. Where an assemblage of signs expresses something or refers to something beyond itself, a machine, rather, functions and produces something. [4]

This paper contributes to this line of thought by proposing a methodology to diagram machines that produces engineering-like schemata, thus facilitating a means for analysis and exploration in aesthetics. In addition, such a tool assists in understanding human creativity in its artistic manifestations. From a practical point of view, the diagrams can be used in education to explore artistic values and philosophy.

Diagrams probably rank among the oldest forms of human communication [6]; e.g., Plato's allegory of the cave visualizes situations and depicts knowledge configurations in representational terms.

> Thinking diagrammatically as a way of conceptualizing our world has been in existence from the moment the first cave-person picked up a soft "rock" and started making markings on the walls of his/her dwelling. As civilization progressed, humanity moved into recording; via tablets, papyri, and paper; their activities and learning for posterity. [7]

Traditional logic diagrams (e.g., Venn diagrams, Euler diagrams, Peirce existential diagrams) have been utilized as conceptual representations [8-9], and it has been claimed that these descriptions, in general, have advantages over linguistic ones [10-11]. "The diagram functions as an instrument of making evident the structure of ontology and epistemology… [Descartes made] two-dimensional geometric figures and linear algebraic equations mutually transferable" [12].

The proposed approach is based on so-called *thing-oriented modeling* and utilizes flow-based diagrammatic methodology. "Modeling can take the form of abstraction, fictionalization, idealization, and also approximate representations of what is observable from nature" [7]. In software engineering, *thing-oriented modeling* is used to model a portion of reality through *things* that flow in abstract "river basins," called flow systems or machines, that are part of greater "territories" called spheres and subspheres. Operations on *flowing things* are categorized and limited to creation, release, transfer, receiving, and processing. Relationships among spheres are represented in Venn-like diagrams. Elements of such a model are matched with elements of the domain system. According to Frigg [13],

> A system is a "compact" and unstructured entity and we have to *carve it up* in order to impose a structure on it. Structures do not really exist until the scientist's mind actually "creates" them or, to put it in a less pretentious way, ascribes them to a system. Structures are not "ready-made" but result from a way of taking, or demarcating, the system. (Italics added)

This thing-oriented modeling utilizes a diagrammatic language called the Flowthing Model (FM). FM has been used in several applications (e.g., [14-19]). For the sake of a complete presentation, the basic notions in FM are briefly described in the next section. The example of *a tree as a machine* is a new contribution.

II. Flowthing Model

The Flowthing Model (FM) was inspired by the many types of flows that exist in diverse fields, including information flows, signal flows, and data flows in communication models. This model is a diagrammatic schema that uses *flowthings* (things that flow) to represent a range of items, for example, electrical, mechanical, chemical and thermal signals, blood, food, concepts, pieces of data, activity, and so on. Flowthings are defined as what can be created, released, transferred, processed, and received as stages of a *flow machine* (see Fig. 1). The machine of Fig. 1 is a generalization of the typical input-process-output model used in many scientific fields (Fig. 2).

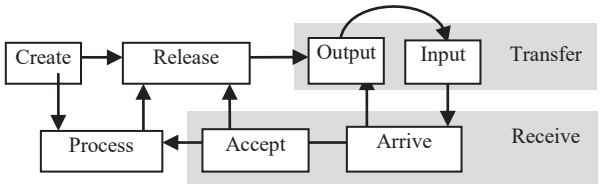


Fig. 1. Flow machine

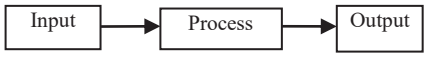


Fig. 2. Input-process-output model

The flow machine depicts processes formed by these five stages at most; fewer stages may be involved. The machine is the conceptual fiber used to handle flowthings (i.e., change them through stages), from their inception or arrival to their de-creation or transmission).

Hereafter, flow machines will be referred to as machines. Machines form the organizational structure (environment) of whatever is described; especially, in our study, these are humans and their physical and nonphysical processes. The machines can be visualized as embedded in a network of assemblies called spheres in which the processes of flow machines take place.

The stages in Fig. 1 can be described as follows:

Arrive: A thing reaches a new machine, e.g., milk flows from breast to mouth.

Accepted: A thing is approved to enter a machine, e.g., milk is not too hot. If arriving things are always accepted, Arrive and Accept can be combined as a Received stage.

Processed (changed): The flowthing goes through some kind of transformation that changes it without creating a new thing, e.g., milk cools.

Released: A flowthing is marked as ready to be transferred outside the machine.

Transferred: The flowthing is transported somewhere from/to outside the machine.

Created: A new thing is born (created) in a machine. This indicates a manifestation of a flowthing in a certain context. For example, the appearance of a character for the first time in a play is a *creation* of this character, who may be released and transferred to another scene in the play.

In general, a flow machine is thought to be an abstract machine that receives, processes, creates, releases, and transfers things. The stages in this machine are mutually exclusive (i.e., a thing in the Process stage cannot be in the Create stage or the Release stage at the same time) with respect to atomic flowthings (flowthings that are not constructed from other flowthings). An additional stage of Storage can also be added to any machine to represent the storage of things; however, storage is not an exclusive stage because there can be stored processed flowthings, stored created flowthings, etc.

Flow machines also use the notions of spheres and subspheres. These are the network environments and relationships of machines and submachines, e.g., a stomach is a machine in the digestive system sphere. Multiple machines can exist in a sphere if needed. A sphere can be a person, an organ, an entity (e.g., a company, a customer), a location (a laboratory, a waiting room), a communication medium (a channel, a wire). A flow machine is a subsphere that embodies the flow; it itself has no subspheres. Control (e.g., interruptions and breaks) of the movement of things is embedded in the stages.

FM also utilizes the notion of triggering. Triggering is the activation of a flow, denoted in flow machine diagrams by a dashed arrow. It is a (causative) dependency among flows and parts of flows. A flow is said to be triggered if it is created or activated by another flow (e.g., a flow of electricity triggers a flow of heat), or activated by another point in the flow. Triggering can also be used to initiate events such as starting up a machine (e.g., remote signal to turn on). Multiple machines can interact by triggering events related to other machines in those machines' spheres and stages.

These flow machines will be used to schematize machines in the Anti-Oedipus. Foucault [20] suggested looking at the Anti-Oedipus as a guide to everyday life to "develop action, thought, and desires by proliferation, juxtaposition, and disjunction, and not by subdivision and pyramidal hierarchization" (italics added). Actions, thoughts, and desires are flowthings. Engineering schematization depicts this development of flowthings as they are created, released, transferred, received, and processed, thus covering all stages of handling action, thoughts, and desires.

Example (taken from [12]): Bogost [21] declared that *a machine is something that operates*. For example, *a tree* is a machine "through which flows of sunlight, water, carbon dioxide, minerals in the soil, etc., flow. Through a series of operations, the machine transforms those flows of matter, those other machines that pass through it, into various sorts of cells

such as photosynthetic cells and bark, while also producing oxygen" [4].

This photosynthethic process is typically explained in educational references in the form of an exploratory sketch. Such a sketch is a simplified drawing that is used to talk about those parts of the world of interest. Fig. 3 shows a typical sketch of the phenomenon of photosynthesis used in a teaching environment.

The purpose of showing such a diagram is to contrast its common diagramming method with the schematic engineering-like form of the FM-based diagramming methodology. An FM diagram provides a schematic language to represent things and input/output processes upon these things, as shown in Fig. 4.

In the figure, carbon dioxide (circle 1), light (2), and minerals (3) flow to the leaves. Note the stream of flow of minerals from the soil (4) through the roots (5) and trunk (6) to the leaves (3). The carbon dioxide, light, and minerals (7, 8, and 9, respectively) reach the leaves (10) where they are processed (11) to trigger the creation of

- Food that is stored (12), and
- Oxygen (13) that flows to the atmosphere (14)

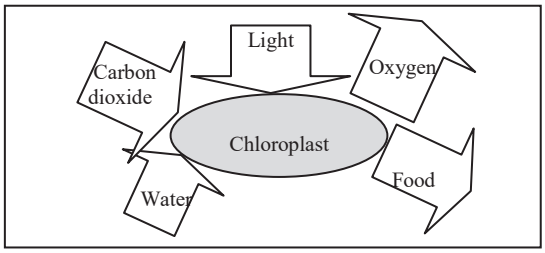


Fig. 3. Photosynthetic process (partial; redrawn from [22])

This type of visual language is usually used to give engineers a base for their design work. In other fields, it can provide a thinking tool. "Today, images 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" [12]. Similarly, such drawings can facilitate analysis in aesthetics. Accordingly, the following section explores the application of FM to modeling of machine-oriented aesthetics.

III. Applying FM to Machine-Oriented Aesthetics

What is true of objects in general is true of art objects as well. Thus, drawing on Bryant's [4] ideas, every art object has a *Creation* (comes into being) or a beginning or origin. But "just as the child is irreducible to her parents," a work of art is irreducible to its creators (e.g., the intention of an author).

Example (taken from [4]): Both Heidegger and Meyer Schapiro analyzed Van Gogh's painting, *A Pair of Shoes* [23].

Heidegger contends that the shoes are a pair of peasant shoes and that the painting discloses or reveals the world of the peasant. Depicting such a view in terms of FM machines can be conceptualized as shown in Fig. 5. Van Gogh's thoughts were processed and reprocessed in his mind (circle 1) to lead (trigger - 2) him to work and rework his painting (3), to trigger (4) the creation of the painting in its manifestation (5). This in turn, according to Bryant's [4] interpretation, created the painting as an autonomous being (6). The manifestation of the painting is transferred to Heidegger (7) to trigger (8) in his mind the thought of a peasant (9).

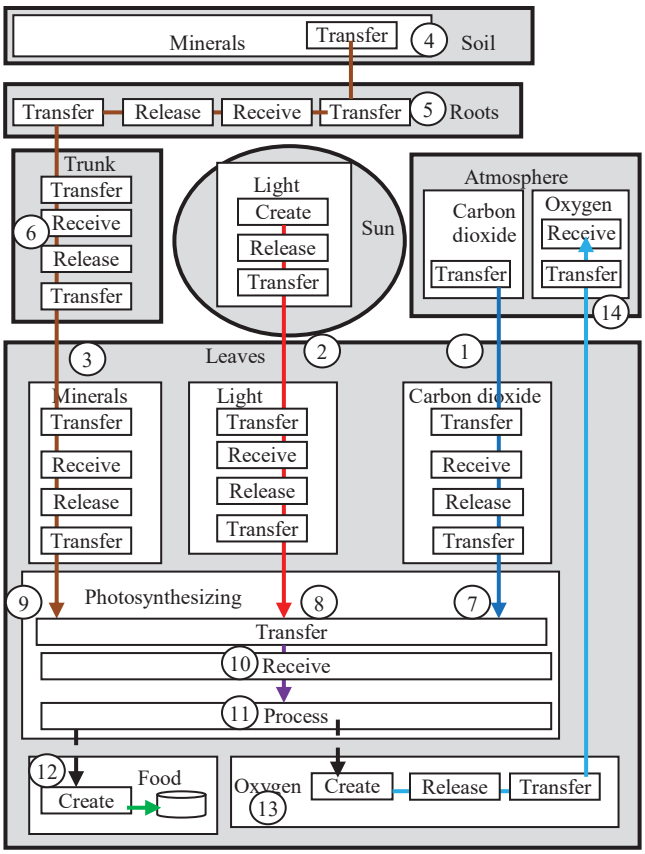


Fig. 4. FM diagram of the photosynthetic process

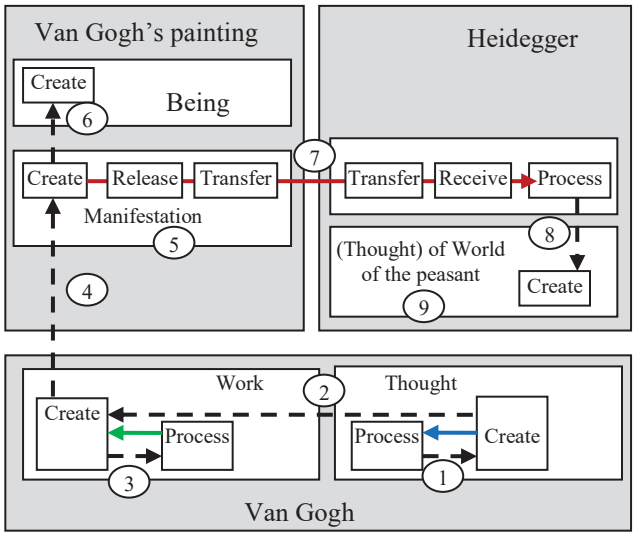


Fig. 5. FM representation of Heidegger's view of Van Gogh's painting

Bryant [4] also discusses the Platonian conceptualization of a work of art as an imperfect expression of the Form of the Beautiful. Such a conceptualization can be recast in the example of Van Gogh's painting, as shown in Fig. 6. The painting is an element in a pedagogy of the soul, directing it (circle 1) towards knowledge of *Forms* (2); however, it draws us away from recollection of the Form of the Beautiful, as in the case of *love*.

> We do not wish to use the beautiful beloved for food, as a tool, for their labor, etc. As such, love is a first step towards a set of values beyond those of the earthly body. On the other hand, love, like art, is one of those instances that assists us in recollecting the Form of the Beautiful. [4]

The danger is that we can confuse a beautiful object we love with Beauty itself, thus making it the real object of our *desire* (see Fig. 7).

> A work of art is a temporary detour on our spiritual path towards the true object of our desire, Beauty as such. At worst, the work of art bars us from access to the true object of our desire insofar as we think it's the work of art that is the object of our desire and regard, rather than Beauty itself and as such. [4]

As exemplified in Figs. 5 and 6 of the example, the diagrammatic language brings the philosophical language to a "tangible" medium of communication, at least for nonphilosophers. It is analogous to supplementing discussion about architectural projects with their blueprints. The stakeholders (owners, architects, users, developers, contractors, …) all participate in this development by expressing their thoughts in language and diagrams. FM provides an additional medium comparable to architectural blueprints for this discussion of the aesthetics of artworks. Bryant [4] might disagree about this understanding of his philosophical thinking; in this case, the diagrams could be redrawn, similar to the case when an architect modifies a blueprint after discussion with stakeholders.

Another conceptualization takes "the form of meditations on an author's or artist's intentions" [4]. The artwork is seen as understanding of an intention (see Fig. 8). The painting is thought to be the world of the peasant (1) in the sphere of Van Gogh's intention (2).

> The critic must engage in a set of translation operations that show how she gets from the features of the work to whatever the work is supposed to express; yet the curious feature of these operations is that they always seem to implicitly suggest that the work could be replaced by the articulation of what is expressed; that what is truly of value is the expressed rather than the expression. The work is treated as a sign standing for something else outside of itself. And just as with the sign, what is important is not the sign itself, but the meaning or referent it points to or stands for, so too with the work of art. [4]

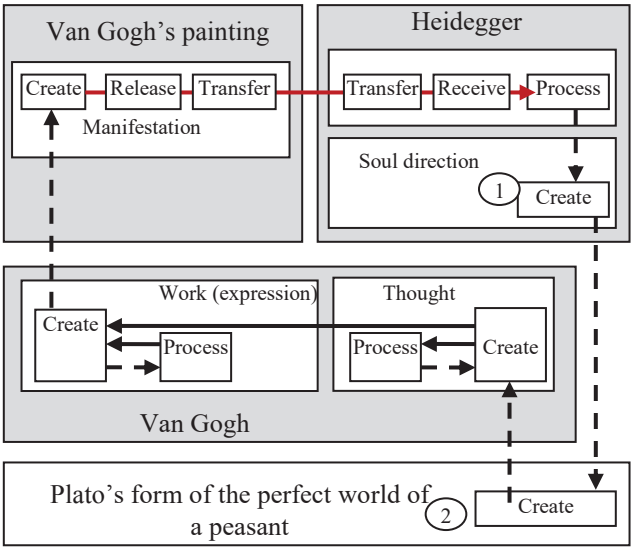


Fig. 6. Platonian conceptualization of the example

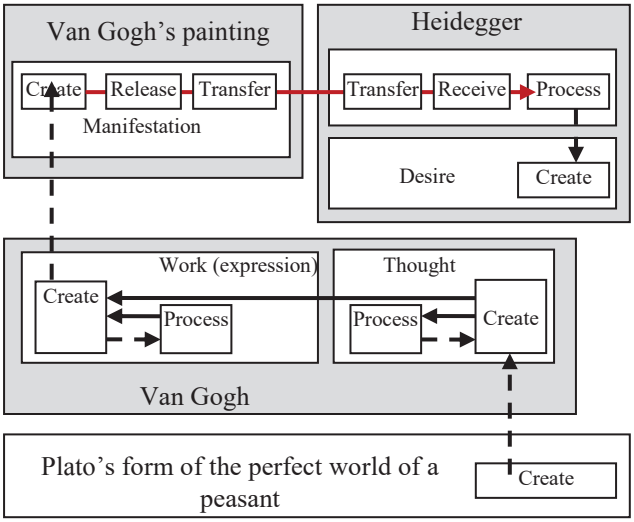


Fig. 7. Making the artwork the real object of one's *desire*

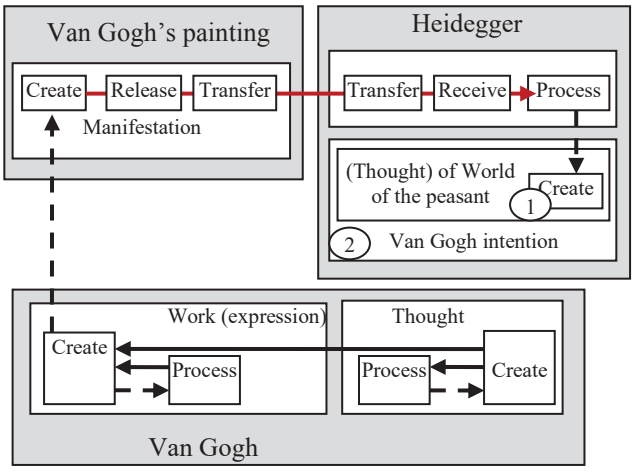


Fig. 8. Understanding the artist's intentions

IV. Being of an Autonomous Existence

Consequently, artworks are treated or intended as assemblages of signs that express or refer to something else. This amounts to erasure of the artwork since the sign as such becomes invisible. Overcoming the erasure of the artwork would be to recognize that a work of art is something. The painting has a completely autonomous existence. It is a real material being in its own right. In this case aesthetics is about the being of artwork.

Fig. 9 shows this case. Van Gogh's painting is a real material being (circle 1) that manifests itself as a phenomenon (2 and 3). Accordingly, it is processed (4) to trigger:

- Its effect: (Thought of) World of the peasant
- Its originator: Van Gogh

It is possible that Heidegger does not have the expertise to recognize that the originator of such a painting is Van Gogh (Fig. 10). Accordingly, he receives (6) this fact from someone (7), as shown in Fig. 10.

In Levi R. Bryant's *The Democracy of Objects* [24], the author wonders whether our encounter with an object touches its reality in our representations. This notion can be conceptualized as shown in Fig. 11, where Heidegger touches the being of the painting.

V. Conclusion

Understanding an artwork as a machine makes the work "no longer understood as expressing something that the critic must decipher, but rather as an entity that affects other entities producing something else" [4]. This paper contributes to this line of thought by proposing a methodology for diagramming machines that produces engineering-like schemata, thus facilitating a means for analysis and exploration in aesthetics. The approach has been demonstrated by diagramming examples from machine-oriented aesthetics. The resultant diagrams seem to enhance analysis and exploration of works of art. Further research will apply the methodology to additional works in the field.

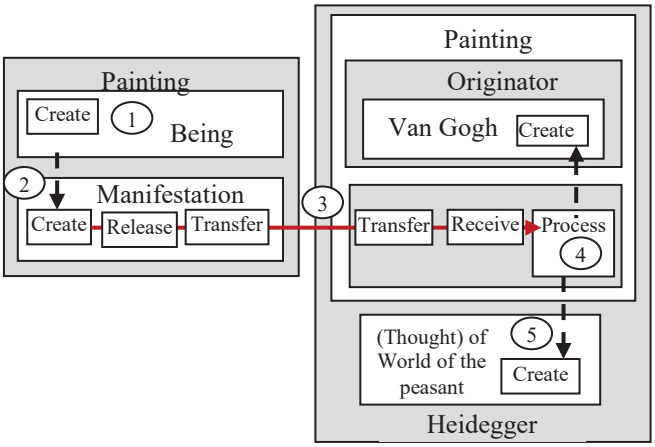


Fig. 9. A work of art is a real material being in its own right

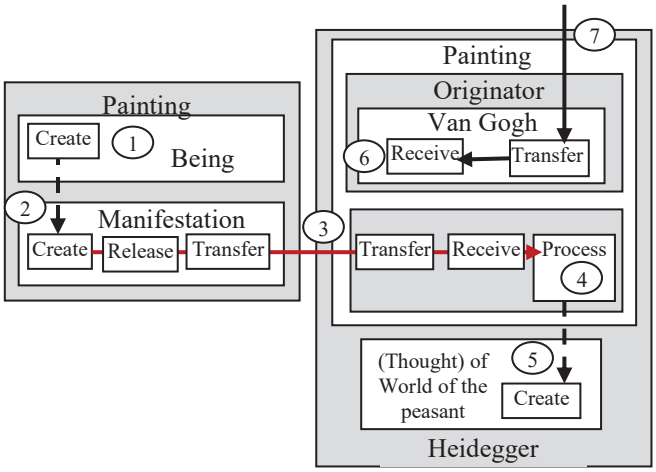


Fig. 10. Receiving knowledge of the artist's identity from outside

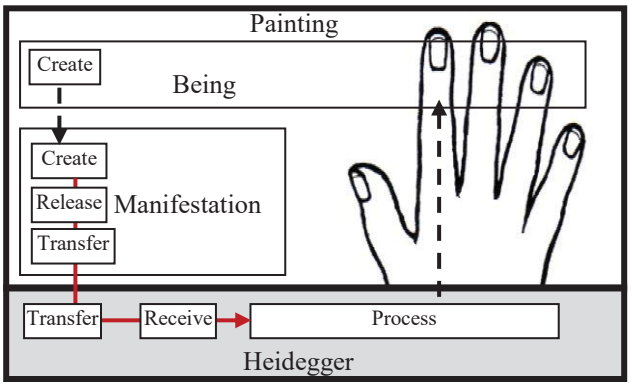


Fig. 11. Touching an object reality

References

[1] G. Harman, "Tool-being: elements in a theory of objects," Ph.D. dissertation, Dept. Philosophy, Coll. Liberal Arts and Sciences, DePaul Univ., Chicago, March 1999.

[2] M. Heidegger, 'The Thing' in Poetry, Language, Thought, trans. A. Hofstadter. 1971.

[3] G. Harman, The Quadruple Object. Washington, USA: Zer0 Books, 2011, p. 47.

[4] L. R. Bryant, "Towards a machine-oriented aesthetics: on the power of art", at The Matter of Contradiction Conference, Limousin, France, 2012.

[5] G. Deleuze and F. Guattari, Anti-Oedipus: Capitalism and Schizophrenia. Minneapolis: University of Minnesota Press, 1983

[6] 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/

[7] C. A. L. Lee, "The science and art of the diagrams: culturing physics and mathematics, part 2 [blog post]. Scientific American, April 2013. http://blogs.scientificamerican.com/guest-blog/the-science-and-art-of-diagramming-culturing-life-and-chemical-sciences-part-ii/

[8] G. Allwein and J. Barwise, Eds. Logical Reasoning with Diagrams. New York: Oxford University Press, 1996.

[9] A. Shimojima, "The graphic linguistic distinction," Artif. Intell. Rev., vol. 13, pp. 313–335, 2001.

[10] C. Gurr, J. Lee, and K. Stenning, "Theories of diagrammatic reasoning: distinguishing component problems." Minds Mach., vol. 8, pp. 533–557, 1998.

[11] 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, pp. 132–148, 2000.

[12] S. Krämer, "Epistemology of the line. Reflections on the diagrammatical mind," in Studies in Diagrammatology and Diagram Praxis, A. Gerner and O. Pombo, Eds. London: College Publications, 2010, pp. 13–38.

[13] R. Frigg. Models and representation: why structures are not enough. Measurement in Physics and Economics Project Discussion Paper Series, DP MEAS 25/02, London School of Economics, 2002.

[14] S. Al-Fedaghi, "Heraclitean ontology for specifying systems, *Int. Rev. Comput. Softw.* (IRECOS), vol. 10, no. 6, 2015.

[15] S. Al-Fedaghi, "On a flow-based paradigm in modeling and programming," Int. J. Adv. Comput. Sci. Appl. (IJACSA), vol. 6, no. 6, pp. 209–217, 2015.

[16] S. Al-Fedaghi, "Conceptualization of various and conflicting notions of information," *Inform. Sci.*, vol. 17, pp. 295-308, 2014.

[17] S. Al-Fedaghi, "System states rediagrammed", IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), 2015, Madrid, Spain.

[18] S. Al-Fedaghi, "High-level representation of time in diagrammatic specification," The 2015 International Conference on Soft Computing and Software Engineering (SCSE'15), University of California at Berkeley, USA, 2015.

[19] 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, 2013.

[20] Michel Foucault, Preface, In: Gilles Deleuze and Felix Guattari, Anti-Oedipus, Translated from the French by Robert Hurley, Mark Seem, and Helen R. Lane, University of Minnesota Press, 1983.

[21] I. Bogost, Unit Operations: An Approach to Video Game Criticism. Cambridge, MA: MIT Press, 2008.

[22] R. L. Kallenbach, Dairy Grazing: Growth of Pasture Plants. University of Missouri Extension, M182, February 2012. http://extension.missouri.edu/publications/DisplayPrinterFriendlyPub.aspx?P=M182

[23] J. Derrida, "Restitutions of the truth in painting," in The Truth in Painting, G. Bennington and I. McLeod, trans. Chicago: University of Chicago Press, 1987.

[24] L. R. Bryant, The Democracy of Objects. Open Humanities Press, 2011.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

METHODOLOGIES FOR ENHANCING EDUCATION AND TOOLS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

An Intelligent Web-Based System for Measuring Students' Attention Levels

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 result from the loss of face-to-face interaction, is the inability to track student's intentional response to instructional content and resources chosen or designed by an instructor. This paper describes an intelligent system that attempts to address this challenge by monitoring student attention while students watch videos in a platform called MyEDBOX. Educators and instructional designers can then use the feedback gathered from the system to evaluate both the individual intentional needs of students and the effectiveness of certain instructional content and resources used in an online course, and adapt the course accordingly. While the results from testing this system are varied, they provide insight that will be used to refine an attention tracking system that can be easily configured and used in a Learning Management System (LMS).*

Keywords: intelligent system, student modeling, machine learning, Bayesian network, attention measurement, computer vision, online education.

1 Introduction

From time to time people in technology and education fields have proclaimed 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 by 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 quiz modules, exercises and online forums where instructors and students can collaborate and ask questions and videos to provide a complete learning package. Moreover, many MOOCs provide relatively easy to use platforms for anyone to create their own online courses. However, even with these advances, 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. Despite our technological advances there seems to be a reluctance to adopt teaching methods that divert from these classic teaching styles. Perhaps one of the most common reasons for this reluctance to fully take advantage of technology in order to change the way we practice education and instruction is the loss of 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 role of an instructor is to facilitate the learning process. This process is facilitated through a complex system made up of a set of components and subsystems that determine what is to be learned and how effectively students 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 classrooms inform educators of 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 learning 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 could 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?

One of the most popular delivery methods of online instruction is video. Video lessons can consist of a series of slides, step-by-step demos or lectures given by one or more instructors. Because of the popularity of videos as a medium to impart instruction, this phase of our research focused on the development of an intelligent, web-based system that uses computer vision to track and estimate students' attention levels as they watch online videos. The system takes advantage of a 3D camera to estimate visual focus of attention (VFOA) based on the student's positions relative to the camera, as well as head pose values (pitch and yaw) as they watch any video. Educators can take advantage of this attention measurement system by using it as a tool to evaluate the effectiveness of their videos based on the estimated levels of attention gathered for each student and for each one of the published videos. In this process, student engagement can be used as feedback for improving online educational content.

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 become more 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 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. 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 as an indication of high levels of attention [12].

While these studies were not conducted in the context of an online course taken individually by a student, they provide a good amount of information that can be used for the design of our system. For instance, Stanley 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 as values used to demine eye gaze and what the users were looking at while performing a test [13].

Nevertheless, as opposed to most of these studies, we had access to a system developed by Intel, a RealSense F200 camera, which when combined with it's associated SDK already handled most computations required for capturing head pose values. Therefore, we were able to focus on

estimating levels of attention and on building a robust modular system that took advantage of this technology. In this case, we focused our efforts on evaluating different methods for measuring and estimating attention and VFOA once head pose data is obtained, regardless of the technology and methods that were used for gathering this data. Bayesian statistics and probabilistic methods, in particular, appear to be efficient methods for making this estimation given that each time new head pose data is obtained the belief and confidence on attention measurements can be updated. Voit and Stiefelhagen, for instance, used a Bayesian approach that uses a model that informs their system of each individual's style of head orientation to derive their focus of attention [14]. Similarly, Ba and Ordobez constructed a Bayesian network to recognize VFOA using head pose data that they were able to gather using probabilistic methods [9].

3 Research Tools and Methods

For tracking head pose and estimating eye gaze, we used the Intel RealSense camera and developer kit. The Intel RealSense camera is able to capture images along the 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 potential for Intel to incorporate the same technology in laptops and mobile devices in the future. This means that systems like the one discussed in this paper and other platforms that take advantage of Human-Computer Interaction (HCI) technology can potentially become more popular and consequently enrich the online education experience.

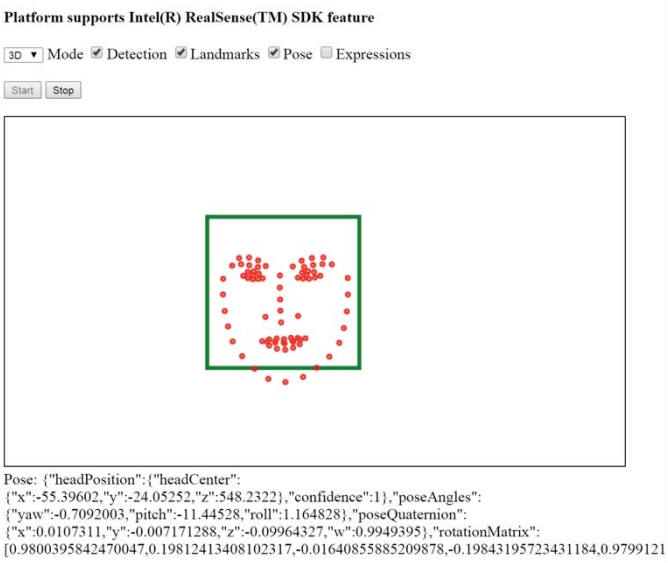


Figure 1. Intel RealSense Image Processing Sample

The main system used for measuring attention based on head pose data is called PyTtention, a modular frontend component built as AngularJs directives. AngularJs is a client-based framework that is compatible with most desktop and mobile browsers, and it allows developers to create interactive UI components using JavaScript [15]. PyTtention allows instructors to display any instructional video inside a modal HTML component that tracks students' faces while tracking head pose data. Once the video is watched in its entirety, head pose data is sent to a server for processing. The server sends head pose data to a number of filters that transform this data into NumPy vectors, which are then processed by a series of Bayesian analysis functions that use historical and calibration data to calculate an estimate of attention. The backend module that PyTtention relies on returns an estimated percentage of attention based on how often it estimates that the student was directing his or her attention to the video in question. Along with this percentage, the API returns statistical data that provides further information as to how attention was measured, such as standard deviation and expected maximum and minimum values for yaw and pose. These values allow us to better evaluate the validity of the attention estimates returned by our API. More specifically, the processes for gathering head pose data is as follows:

1. A student uses the PyTtention calibrator to start supplying the system with head pose values. The calibrator consists of a modal window with a moving red dot that users must click on. These dots move to nine different positions around the modal. As the user clicks on the modal, yaw, pitch and roll values are captured. These values are sent to an Attention backend module that determines and saves the maximum and minimum yaw and pitch values for each calibration activity and associates those values with the average X, Y and Z position where the user's face was positioned while clicking on each dot.
2. After performing a minimum of three calibrations, the user watches a video using the PyTtention video player. As the user is watching a video, the PyTtention components capture head pose data (yaw, pitch and roll) every second. Anytime the RealSense camera does not detect the user's face, the video is paused automatically and the user is asked to continue the video after making sure that his or her face is in the camera view. Once the video concludes, the user is asked for an estimated percentage of attention. This value is used temporarily for testing purposes and can eventually be replaced by a test based on the content for the video that the user watched. At this point, head pose data is sent to the server using REST calls.
3. Data is transformed into NumPy vectors for processing. The Attention API compares the average position of the user's face relative to the camera to data gathered during calibration activities for that particular user. Head pose values are then passed through a series of functions that perform Bayesian analysis and gather statistical information on that data.

4. Results are returned to the browser, stored as Activity objects in the database and displayed for further analysis. Both the user and the instructor have access to these data.

To estimate attention we used Bayes' Theorem to get better estimates of attention every time a student performs a new activity. An Activity, in this context, is recorded anytime a user watches video using the PyTtention system. Bayes' Theorem gives us the following formula for determining the probability of an event occurring given new data:

$$P(H|E) = P(H) * P(E|H)/P(E) \quad (1)$$

In this case, we want to know the probability that the user was attentive for any given video given the new head pose data gathered by the PyTtention video player. In this sense, we are only concerned with two hypotheses:

H_1 = the user was attentive for video X.
H_2 = the user was inattentive for video X.

In order to make this assessment for every second that user watched a video, the system relies on calibration data. A function searches the database for the calibration data set that most closely matches the average position of the user's face while watching the video in question. Therefore, in this case the likelihood value of Bayes' theorem is calculated based on the amount of time that the user's yaw and pitch fall within the expected maximum and minimum yaw and pitch values captured during the selected calibration data set. In this sense, the system assumes that the user is attentive when pitch and yaw values for any given second are within the range of maximum and minimum yaw and pitch values for the best calibration found for the current user.

For calculating *P(H)*, we initially generated a random Gaussian distribution of attention using historical attention measurements for each individual user; however, we found that measurements were more accurate when averaging attention values for the user regardless of the video that he/she watched. Because of this, the first time a user's attention is measured, only the likelihood values are returned.

For our normalizing constant we used the following formula:

$$\sum_{i=1}^{n} P(H_i)P(E|H_i) \quad (2)$$

where we simply add the result from multiplying prior distribution values for both our hypothesis and multiplying them by their livelihoods.

Figure 2 shows what a typical report looks like for a particular activity. The data supplied in each report allows instructors to objectively analyze and valuate attention measurements.

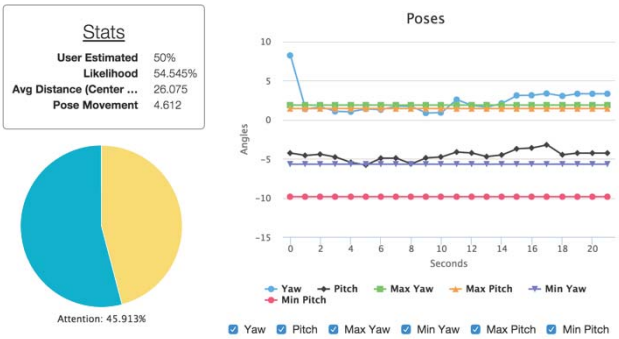


Figure 2. Sample Results Report

In addition to the per activity reports that are returned by PyTtention, each user has access to his/her baseline data, depicted in Figure 3, which instructors can also access for all of their students. Each time a new activity is completed, statistical information is updated in order to get a sense of the overall attention levels of each student over time.

Figure 3. Baseline Data

In this particular case, the user is only 18% likely to be attentive for any video. The report also includes information regarding the last activity performed. We can see how the user estimated that he was 50% attentive and the system estimated 45%. Additionally we can see how the much the position of the user's face varied during the duration of the video as compared to the best calibration available (SD (Center) and Z-Score (Center)).

4 System Architecture

The initial design focused on building PyTtention as a plugin for a popular MOOC that could be added or activated by online instructors and educators as a way to track attention data using an existing platform and content. However, after initial attempts using this approach, we decided to develop a

modular system that could be broken into several components and used not only for the main goal, tracking attention, but also for extending those components to build additional intelligent tools for education. Therefore, we developed a web-based platform called MyEDBOX that consists of a Django backend that uses Python to create an API that can potentially be called by other tools and systems to process head pose data in order to get an estimate of attention for different activities. Moreover, the frontend of this application consist of multiple, reusable, components built with AngularJs that perform a number of tasks including managing the RealSense cameras using custom services, controllers and directives. This approach allows us to work and experiment without the restrictions and limitations imposed by a specific LMS system. Therefore, PyTtention was built as a module inside the MyEDBOX platform. MyEDBOX is currently under development as an open source LMS that is easy to use for instructors and developers. Many of the components built for MyEDBOX will be published in the future as open source, frontend components.

4.1 Frontend Architecture

The frontend consists of multiple modules, each with its own set of controllers, directives and services. Services act as singletons that can be injected in controllers to perform different tasks, such as calling API services or performing common operations. Directives on the other hand use HTML templates to create reusable UI elements. Figure 4 depicts the main components of PyTtention. PyTtention consists of a series of reusable AngularJs components that are combined using dependency injection and that communicate with an attention processing view class in the server called AttentionProcessor.

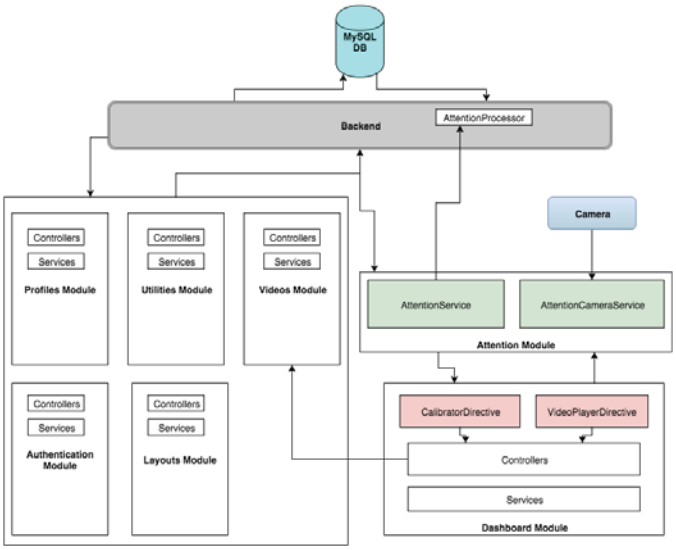


Figure 4. Frontend System Architecture

The main frontend components of PyTtention work together as follows:

1. Head pose data is captured using the AttentionCameraService. This service includes functions for verifying that all camera drivers are installed as well as functions for initializing the camera, pausing, stopping, and starting streaming of head position and head pose data. It also includes event-reporting functions that broadcast messages to all and any of the frontend code for notifying the application of different camera events.
2. The CalibratorDirective and the VideoPlayerDirective use HTML templates that wrap the functionality required for capturing attention data from users as they complete a calibration or a watch a video. This is accomplished this by querying the AttentionCamera service. In this case, the controllers are setup to request the AttentionService to send head pose data to the server.
3. The AttentionService is responsible for sending head pose data to a server using POST calls formatted as JSON objects, as well as getting head pose, attention and calibration data from the server for specific users using GET calls.
4. Head pose data is received and processed by a number of View classes in the backend that correspond to different data models.

4.2 Backend Architecture

The backend is where most processing of head pose data occurs. It consists of a Django backend that follows the Model-Template-View (MTV) pattern. The backend is also responsible for other tasks that are not specific to the PyTtention module. This includes tasks such as authenticating users, saving and managing videos. Figure 5 highlights the most relevant classes for processing and managing attention data for each user.

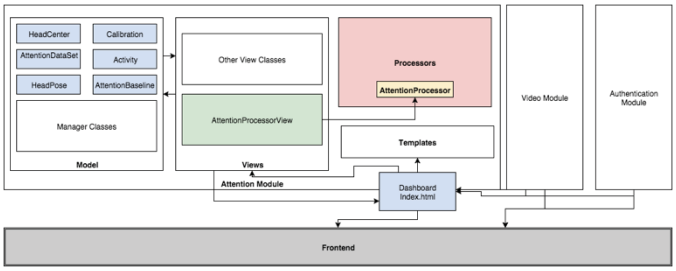


Figure 5. Backend System Architecture

The backend processes attention data in the following manner:

1. Head pose data is received from frontend code as a POST call from the frontend code using the AttentionProcessorView class.
2. This class then uses a processor class called AttentionProcessor that contains a series of static methods responsible for transforming data into NumPy vectors, which are then processed using

instance methods to estimate attention. Operations described in the Research Tools and Methods section occur at this point, as this class updates baseline values for each user as well. Measurements are stored in an object that the AttentionProcessorView class has access to. Data is then saved in the database using the model and model manager classes and results are returned to the user.

5 Tests and Results

5.1 Testing Procedure

In order to test the system and evaluate the accurateness of the attention estimates that it returns we had six people of different ages test the system in a controlled setting. We published the system on the web at www.myedbox.com and asked testers to perform the following steps:

1. Complete three calibrations.
2. Select and watch two videos from a list that we created for all users. Most users preferred to add their own videos.
3. Add a video of their choice and watch it as well.
4. Enter estimated levels of attention at the end of each video as required by the application.
5. Discuss results to determine whether they believed the attention measurements were accurate.

The first three testers went through the steps above in an office setting. The last three performed the test in a much more relaxed setting. All tests used the same laptop, and the camera was balanced on top of the screen.

5.2 Results and Conclusions

Out of all testers, the best results were obtained for the last three testers, as attention measurements more closely approximated the estimations made by users regarding their own levels of attention. In fact, only two of the measurements that were obtained for the first three users returned values above 0% attention. After analyzing the results for all activities (videos watched) for each user, we can draw the following conclusions:

1. The least accurate attention measurements (where attention values were 60 to 99% lower than those estimated by users) resulted for users who performed all three calibrations with the least amount of movement during each calibration. For instance, two users who thought that the system tracked eye gaze rather than head pose focused on moving only their eyes while calibrating the system. When they watched a video, they repositioned themselves in their seats and adjusted the screen and/or camera. The low variation in calibrations often resulted in a poorly trained profile and baseline that the system could use data to calculate attention. See Figure 6.
2. Because the likelihood functions assign the same importance to yaw and pitch, attention measurements were often greatly affected by jumps in yaw values. Even when a graph for pitch seemed to closely match a user's estimations of attention, attention indices were off by 20-40% given that yaw varied a lot more than pitch. See Figures 7a and 7b.
3. The best results (when estimates of attention made by the system more closely matched those made by users) resulted from users whose average head position relative to the camera more closely matched the average head position of any of their available calibration data sets. This was the case of users who moved the most during each one of their calibrations but did not reposition the camera or screen while watching videos. See Figure 8.

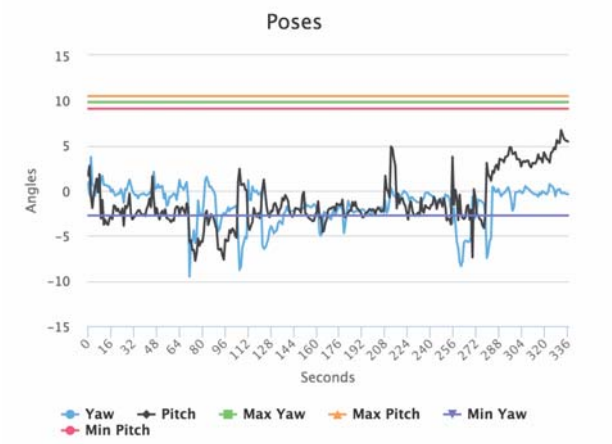


Figure 6. Results from a user who performed all calibrations while consciously focusing on eye movements. Head pose data is completely outside the range of the best calibration dataset available for comparison.

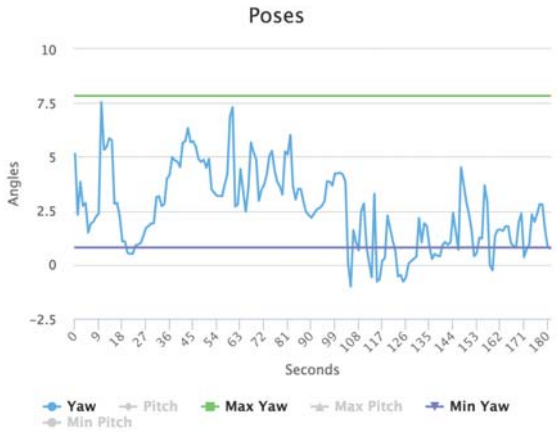


Figure 7a. User estimated 90% of attention for this activity; however, the system estimated only 60.44% of attention. Nevertheless, we can see in this graph how pitch values seem to be within the expected range for attention more than 90% of the time.

Figure 7b. Pitch values may have incorrectly affected attention estimates.

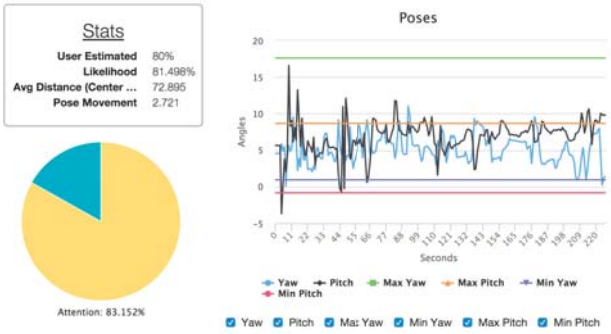


Figure 8. The user estimated 80% attention and the system measured 81.15%. The average changes in pose values are relatively minimal. The accurateness of these results are also likely due to the fact that the best available calibration closely matched the average head pose of the user (Z-score is only 0.007).

6 Discussion and Future Work

Capturing attention data with respect to specific online educational content allows online instructors to further understand the needs of their 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 likely 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 provide additional or other supporting materials and resources for students.

The above scenario will of course require the judgment of the instructor when determining whether the instructional content is the 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 for individual students and for a class a whole.

One of the most important enhancements to the MyEDBOX system could be an evaluation module that allows instructors to create short tests or quizzes for each video. Results from those evaluations could be used to better judge attention estimates. Moreover evaluation results could be used to create a more robust Bayesian network that uses test scores for each video to adjust attention probabilities. This would require an analysis of the dependency between test scores for each video and attention measurements. Additionally, based on a statistical analysis of the results obtained during the testing phase, the following changes will be made and re-tested:

- Discard the use of pitch as an indication of attention and rely only on yaw as the only head pose estimation value.
- Integrate an eye gaze module recently added to the RealSense SDK that will inform the system where the user's eyes are focused on.
- Incorporate amount of variation in head pose changes in attention calculations.
- Desensitize the head pose capture modules by using baseline data (average Z-score and SD) for each user to adjust head pose values used for attention measurements.
- Conduct blind studies, rather than explaining how the system works or what it is trying to do to each tester.
- Build an administration portal that would allow instructors to refine the way calculations are made.

In order to make the system more robust and to continue working towards the greater goal of making online education more effective and intelligent, we will continue working on the MyEDBOX system as an intelligent LMS by expanding its functionality and adding other planned features and modules.

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] Voit, M., & Stiefelhagen, R. (2006, November). Tracking head pose and focus of attention with multiple far-field cameras. In *Proceedings of the 8th international conference on Multimodal interfaces* (pp. 281-286). ACM.

[15] AngularJs Developers Guide: Introduction. *What is Angular?* Retrieved from https://docs.angularjs.org/guide/introduction.

Programming Languages with Plan Knowledge Representation for Learning

Christina Schweikert
Division of Computer Science, Mathematics and Science, St. John's University, Queens, NY USA

Abstract – *Introductory programming courses can be taught in a variety of languages that are current with industry demands and support essential features such as object oriented programming. Programming languages can also be compared using a variety of language criteria, based on the goals of a particular course. Exposure to a variety of programming paradigms is also an important component of a computer science curriculum. A plan-oriented approach for teaching programming provides a framework that is language independent, but can be applied to any language. A planning approach can also help students design classes for object oriented programs and incorporate objects into a larger solution. In addition to learning a programming language, it is vital for students to emphasize computational thinking, as well as language flexibility.*

Keywords: programming languages, planning, novice programming

1 Introduction

The computational and algorithmic thinking skills developed in undergraduate computer science course are fundamental to students' success in the field. In programming, data structures, and algorithms courses, these skills are taught conceptually and practically utilizing a programming language of choice. There has consistently been a debate as to which programming language is the best suited for teaching novice programmers. Python, Java, C, and C++ are among the most common language for introductory programming courses. From a comparative programming language perspective, these languages can be evaluated based on typical programming language evaluation criteria, including: simplicity, orthogonality, data types, syntax, abstraction, expressivity, type checking, exception handling, and other measures [8]. There are existing studies that compare the language features of first programming languages. In [2], the authors conduct a thorough investigation in which they analyze common imperative and object oriented languages, by examining these features of the language: high level, orthogonality, strongly typed, enforcing good programming habits, security, uniformity of features, ease of writing simple programs, industry demand, contemporary features, ease of transition to other languages, readability of syntax, coding standards, and the existence of a user-friendly Integrated Development Environment. According to the framework and scoring system designed by the authors in [2], Java is top-ranked, second-ranked is Python, followed by others.

In this paper, programming languages are examined according to features that may play a role in students' experience, as well as equipping students with necessary concepts. Problem solving strategies will be similar within a particular paradigm of languages. It is vital that students have a working knowledge of the intricacies of a particular language (languages), in addition to becoming skilled at computational thinking and problem solving in general. Planning, as it has been applied to programming, may serve as a useful tool to pull these skills together.

2 Brief Programming Language Comparison

Choice of programming language for a class naturally varies depending on the type of class: introductory programming course, data structures and algorithms, software engineering, or web development. For a first programming course, the decision may be different depending on whether a course is intended for computer science majors or non-majors. Computer science majors need the material as building blocks for more advanced courses, while non-majors may like to get started quickly and absorb programming basics that may be applied to a different domain.

2.1 Getting Started

Sample programs to print a simple string (`"Hello World"` type programs) are shown in Table 1 for C, C++, Java, Python, Scheme, JavaScript, PHP, and Perl. Of course, the scripting languages (i.e. Python), necessitate the least overhead when writing a simple one-line program. In C (procedural), students need to understand how to include libraries and create a main method (similar in C++, where a class is not required). However, to display a simple string in Java, students are required to create a class, as well as a main method.

2.2 Python and Java Features

Java and Python, both popular languages for introductory programming courses, have some important distinctions which are the focus for this discussion. Data typing: In Python, typing is dynamic, while typing is static in Java. In Java, variables also need to be declared with a type

before they are used, which is not necessary in Python. Though Python's typing may seem easier to use, it may result in unintended errors from novices (i.e. unintentionally reusing an existing variable with a different type). In addition, the need to declare variables in Java helps novices learn about the different data types and their appropriate usage. Code structure: Programs in Python are structured using indentation, while Java programs use braces { }. Some can argue that use of braces provides a more clear structure, while others may argue that Python's indentation requirements enforces good coding style. Java is considered more portable, since it runs on any platform with the JVM (Java Virtual Machine) installed, while Python requires the compiler. It is worth noting that Java is also used to develop applications for Android devices. Equivalent sample programs in Python and Java that involve defining a class and creating an object of that class are shown in Fig. 1 and 2, respectively. When implementing a class in Java, instance variables are declared with a data type and can be set to private accessibility (encapsulation). In Python, data attributes are created when they are initialized in the class' `__init__` method; only class variables (static variables in Java) are declared inside the class, outside of a method. In Python, all data attributes and class variables are accessible from outside the object. All object methods in Python must be explicitly passed the object reference (called `self`) as the first parameter; in Java the reference to the calling object is an implicit parameter (called `this`) to the methods of a class. Python provides multiple inheritance capabilities, while Java has single inheritance, but allows for classes to also implement multiple interfaces.

Table 1. Sample "Greetings!" programs in C, C++, Java, Python, Scheme, JavaScript, PHP, and Perl

C	**C++**

`#include<stdio.h>`
 	
`int main(){`
`  printf("Greetings!\n");`
`  return 0;`
`}`
	`#include <iostream>`
	`using namespace std;`
	`int main(){`
	`  cout << "Greetings!"<<endl;`
	`  return 0;`
	`}`

Java	**Python**

`public class HelloPrinter{`
`  public static void main(String[] args){`
`  System.out.println("Greetings!");`
`  }`
`}`
	`print('Greetings!')`

Scheme	**JavaScript**

`(display "Greetings!")`	`<script>`
	`  document.write("Greetings!");`
	`</script>`

PHP	**Perl**
`<?php echo 'Greetings!'; ?>`	`#! /usr/bin/perl` `print "Greetings!";`

Regardless of the specific language chosen for an introductory course (among the imperative, object-oriented languages), students will need to master the same computational thinking skills and problem solving strategies. Working in the object-oriented paradigm adds another layer of abstraction that requires the students to understand how to design classes and correctly assimilate them into a larger program.

2.3 Breadth of Programming Paradigms

In addition to mastery in a particular language, it is advantageous for students to be exposed to programming languages from different paradigms in their curriculum (either in a programming language course or across different courses). This will serve students well in their career as they will be able to identify the most suitable type of language for a particular project. For example, some experience with a functional programming language (such as Common Lisp, Scheme, Haskell, F#, among others) will force students to think and solve problems in a different mindset (understating syntax such as `(+ 7 5 3)` in Scheme, as well as how functional programming works). It is also useful to have experience with web scripting languages and event-driven programming in order to create applications with that type of user interaction. Application to solving problems in interdisciplinary areas, such as business and science, can help students identify strengths of different languages. For example, in biology, languages such as Python, Perl, and R, among others have been popular, as are Python and R for statistics and data analytics.

```
>>> class Rectangle:
...     def __init__(self, l, w):
...             self.length = l
...             self.width = w
...     def computeArea(self):
...             return self.length * self.width
>>> x = Rectangle(10, 5)
>>> x.computeArea()
```

Fig 1. Python program with class and object

```
public class Rectangle {

  private double length;
  private double width;

  public Rectangle(double l, double w){
    length = l;  // or this.length = l;
    width = w;  // or this.width = w;
  }

  public double computeArea(){
    return length * width;
    //or return this.length * this.width;
  }
}

public class RunRectangle {

  public static void main(String[] args) {
    Rectangle x = new Rectangle(10, 5);
    System.out.println(x.computeArea());
  }
}
```

Fig 2. Java program with class and object

3 Representing Programming Knowledge with Plans

Planning, as applied to representing programming knowledge, focuses on capturing typical patterns used to solve programming problems. Plans are based on the idea that expert programmers have, over years of experience, developed a set of structured patterns that can be drawn upon and combined to create solutions when faced with a new problem. In general terms, a plan is a series of actions required to reach a goal. This can be likened to a chess master who has accumulated experiential knowledge of the game. When faced with a new situation in a chess match, the chess master exploits plans (in this case a series of moves to win the game) and combinations thereof, that they have gained knowledge of from similar scenarios in the past. Scripts and plans come from their use in natural language processing (NLP). Plans represent experiential knowledge that can be applied, and customized to solve new problems. It has been shown that expert programmers can remember programs better than beginners if the programs have a recognizable structure, pattern, or plan [9]. An example would be a simple plan to find the average of an array of values (Average Plan), which consists of sub-plans for computing a total (Total Plan) and dividing (Divide Plan). These plans can then be applied to compute the average of GPA values stored in an array of student objects, for example. Similarly, plans for searching or sorting can be demonstrated with a simple array of integers, and then applied to an array of Student objects to be searched by id or sorted by GPA. A Plan is language-independent, but when implemented as code, it will be represented in a specific language of choice. This notion of plans is specific to teaching programming, which is separate from the broader field of planning in artificial intelligence [5].

Plans can be used to address difficulties students have with object-oriented programming, such as determining the appropriate instance variables and methods, implementing methods, and integrating objects into a larger program solution. A system for learning programming with plans can have the following phases: Plan Observation, Plan Integration, and Plan Creation. The Plan Observation phase demonstrates the solution to a problem step by step. A program description is shown, along with the plans needed and how they are integrated to form the solution. The plans are then implemented in code step by step, showing how each plan is represented in the chosen programming language. Plan integration tests whether a student can properly integrate plans to solve a particular problem. The Plan Creation phase enables students to customize their own plans, for example by designing new classes (for objects required in a program). A web system that contains these phases has been designed [6, 7].

When teaching object-oriented languages, the order in which topics should be covered has been debated [1, 3, 4]. Specifically, whether to start with the basic building blocks (variables, decisions, loops, methods) and work up to objects, or to start with objects early on. The use of plans can aid in the teaching of objects from a conceptual and problem solving perspective.

4 Conclusions

Programming language choice for introductory programming courses can shape the way the students experience the material and influence their problem solving strategies. Experience with various programming paradigms, including object-oriented programming, functional programming, and different types of languages and applications, is also valuable for students' overall learning. Developing expertise in a particular programming language is essential. However, it is also critical that students develop a conceptual understanding of programming, computational thinking, and implementing a problem solution as a program. Students often have difficulties making the leap from solving a problem to writing the corresponding code. Plan representation of programming knowledge may assist students in the integration of pieces of a solution, and the transformation of the solution into code. Planning will also reinforce the concepts needed to design classes for object-oriented programs, as well as to incorporate objects into a larger program.

5 References

[1] 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.

[2] M.S. Farooq, S.A. Khan, F. Ahmad, S. Islam, A. Abid, An Evaluation Framework and Comparative Analysis of the Widely Used First Programming Languages. PLoS ONE 9(2): e88941, 2014.

[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] S. Lucci and D. Kopec, Artificial Intelligence in the 21st Century, 2nd ed. Dulles, Virginia: Mercury Learning & Information, 2015.

[6] 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.

[7] C. Schweikert, The Role of Planning in Object-Oriented Programming for Beginners. Proceedings of the 2015 International Conference on Artificial Intelligence (ICAI 2015), Volume 1, 2015, pp. 239-245.

[8] R. Sebesta, Concepts of Programming Languages, 11th Edition, Pearson, 2015.

[9] 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.

SESSION

XVI 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

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Analytic Knowledge Process: An Application of decision making techniques in an Implementation Information System

Alcalde R.[1], Sáiz L.[2], Manzanedo M.A. [2], and Alonso B.[3]
[1]Department Economics and Business Administration, University of Burgos, Burgos, Spain
[2]Department Civil Engineering, University of Burgos, Burgos, Spain
[3]Department Chemistry, University of Burgos, Burgos, Spain

Abstract - *This paper presents a methodology to develop a methodology of Analytic Knowledge Process (AKP) at the same time as IS implementation. Furthermore, it explores the possibilities, difficulties and substantial advantages of applying AKP to answers to the following questions: How can AKP contribute to the implementation of an IS? How can AKP be used to quantify the resources, to identify the people that require training, to assign authorizations and to understand information flows within the organization of workers? What post-implementation benefits does AKP bring with it? How much does AKP cost? When is the best time to begin AKP? How can represent the knowledge network?.*

Keywords: Analytic Knowledge Process, Lean Knowledge management System, Information System, Knowledge networks and management, Knowledge Acquisition

1 Introduction

Over recent years, many organizations have been carrying out knowledge management (KM) projects and projects to renew or implement new functional aspects of their information systems (IS) [14]. Although an IS is software for enterprise management, it has to support every one of the processes behind the company's value chain [13]. Consequently, a large part of the capitalization of the company is contained in the IS [1].

1.1 Knowledge Process inside an Enterprise

Knowledge can also be found in different states: Explicit, Tacit, Singular and Collective. Explicit knowledge is objective and concrete. It can be expressed in definite language and propagated, and is therefore found in quality system manuals, and is evident in the company strategy, as well as in normalized forms, regulations, and supplier and client documentation. In contrast, tacit knowledge is subjective and unique. It is in an organization, clients, suppliers, competitors, and shareholders and it is this shapeless knowledge that is of most value to the firm, because it belongs to a person instead of a community and can therefore easily disappear. Moreover, it may be held individually or collectively [2] [3].

Knowledge is not only obtained from outside but also from the interaction of the members of an organization. This means that intelligence can be propagated and accumulated to create substantial wisdom [4].

Knowledge process principle mentions that the most suitable resource to perform the operation is the person that knows all of the inputs of the operation. In other words, the action must be undertaken by whoever possesses the knowledge [5] [6].

2 The Methodology of Analytic Knowledge Process

The methodology of analytic knowledge processes (AKP) acts as a guide to assist with the integration of knowledge management and IS implementations [12]. The key tool is the table in figure 1, which summarizes data flows, information and knowledge inside the organization and the IS [7], after which come the different phases of the method.

			DEPARTAMENT RESOURCES
PROCESS - ACTIVITY - OPERATION	INPUTS Description OUPUT Description	FREQUENCY (F) EFFORT (E)	INPUTS (I) *Knowledge* OUTPUTS (O) *Transference* ACTION (A) *Who makes it* FREQUENCY x EFFORT (FxE) *Resource Occupation*

Figure 1: Analytic Knowledge Processes (AKP) Table.

2.1 Determine the process, activities and operations

Initially, the processes that the firm wishes to implement should be identified. Then, it must determine the activities that compose each process and finally, the operations of each activity [11].

The main premise is that an operation is the minimum unit of any task that a person or resource of a department can do when interacting with the SI. Each operation has one transaction or application in the SI. Furthermore, each operation has data inputs and outputs and information, which

may be difficult to identify correctly, but which it is essential to identify.

Moreover, the person who executes a transaction must possess the skills to enter the inputs of the operation and sufficient knowledge of the business process to enter the right inputs and to interpret the output.

2.2 Determine inputs, outputs and resources

According to the knowledge of data or information belonging to an operation's input, a resource can placed in one of the following categories: nothing, partially, or everything. "Nothing" means that the resource does not know the whole input. "Partially" means that resource sometimes knows the whole input or is also unaware of some of it. "Everything" means that the resource always knows the whole input.

In accordance with these categories, an operation can be classified as: attainable, ambiguous, or unattainable. When one or several resources have been categorized as "everything", the operation is classified as attainable. On the other hand, if no resource has been categorized as "everything", the operation is classified as ambiguous, and if all resources have been categorized as "nothing", the operation is classified as unattainable.

Every operation classified as unattainable or ambiguous is analyzed again, until it can be classified as attainable. Repeated analyses might be due to multiple causes and the solutions are arrived at on a case by case basis. They might for instance involve a supplier that does not facilitate the right information or information that is dispersed between different resources. An operation is considered attainable when at least one resource has the knowledge to do it.

The desired result is the conversion of implicit, explicit and individual knowledge into explicit and collective knowledge in a process that is referred to as externalization and combination [2] [3].

According to Knowledge process principle, where there are several resources categorized as 'everything' for an operation, the action is carried out by the resource that meets the first of the following conditions: higher authority in the company, prior completion of the operation, knowledge of the inputs for subsequent operations, or any other approaches that may be defined. Consequently, the selected resources are the best equipped ones in terms of their knowledge to perform the operation.

2.3 Estimate Resource Occupation

The occupation per operation of each resource can therefore be quantified, although this requires an estimation of the effort and the frequency per operation. Effort is the average monthly time required by a resource to complete the operation, and frequency is the average number of times per month an operation has to be performed.

The variables used to represent these parameters are as follows:

Effort is represented by variable E(i,j,k) and it refers to the average time (in hours) that is needed to carry out operation "i" of activity "j", and process "k".

Frequency is represented by variable F(i,j,k) and it refers to the number of times per month operation "i" of activity "j", and process "k" is performed.

The performance of an action is represented by variable D(l,i,j,k). Its value is "1", when department "l" is selected to perform operation "i" of activity "j", and process "k". Otherwise, its value is "0".

Available monthly hours for a human resource are represented by variable H. Its common value is 160 hours.

The number of departments is represented by variable N.

We have considered two possible types of organizations: departmental and functional. In the first, human resources are distributed into departments, and these execute the operations. Nevertheless, in the second type, the human resources execute the operations as a functional distribution.

In the case of a departmental organization, formula 1 allows the calculation of the amount of monthly human resources for a department "l", represented by variable HRD (l). Formula 2 works out the amount of monthly human resources in the organization.

$$HRD(l) = Integer\left\{\frac{\sum_{k}\sum_{j}\sum_{i} E(i,j,k) * F(i,j,k) * D(l,i,j,k)}{H}\right\} \quad (1)$$

$$R = \sum_{l=1\ldots N} HRD(l) \quad (2)$$

In case of a functional organization, formula 3 is used to determine the amount of monthly human resources.

$$R = Integer\left\{\frac{\sum_{k}\sum_{j}\sum_{i} E(i,j,k) * F(i,j,k)}{H}\right\} \quad (3)$$

The occupation of the resources for both cases is obtained by formula 4.

$$\frac{\sum_{k}\sum_{j}\sum_{i} E(i,j,k) * F(i,j,k)}{R * H} * 100\% \quad (4)$$

As a result, a departmental organization requires more human resources than a functional organization; it is nevertheless able to respond more quickly to calls for increased effort or frequency of operations. The functional structure implies fewer human resources, greater versatility and training, which is typical of small or emerging companies [8]. As the companies are growing the organization evolves towards the departmental type, although hybrids of both types can also exist.

2.4 Implementation of an Information System

Before starting up an IS, an AKP table is completed and all operations have to be attainable. According to the table, the authorizations are assigned to the transactions of the IS and the training for each human resource is distributed.

After starting the IS, the organization familiarizes itself with the new transactions. Moreover, in due course further improvements and initially unforeseen functionalities will be required.

Accordingly, improvements to the organization do not finish when the IS implementation begins. On the contrary, this is when KM and the AKP table must be used, above all to help reorganize and begin new IS implementations so that documentation remains updated, and knowledge continues to be explicit and collective [9] [12].

Consequently, KM has to begin, at the latest, in during the first phase of the IS implementation, although it is suggested that it should start before, because it allows the intangible capital in the next phases of the implementation to be anticipated and managed more efficiently.

The benefits of AKP are difficult to estimate, but several improvement rates [10] can be defined that compare the two situations, post-implementation and pre-implementation, which may focus on:

Fluency, decentralization and no personalization of the information and communication. For example: Reduction of internal telephone calls

New implementations motivated by the needs of external clients in relation to internal ones.

Implicit knowledge being converted into explicit tacit knowledge.

Information provided by IS that helps the monitoring of the objectives and goals of the company.

PROCESS: P01 - RETURNED GOODS		FREQUENCY (F) EFFORT (E)	MANAGEMENT	PURCHASES	SUPPLIES	COMMERCIAL	QUALITY
A01: VALIDATE THE RETURN	<== AMBIGUOUS ==>						
Op01: ANALYZE THE CLAIM	AMBIGUOUS						
INPUT =>	- CLIENT - MATERIAL / DESPATCH NOTE/ INVOICE / ORDER - CAUSE OF RETURN		NOTHING	NOTHING	PARTIALLY	NOTHING	PARTIALLY
OUTPUT<=	- ACCEPTANCE / REJECTION OF THE RETURNED GOODS - CLAIMABLE AGAINST THE SUPPLIER (YES OR NO)		YES	YES	NO	YES	YES
Op02: GENERATE REPORT ON RETURNED GOODS	AMBIGUOUS						
INPUT =>	- CLIENT - MATERIAL / DESPATCH NOTE / INVOICE / ORDER - CAUSE OF RETURN - OTHER INFORMATION		NOTHING	PARTIALLY	PARTIALLY	PARTIALLY	PARTIALLY
OUTPUT<=	- N° REPORTS GENERATED IN THE I.S.		YES	YES	YES	YES	YES
A02: AUTHORIZE THE REFUND	==> ATTAINABLE <==						
Op01: ISSUE REFUND AUTHORISATION	ATTAINABLE		DO IT				
INPUT =>	- N° OF AUTHORISATIONS		EVERYTHING	EVERYTHING	EVERYTHING	EVERYTHING	EVERYTHING
OUTPUT<=	- N° OF AUTHORISATIONS PAID OUT		YES	YES	YES	YES	YES
Op02: ACCEPTANCE OF RETURNED GOODS	ATTAINABLE				DO IT		
INPUT =>	- N° OF ACCEPTANCES OF RETURNED GOODS		EVERYTHING	EVERYTHING	EVERYTHING	EVERYTHING	EVERYTHING
OUTPUT<=	- N° OF RETURNED GOODS		YES	YES	YES	YES	YES
Op03: ISSUE REFUND PAYMENT	ATTAINABLE				DO IT		
INPUT =>	- N° OF PAYMENTS CONFIRMED - ENTER CONFIRMATION OF RETURNED GOODS		PARTIALLY	PARTIALLY	EVERYTHING	PARTIALLY	PARTIALLY
OUTPUT<=	- ISSUE PAYMENT		YES	YES	YES	YES	NO
A03: LOOK FOR AN ALTERNATIVE	==> ATTAINABLE <==						
Op01: TO SELL TO ANOTHER CLIENT	ATTAINABLE					DO IT	
INPUT =>	- CLIENT + ORDER - RETURNED GOODS		PARTIALLY	PARTIALLY	EVERYTHING	EVERYTHING	PARTIALLY
OUTPUT<=	- DESPATCH ADVICE, INVOICE		YES	YES	YES	YES	YES
Op02: RETURN GOODS TO THE SUPPLIER	ATTAINABLE			DO IT			
INPUT =>	- RETURNED GOODS - SUPPLIER		PARTIALLY	EVERYTHING	PARTIALLY	PARTIALLY	EVERYTHING
OUTPUT<=	- ACCEPTANCE / REJECTION SUPPLIER		YES	YES	YES	YES	YES
Op03: WRITE OFF AS WASTE	ATTAINABLE						DO IT
INPUT =>	- RETURNED GOODS		EVERYTHING	EVERYTHING	EVERYTHING	EVERYTHING	EVERYTHING
OUTPUT<=	- WASTE MATERIAL		YES	YES	YES	YES	YES

Figure 2: Example of AKP for Returned Goods Process

2.5 Example of AKP method

Let us suppose that we wish to implement a quality IS module. We shall imagine four processes related to this module inside the organization that must first of all be identified: returned products, internal quality problems, quality inspections, and material testing (figure 2). These are then coded, to facilitate their management.

The firm has a departmental organization, such that the resource groups are organized into the following departments: Management, Purchases, Provisioning, Commercial Relations and Quality.

This example focuses on process 'P01 – Returned Goods' that consists of inverse logistics due to quality problems in a sold product. This process is composed of three activities: validation of the return, authorization of the refund, and search for an alternative product. These too are codified.

Next, the operations for each activity are identified, as may be seen in figure 2. Finally, the inputs and outputs of each operation are determined and the knowledge for the input for each resource is categorized. At this point, the person to perform each operation is also indicated, as previously explained

However, two operations have been categorized as ambiguous (figure 2), as there is no resource that always knows the inputs of the operation. In these cases, both operations are analyzed with the key-users, and the conclusion is that some clients do not provide all of the required information because they generally do have the number of the lot or the delivery or dispatch note. To overcome this, all clients are asked to enclose this information when they return goods.

After having reengineered the system so that each operation is attainable, and estimated the effort and monthly frequency of the operations, the AKP table shown in figure 3 is complete.

PROCESS: P01 - RETURNED GOODS		FREQUENCY (F) EFFORT (E)	MANAGEMENT	PURCHASES	SUPPLIES	COMMERCIAL	QUALITY
A01: VALIDATE THE RETURN	==> ATTAINABLE <==						
Op01: ANALYZE THE CLAIM	ATTAINABLE						*F x E = 40*
INPUT =>	- CLIENT - MATERIAL / DESPATCH NOTE/ INVOICE / ORDER - CAUSE OF RETURN	F = 20	NOTHING	NOTHING	PARTIALLY	NOTHING	*EVERYTHING*
OUTPUT<=	- ACCEPTANCE / REJECTION OF THE RETURNED GOODS - CLAIMABLE AGAINST THE SUPPLIER (YES OR NO)	E = 2	YES	YES	NO	YES	YES
Op02: GENERATE REPORT ON RETURNED GOODS	ATTAINABLE						*F x E = 9*
INPUT =>	- CLIENT - MATERIAL / DESPATCH NOTE / INVOICE / ORDER - CAUSE OF RETURN - OTHER INFORMATION	F = 18	NOTHING	PARTIALLY	PARTIALLY	PARTIALLY	*EVERYTHING*
OUTPUT<=	- N° REPORTS GENERATED IN THE I.S.	E = 0,5	YES	YES	YES	YES	YES
A02: AUTHORIZE THE REFUND	==> ATTAINABLE <==						
Op01: ISSUE REFUND AUTHORISATION	ATTAINABLE		*F x E = 0,9*				
INPUT =>	- N° OF AUTHORISATIONS	F = 18	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*
OUTPUT<=	- N° OF AUTHORISATIONS PAID OUT	E = 0,05	YES	YES	YES	YES	YES
Op02: ACCEPTANCE OF RETURNED GOODS	ATTAINABLE				*F x E = 9*		
INPUT =>	- N° OF ACCEPTANCES OF RETURNED GOODS	F = 18	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*
OUTPUT<=	- N° OF RETURNED GOODS	E = 0,5	YES	YES	YES	YES	YES
Op03: ISSUE REFUND PAYMENT	ATTAINABLE				*F x E = 1,8*		
INPUT =>	- N° OF PAYMENTS CONFIRMED - ENTER CONFIRMATION OF RETURNED GOODS	F = 18	PARTIALLY	PARTIALLY	*EVERYTHING*	PARTIALLY	PARTIALLY
OUTPUT<=	- ISSUE PAYMENT	E = 0,1	YES	YES	YES	YES	NO
A03: LOOK FOR AN ALTERNATIVE	==> ATTAINABLE <==						
Op01: TO SELL TO ANOTHER CLIENT	ATTAINABLE					*F x E = 10*	
INPUT =>	- CLIENT + ORDER - RETURNED GOODS	F = 10	PARTIALLY	PARTIALLY	*EVERYTHING*	*EVERYTHING*	PARTIALLY
OUTPUT<=	- DESPATCH ADVICE, INVOICE	E = 1	YES	YES	YES	YES	YES
Op02: RETURN GOODS TO THE SUPPLIER	ATTAINABLE			*F x E = 6*			
INPUT =>	- RETURNED GOODS - SUPPLIER	F = 4	PARTIALLY	*EVERYTHING*	PARTIALLY	PARTIALLY	*EVERYTHING*
OUTPUT<=	- ACCEPTANCE / REJECTION SUPPLIER	E = 1,5	YES	YES	YES	YES	YES
Op03: WRITE OFF AS WASTE	ATTAINABLE						*F x E = 1*
INPUT =>	- RETURNED GOODS	F = 4	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*	*EVERYTHING*
OUTPUT<=	- WASTE MATERIAL	E = 0,25	YES	YES	YES	YES	YES
Total Occupation of the Resources (F x E)		77,7	0,9	6	10,8	10	50

Figure 3: Example of AKP which have every operation attainable

.

PROCESS: P01 - RETURNED GOODS		FREQUENCY (F)	EFFORT (E)	RESOURCE THAT CARRIES OUT THE OPERATION OCCUPATION OF THE RESOURCE (HOURS/MONTH)
INPUT (I)	OUTPUT (O)			
A01: VALIDATE THE RETURN				
1 Op01: ANALYZE THE CLAIM - CLIENT - MATERIAL / DESPATCH NOTE/ INVOICE / ORDER - CAUSE OF RETURN	- ACCEPTANCE/REJECTION OF THE RETURNED GOODS - CLAIMABLE AGAINST THE SUPPLIER (YES OR NO)	20	2	QUALITY (40 H/month)
2 Op02: GENERATE REPORT ON RETURNED GOODS - CLIENT - MATERIAL / DESPATCH NOTE/ INVOICE / ORDER - CAUSE OF RETURN - OTHER INFORMATION	- Nº OF REPORTS IN THE I.S.	18	0,5	QUALITY (9 H/month)
A02: AUTHORIZE THE REFUND				
1 Op01: ISSUE REFUND AUTHORIZATION - Nº OF AUTHORISATIONS	- Nº OF AUTHORISATIONS PAID OUT	18	0,05	MANAGEMENT (0.9 H/month)
2 Op02: ACCEPTANCE OF RETURNED GOODS - Nº OF ACCEPTANCES OF RETURNED GOODS	- RETURNED GOODS	18	0,5	SUPPLIES (9 H/month)
3 Op03: ISSUE REFUND PAYMENT - Nº OF PAYMENTS CONFIRMED - ENTER CONFIRMATION OF RETURNED GOODS	- ISSUE PAYMENT	18	0,1	SUPPLIES (1.8 H/month)
A03: IDENTIFY AN ALTERNATIVE				
1 Op01: SELL TO ANOTHER CLIENT - CLIENT + ORDER - RETURNED GOODS	- DESPATCH NOTE, INVOICE	10	1	COMMERCIAL (10 H/month)
2 Op02: RETURN GOODS TO THE SUPPLIER - RETURNED GOODS - SUPPLIER	- ACCEPTANCE / REJECTION SUPPLIER	4	1,5	PURCHASES (6 H/month)
3 Op03: WRITE OFF AS WASTE - RETURNED GOODS	- WASTE	4	0,25	QUALITY (1 H/month)
Total Occupation of the Resources (F x E)				5 DEPARTMENTS (77,7 H/month)

Figure 4: Assignment and Occupation of the Resources

In figure 4, you can observe the identified resources that carry out the actions on which basis it is possible to identify the one that should receive the training and the authorizations

By applying the methodology to the other processes, the occupation of the resources can be calculated. Also a comparison between a departmental and functional organization can be made. When the AKP table has been developed, each operation has to be documented.

Moreover, defined improvements rates help compare the pre- and post-implementation situations:

Number of available parameters of quality implemented: an increase from 1 to 5 parameters.

Number of monthly internal telephone calls: a decrease by 10%.

Number of documents available on quality: increase from 2 to 6.

Number of new implementations in the process 'returned goods' as a percentage of total new implementations in the quality module: 25%.

QUALITY MODULE PROCESS	Occupation (H/month)	MANAGEMENT	PURCHASES	SUPPLIES	COMMERCIAL RELATIONS	QUALITY
P01: RETURNED GOODS	77,7	0,9	6	10,8	10	50
P02: INTERNAL QUALITY PROBLEMS	92	1	6	2	3	80
P03: QUALITY INSPECTIONS	105	1	1	1	2	100
P04: MATERIAL SPECIFICATION TESTS	206	1	2	1	2	200
Total Occupation (H/month)	480,7	3,9	15	14,8	17	430

SIZE OF EACH ORGANIZATION (1 Person => 160 Hours/month)			MANAGEMENT	PURCHASES	SUPPLIES	COMMERCIAL RELATIONS	QUALITY
FUNCTIONAL ORGANIZATION	NUMBER OF PEOPLE	3	0,02	0,09	0,09	0,11	2,69
	OCCUPATION LEVEL	100%	100%	100%	100%	100%	100%
DEPARTMENTAL ORGANIZATION	NUMBER OF PEOPLE	7	1	1	1	1	3
	OCCUPATION LEVEL	43%	2%	9%	9%	11%	90%

Figure 5: Size of the Organization

Seven people would be required for a departmental organization, representing an occupation rate of 43% (Figure 5). On the other hand, in the case of a functional organization, only three people would be required with a 100% occupation rate.

Finally, we cannot forget that in a parallel process to its preparation, the AKP table on organizational processes requires the latter processes to be fully documented, and also requires periodic revisions of the estimated values for effort and frequencies in order to validate or to reorganize the resources.

3 Conclusions

The implementation of an IS does not follow a typical life cycle with a beginning and an end, but is rather a continual process. This study has provided evidence in support of initiating KM prior to or at the same time as an IS, as it can provide essential support to its implementation or its reengineering.

The development of Analytic Knowledge Processes is shown to contribute to the completion of IS implementation targets. It is able to clarify the required size of the organization, to define the training and the authorizations of each resource, to document the information inside the organization, and to fix a starting point for future implementations or reengineering. For a company, the initiation of KM in conjunction with IS can lead to organizational synergies and benefits that are difficult to ignore.

4 References

[1] Gable, G., Scott, E. and Davenport, T. (1998) "Cooperative ERP Life-cycle Knowledge Management", Proceedings of the Ninth Australasian Conference on Information Systems, pp 227–240.

[2] Nonaka, I. (1994) "A dynamic theory of organizational knowledge creation", Organization Science, Vol.5, No.1, pp 14-37.

[3] Nonaka, I. and Hirotaka, T. (1995) "The Knowledge-Creating Company" Company, Oxford University Press, New York.

[4] Chih-Hung, T., Ching-Liang, C. and Lieh, C. (2006) "A Case Study of Knowledge Management Implementation for Information Consulting Company", International Journal of the Compute, The Internet and Management, Vol. 14, No. 3, pp 60–78.

[5] Sáiz, L. and Manzanedo, M. (2003) "Claves para la implantación con éxito de la gestión del conocimiento", Boletín de Estudios Económicos, University of Deusto. Vol. LVIII, No. 180.

[6] Roblek, Matjaz, Tomaz Kern, and Maja Zajec. (2013) "Knowledge Management of Knowledge Intensive Business Processes with PKA Method." Active Citizenship by Knowledge Management & Innovation: Proceedings of the Management, Knowledge and Learning International Conference 2013. ToKnowPress

[7] Kogut, B., and Zander, U. (1993) "Knowledge and the firm and the evolutionary theory of the multinational corporation". Journal of International Business Studies, vol. 24, n. 4 pp 625-645

[8] Purvis, R., Sambamurthy, V. and Zmud, R. (2001) "The assimilation of knowledge platforms in organizations: an empirical investigation". Organization Science, No.12, pp 117-135

[9] McGinnis, T.C. and Huang, Z. (2004) "Incorporation of Knowledge Management into ERP continues Improvement: A research Framework", Issues in Information Systems. Vol. V, No. 2

[10] Robertson, J. (2003). Metrics for knowledge management and content management. KM Column.

[11] Parthasarathy, S., & Sharma, S. (2014). Determining ERP customization choices using nominal group technique and analytical hierarchy process. Computers in Industry, 65(6), 1009-1017.

[12] Anand, A., Kant, R., Patel, D. P., & Singh, M. D. (2015). Knowledge management implementation: a predictive model using an analytical hierarchical process. Journal of the Knowledge Economy, 6(1), 48-71.

[13] Seethamraju, R., & Sundar, D. K. (2013). Influence of ERP systems on business process agility. IIMB Management Review, 25(3), 137-149.

[14] Chaabouni, A., & Ben Yahia, I. (2014). Contribution of ERP to the decision-making process through knowledge management. Journal of Decision Systems, 23(3), 303-317.

Forecasting erratic demand of medicines in a public hospital: A comparison of artificial neural networks and ARIMA models

A. Molina
Instituto Tecnológico de Culiacán, Culiacán, Sinaloa, México

B. Ponte, J. Parreño, D. De la Fuente
Universidad de Oviedo, Gijón, Asturias, España

J. Costas
Polytechnic Institute of Viana do Castelo, Viana do Castelo, Portugal

Abstract- *Demand planning is the process that helps in making decisions about inventory, in order to anticipate future demands from historical data. In an inventory with hundreds of items, a significant amount of these is subject to erratic demand. For these products, accurate forecasts are essential. This research work aims to show a comparison of ARIMA models and artificial neural networks for forecasting the demand medicines with an erratic nature.*

Keywords: ARIMA, Artificial Neural Networks, forecasting, erratic demand.

1. Introduction

The right to have timely and good quality health services is a prerequisite for maintaining proper health conditions of humanity requirement. One aspect of great importance in the health care is the access to medicines, because they are a basic input to healthcare process.

The *Results of the global partnership to achieve the Millennium Development Goals* report, from the World Health Organization (WHO), highlights the existence of great inequalities in regard to the availability of medicines in both public and private sector [6]. For this reason, the research focused on enhancing the hospital supply chain has been of great importance.

The hospital supply chain is more complex than industrial supply chains; since there are some hospital supplies, such as medicines, that can be considered critical for healthcare. Hence, its replenishment should have a high level of response. In addition to the requirement of quick replenishment, it has to take care of efficient management, since many of them are exposed to obsolescence [7].

Regarding hospital logistics cost, a major cost is the pharmaceutical one. There are 4 types of costs associated with the pharmaceutical inventory; namely, the acquisition costs, the ordering costs, the carrying costs and the stock-out costs [1].

In order to improve inventory management, demand forecasting plays a key role. This fact is highlighted in those medicines with an erratic pattern (i.e. those which exhibit large and unpredictable fluctuations over time), which tend to have a high acquisition cost. In this sense, an optimal control of these medicines will result in significant savings in the aforementioned costs.

Consequently, an adequate demand forecasting mechanism is indispensable. It contributes to the improvement of the inventory management; therefore, many research papers analyze the behavior of time series by means of quantitative forecasting models [4].

This paper aims to compare the utilization of statistical and AI-based mechanisms to forecast the erratic demand of medicines. To illustrate the former, we employ the widely used ARIMA models; while artificial neural networks have been developed to analyze the suitability of AI techniques.

The organization of this paper is as follows: Firstly, we introduce the ARIMA models. Secondly, we show the application of artificial neural networks for demand forecasting. Next, we analyze a specific case study. Finally, we conclude by revisiting our goals.

2. ARIMA Models

The autoregressive integrated moving average models (ARIMA) are the most important and used for time series analysis, these were introduced by Box & Jenkins in 1970. The ARIMA models are considered autoregressives because the variable to predict will be explained as a linear function of its observations in the past and random errors [8].

This type of modeling is useful when there is not a model which satisfactorily explains the relation between the forecast variable and other explicative variables. The ARIMA models are characterized for being flexible, this is, they represent different types of series, such as autoregressive (AR), moving averages (MA) and autoregressive moving averages (ARMA).

A non-seasonal ARIMA model can be classified as an "ARIMA (*p,d,q*)" model, where:

p *is the number of autoregressive terms;*
d *is the number of nonseasonal differences; and*
q *is the number of moving-average terms.*

The methodology Box-Jenkins includes three steps 1) model identification, 2) parameter estimation and 3) model checking.

Assuming for the moment that there is no seasonal variation, the objective of the model identification step is to select values of *d, p* and *q* in the *ARIMA (p,d,q)* model. The parameters estimated in the second step are a_i and b_i, finally in the model checking, it verified if the estimated model conforms satisfactorily to the observed series. If the estimation is inadequate, we have to return to step one and attempt to build a better model.

3. Artificial Neural Networks

The artificial neural networks (RNA) are techniques of computational modeling for complex problems. This discipline of artificial intelligence has emerged recently and has big acceptance in different areas [2].

One of the big areas of application of artificial neural networks is time series forecasting. In contrast to the quantitative forecasting models, the RNA have the learning ability by examples and it can identify the relations between the data, even though these are difficult to describe [9].

There are six important aspects to consider in the ANNs creation:

1. Network architecture

There are different network architecture, among them, should be highlighted the multilayer perceptron, because it has proven to be an effective alternative to quantitative techniques [3].

The multilayer perceptron, also named backpropagation network consist of inputs and outputs system with processing units knows as neurons or nodes (Fig.1). The neurons in each phase are interconnecting by forces called synaptic weights and output signals that are a function of the sum of the inputs to the neuron. Additionally to the processing neuron there is an slant neuron connecting to each processing unit of the hidden layer and output layer [5].

Other structure of ANN is the basis radial function (RBF), an RBF network is designed with neurons in the hidden layer activated by radial non-linear functions with its own gravitational centers, while the output layer is activated by linear functions.

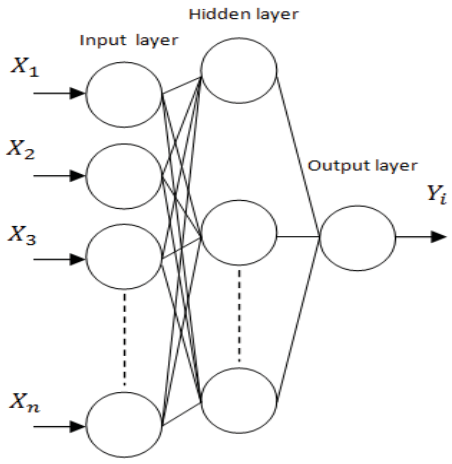


Figure 1. Multilayer perceptron neural network structure.

For a forecasting problem, the inputs to an ANN are usually the independent variables. The functional relationship estimated by the ANN can be written as:

$$y_i = f(x_1, x_2, x_3, \ldots x_n)$$

Where $x_1, x_2, x_3, \ldots x_n$ are independent variables and y_i is the dependent variable.

2. Network configuration

The network configuration is determined by:

- The number of input nodes.
- The number of hidden nodes and hidden layers.
- The number of output nodes.

3. Activation function

The activation function determines the relationship between input and outputs of a node. The commons activation functions are: the sigmoid function, the hyperbolic tangent function, the sine or cosine function and the linear function.

4. Training algorithm

The neural artificial training is an unconstrained non linear minimization problem in which are weights of a network are iteratively modified to minimize the overall mean or total squared error between the desired and actual output values for all output nodes over all input patterns. The training algorithm most used for the multilayer perceptron is the backpropagation.

5. Data normalization

The data normalization can be realized by four methods:

- Along channel normalization
- Across channel normalization
- Mixed channel normalization
- External normalization

The choice of the normalization methods generally depends on the composition of the input layer.

6. Training sample and test sample

Training and test sample are required for the execution of artificial neural network. The training sample is used for ANN model development and the test sample is adopted for evaluating the forecasting ability of the model.

7. Performance metrics

The most important measure of the results of a neural network is the accuracy in forecasting. There are different measures of accuracy in the forecasting; in this work we measure the Mean Absolute Deviation (MAD), the Mean Squared Error (MSE) and the Root Mean Squared Error (RMSE).

4. Medicine demand forecast

In this study, the demand of two drugs in a public hospital will be analyzed. Demand data of these medicines corresponds the period 2012 - 2015 and were obtained monthly, resulting in series of 48 data.

The selected medicines have the property of an erratic demand. That is, they show variations over time for example a zero demand and then large quantities of demand.

ARIMA models are considered for the forecast demand, they are executed in Statgraphics software and the application of neural networks is executed in SPSS software.

This section shows the time series graphics, residual autocorrelations functions and residual partial autocorrelations for the 2 medicines, which are identified by codes 5250 and 4250.

As a first stage of analysis the forecasts models were executed in Statgraphics and the best model was selected according to the Akaike criterion. The models selected were ARIMA (2, 0, 4) for 5250 medicine and ARIMA (0, 1, 3) for 4250 medicine. The figures 8 and 9 show the real demand and the forecast demand.

After the forecasts with ARIMA models, the forecasts with neural networks were realized. For that it was used the multilayer perceptron and the radial basis function. The structure of the neural network is similar to that showed previously in Figure 1, the inputs of the hidden layer are the demand in different periods and the output is the forecast demand.

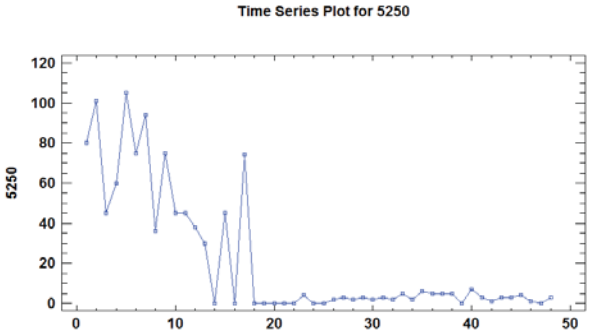


Figure 2. Time series 5250.

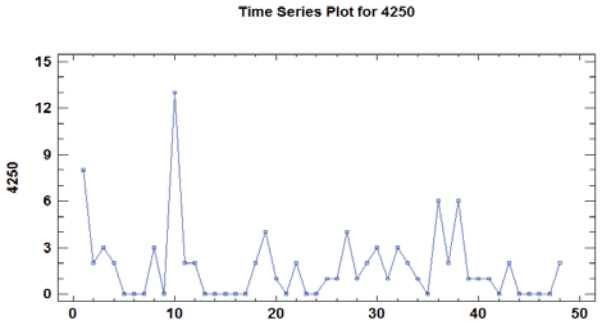


Figure 3. Time series 4250.

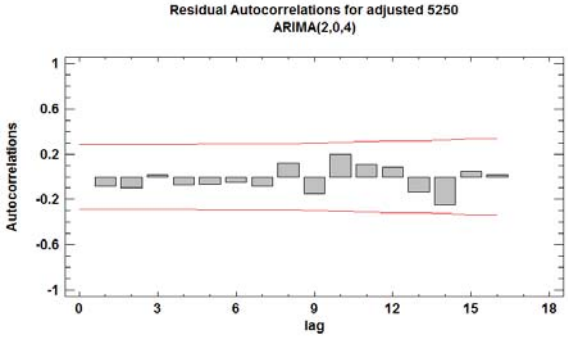


Figure 4. Residual autocorrelations function 5250.

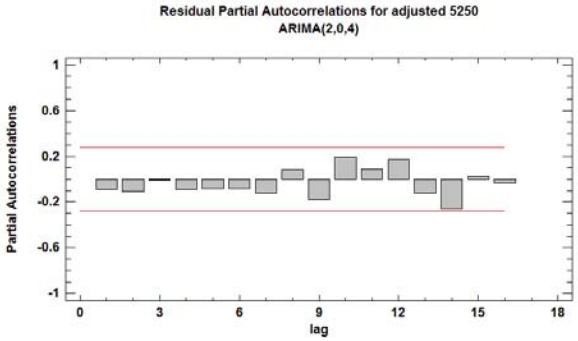


Figure 5. Residual partial autocorrelations 5250.

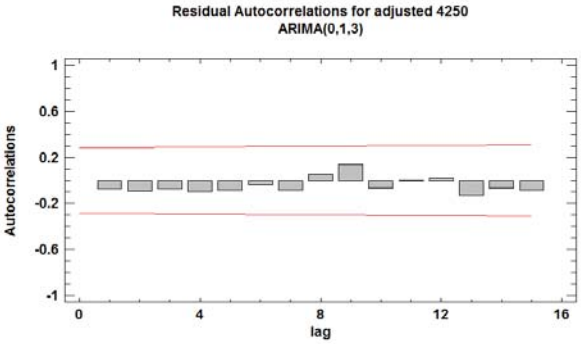


Figure 6. Residual autocorrelations function 4250.

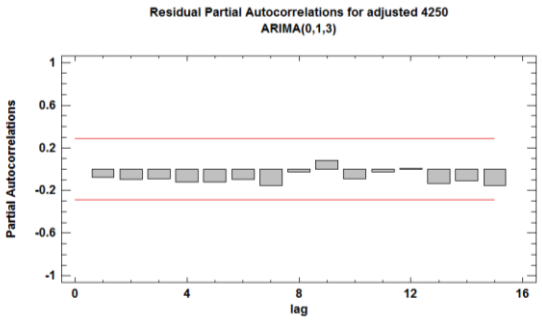


Figure 7. Residual partial autocorrelations 4250.

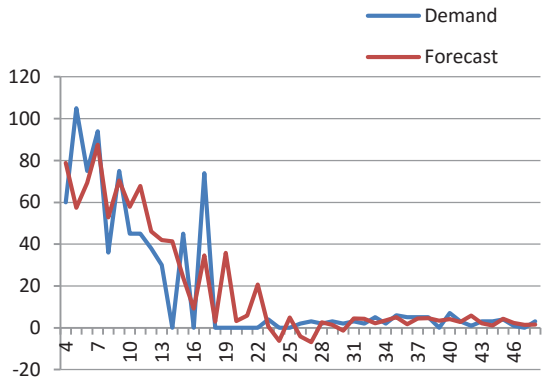


Figure 8. Demand and Forecasts graphic 5250.

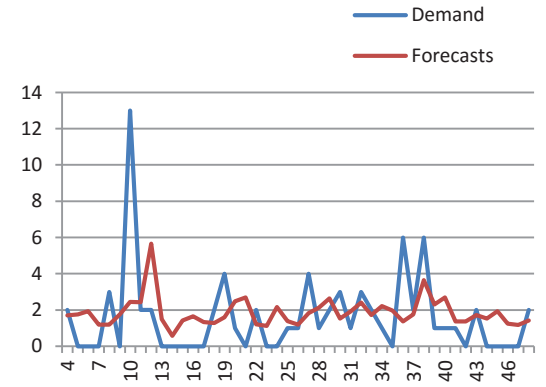


Figure 9. Demand and Forecasts graphic 4250.

Figures 10 and 11 show the real demand and the forecast demand.

In order to evaluate the forecast demand with ARIMA models and artificial neural networks we calculate the Mean Absolute Deviation (MAD), the Mean Squared Error (MSE) and the Root Mean Squared Error (RMSE). The results are in tables 1 and 2.

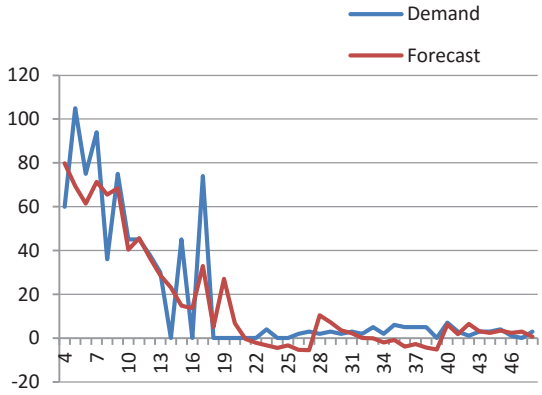


Figure 10. Demand and Forecasts graphic 5250 with ANN.

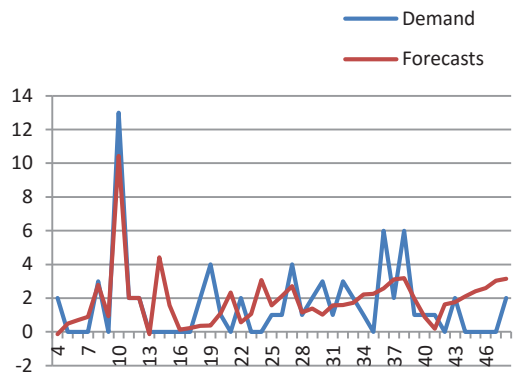


Figure 11. Demand and Forecasts graphic 4250

TABLE 1. Results of ARIMA model and ANN for 5250.

Model	MAD	MSE	RMSE
ARIMA (2,0,4)	8.952	216.770	14.723
ANN	8.783	180.443	13.432

TABLE 2. Results of ARIMA model and ANN for 4250.

Model	MAD	MSE	RMSE
ARIMA (0,1,3)	1.574	5.097	2.257
ANN	1.367	3.082	1.755

In both cases, the artificial neural networks mechanism showed better results in terms of the performance measure.

5. Conclusion

This paper presents an analysis of forecasting demand of medicines with erratic demand. These medicines are characterized by a high acquisition cost and a sporadic demand over time. These special chatacteristics cause a high precision requirement in the demand forecast.

In this regard, we analyze two different techniques, ARIMA models and AI techniques, such as MLP and RBFs ANNs. Our results show that for this type of medicines, the ANNs offer a better performance than the ARIMA models.

6. References

[1] Ali, A. (2011). Inventory Management in Pharmacy Practice: A Review of Literature. *Pharmacy Practice*.

[2] Basheer I, Hajmeer M (2000). Artificial neural networks: fundamentals, computing, design and application. *Journal of Microbiological Methods.*

[3] Gardner M, Dorling S. (1998). Artificial Neural Network (The multilayer perceptron). *Atmospheric environment.*

[4] Gheyas I, Smith L (2011). A novel neural network ensemble architecture for time series forecasting. *Neurocomputing.*

[5] Kaastra I, Boyd M (1996). Designing a neural network for forecasting financial and economic time series. *Neurocomputing.*

[6] United Nations. (2008). *United Nations*. Retrieved on May 1, 2015, of Results of the global partnership to achieve the Millennium Development Goals.: http://www.un.org/

[7] Romero, A. (2013). Managing Medicines in the Hospital Pharmacy: Logistics Inefficiencies. *Proceedings of the World Congress on Engineering and Computer Science* .

[8] Shumway R, Stoffer D (2011). Time Series Analysis and its applications. London: Springer.

[9] Zhang G, Patuwo E, Hu M (1997). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting.*

A vision of industry 4.0 from an artificial intelligence point of view

Dopico, M.[1], **Gomez, A.**[1], **De la Fuente, D.**[1], **García, N.**[1], **Rosillo, R.**[1], **Puche, J.**[1]
[1]Administracion de Empresas, University of Oviedo, Gijón, Asturias, Spain

Abstract - *During the first years of the so called fourth industrial revolution, main attempts that tried to define the main ideas and tools behind this new era of manufacturing, always end up referring to the concept of smart machines that would be able to communicate with each and with the environment. In fact, the defined cyber physical systems, connected by the internet of things, take all the attention when referring to the new industry 4.0. But, nevertheless, the new industrial environment will benefit from several tools and applications that complement the real formation of a smart, embedded system that is able to perform autonomous tasks. And most of these revolutionary concepts rest in the same background theory as artificial intelligence does, where the analysis and filtration of huge amounts of incoming information from different types of sensors, assist to the interpretation and suggestion of the most recommended course of action. For that reason, artificial intelligence science suit perfectly with the challenges that arise in the consolidation of the fourth industrial revolution*.

Keywords: Industry 4.0. Artificial Intelligence, Big Data, embedded systems.

1 Introduction

Industry and industrial processes are continuously evolving. The needs for competitive advantages in manufacturing have been historically the engine for the development of advanced and cost effective new mechanisms to manufacture. In this effort, and since the beginning of industrialization, from time to time, a technological leap takes place that revolutionizes the concept of industrial production, being refer to as industrial revolutions: First industrial revolution took place in the field of mechanization and steam engines; second industrial revolution was based in the intensive use of electrical energy and mass production; and third industrial revolution was founded in the IT environment and the widespread of digitalization.

The actual scenario of industrial production is stigmatic with a low predisposition for changes and unexpected situations. Production chains are characterized by static lines (with predefined sequences), which are hard to reconfigure to make new product variants. Today, regular situations of troubleshooting decrease or even stop production and lead to demotivation of employees. In conclusion, current industrial activity is rigid and difficult to change; innovations can hardly be afforded, reduction of raw material prices generally lowers quality and can increase process costs and, in consequence, profit margins decline constantly. There is also a great dependence on the in-house knowledge base, so when it drops, improvisation increase and development times get longer. Production systems, although strongly automated, do not possess a self-conscious core to learn and act without constant human monitoring. All these inconvenient points to the need for the next great revolution in industrial manufacturing, which will lead to several enhance attributes in comparison with actual production activities, like it is shown in next table, and which is starting to be referred to as industry 4.0[1].

For this new revolutionary industrial age, there is a concept that keeps being repeated, which can be narrowed into the self-consciousness of technology. An attribute that is directly linked with what artificial intelligence (AI) aims to achieve: The creation of systems that can perceive their environment and, consequently, can take action towards increasing the chances of success.

But the trigger for the new industrial revolution is based in two great pillars; the already described fixed industrial scenario that is not ready for unexpected changes in production, which adds up with another reality, which is that the forms of consumption are changing. Nowadays, producers want to focus into almost individualized demands in their efforts to reach every potential client, hence tomorrow's industry will need to provide a dynamic production line, where not only products are made, but a combination of product + service is offered to gain advantage against their competitors and driving production to constantly changing environments [2]. To achieve this, the degree of automation should move to the next level, where sensitive computing, taking information from the environment, should be able to predict next steps in production with barely none interaction with the user, in the same way artificial intelligence can work.

For that reason, under the intention to adjust to more flexible production schemes and increase the ability to compete, it can be said that we are in the prelude of a new paradigm shift in industrial production: Every step in the industrial process generates data (energy consumption, speed, power, weight

etc). Hence, once established an advanced digitalized network in factories that monitors this information, next logical step leads to the combination of internet technologies and "smart" objects to interpret this huge flow of information towards an advanced form of manufacturing, where, for example, it would be possible to foresee the need for preventive actions and adapt production before it happens. Advanced manufacturing technology is then a term that refers to a set of highly flexible, data-enabled, cost-efficient processes. In other words, the implementation of modular and efficient manufacturing systems in scenarios where products control their own manufacturing process, embedded in a futuristic environment where machines, products, humans and systems are able to communicate with each other and make decisions related with the process itself, adapting to new situations, being able to automatically detect anomalies or requirements, and acting in consequence. An example of what this fourth industrial revolution is planning to achieve in manufacturing environments, can be seen in a common object like the "airbag" from vehicles, which is one of the first developed systems that behave as a self-conscious entity; that in front of a sudden and new problem (a crash), is able to detect, by themselves, the need for actuation. And precisely the advance in fields like artificial intelligence, nanotechnology, robotics or additive manufacturing are the starting point for the development of more examples of flexible and adjustable operation of machines.

Under this hypothesis, the term industry 4.0 (or fourth industrial revolution) would not only imply a technological change, but versatile organizational implications as well. As a result, a change from product- to service-orientation is sought and expected: manufacturing and service industry will become complementary, encouraging a new form of production that some authors have labeled as servitization: "Servitization is the strategic innovation of an organization´s capabilities and processes to shift from selling products, to selling an integrated product and service offering that delivers value in use. Here the market goal of manufacturers is not one-time product selling, but continuous profit from customers by total service solution, which can satisfy unmet customers´ needs" [3].

Although complete development and employment of these technologies are still years ahead of us, first steps have already been taken towards the implementation of some of these measures, in order to achieve a more intelligent and efficient way of developing different industrial objectives. The journey to a flexible, heterogeneous, decentralized, standardized, and self-aware production system has started, and, as consequence, so have all the revolutionary aspects associated with industry 4.0.

This terminology, or in its defect the term in German "Industrie 4.0", has been embraced by the German industry (Hannover fair 2011), one of the main precursors of this new technological shift, as part of the High-Tech strategy 2020 action plan. Anyhow, a variety of different terms are used around the world to describe the phenomenon of Industrie 4.0. Some countries like Japan seem to have maintained "industry 4.0", but others like the U.S. have defined terms like the "Internet of Everything". Nomenclature like "Smart Production", "Smart Manufacturing", "Smart Industry" or "Smart Factory" are used in Europe, China and also the U.S. to refer specifically to digital networking of production to create smart manufacturing systems.

In a different field of interpretation, the terms "Advanced Manufacturing" [4] or "Predictive Manufacturing" [5] embrace a broader spectrum of modernization trends in the manufacturing environment. Advanced manufacturing is, by definition: "a family of activities that depend on the use and coordination of information, automation, computation, software, sensing, and networking, and/or make use of cutting edge materials and emerging capabilities enabled by the physical and biological sciences, for example nanotechnology, chemistry, and biology. It involves both new ways to manufacture existing products, and the manufacture of new products emerging from new advanced technologies" [4].

Individual companies like General Electric also invented their own name when they launched in 2012 a broad-based initiative named "Industrial internet", with applications in several areas, but where manufacturing only plays a minor role. Others like Bosch have been using "connected manufacturing" to emphasize precisely the importance in manufacturing aspects.

Anyway, independently of the specific given name, all these proposals are driven towards the need for the construction of an automated, self-conscious, interconnected, heterogeneous, embedded system, that will push manufacturing into the next level.

2 Main challenges of industry 4.0

The main message that can be extracted from the previous paragraphs, is that the basic idea of industry 4.0 rests in the combination of hardware and software devices, constructing a "Smart factory" where humans, machines and resources communicate with each other and work collaboratively; building more complex networks that, on the other hand, will reduce efforts. Precisely, artificial intelligence structure will assist in the creation of this networks, doting machines with the capability to learn, reason and act, basing on the information gathered during the industrial process. The required changes towards those smart factories are then defined by two development directions: Changes in the operative conditions themselves towards more flexible and decentralized processes; and changes in the technological means to achieve this objective [6].

Moreover, every change is boost by the main challenges that this new revolution has to face [7, 8, 2, 9, 4]:

1. Energy and resource efficiency as decisive competition factors.

2. Reduce time-to-market by shorter innovation cycles of complex products and larger data volumes.

3. Increase flexibility towards an almost individualized mass production in the middle of high productivity and volatile markets.

4. Gradually upgrade of existing infrastructure, connecting the different embedded systems by different mechanisms (M2M, wireless etc.), towards a constant self-prognosis, self-configuration, self-decision system.

These changes involve obvious benefits under the influence of "smart environments", like for example: processes will be more quickly (autonomous collision avoidance), more precise (nanotolerance manufacturing), more automated (analysis and determination of best suitable specifications by artificial intelligence), safer for workers (smart robots working in dangerous or inaccessible environments), more coordinated (automated traffic control), more efficient (energy sustainability), and maintaining a more collaborative relation between machines and humans (assistive technologies).

To achieve this goal, there are different factors that need deep consideration during this transition:

- Individualization: As it was said before, future production chains will drive to what years ago sounded as an utopist idea: Efficient production in batch sizes of 1 [7, 8, 10, 11, 2, 12].

- Flexibility not only during the process itself, but with simulation tools that can assure the feasibility and security of every change [7, 8, 10, 2, 12].

- More dynamic designs of business and engineering processes towards a holistic optimization that eliminate possible bottlenecks [7, 8, 10, 2, 12].

- Older employees must adapt to newer process dynamics (smart assistance systems). This challenge is bidirectional. On the one hand, older employees must adapt to a new production technology (intuitive tools), and on the other hand this smart system needs to adjust to every worker's speed in order to constitute a real collaborative and safe partnership [7, 8, 11, 13, 4].

- Proactive maintenance of machines in order for it to fall under periods of already scheduled process stops. For that, intelligent diagnostic protocols must be embedded in all devices. Final goal is to eliminate unplanned stops, thus increasing productivity [8, 11, 3].

- Secure communication networks and high availability independently to unpredictable threats [8, 11, 14, 15, 9].

- Unification and integration into a common platform of hardware and software from different applications or generations. Main objective for the instauration of a global network of communication is that the entire lifecycle information of all the assets in the plant can be managed homogeneously [16, 7, 8, 14, 15, 9].

3 Main challenges of industry 4.0

As it was said before, the shift in manufacturing processes will be settled in the employment of smart objects interacting with each other and with the user. It has already been stated that the basic pillar in this trend will be defined by the combination of hardware and software into embedded systems denominated cyber physical systems (CPS) that will make use of the internet of things (IoT) and big data to connect every device, that will have its correspondent identifier and basic computer capabilities in order to sense and/or act.

Hence, several tools are needed not only to conform and function in the embedded system, but to develop as well some kind of interface or network in order to extend it, which constitutes the base for the development of this industry 4.0, from the sensors, actuators and control units that gathered information provided by different elements in the process, to a cyber physical system that can manage this information and make decentralized decisions, to the smart machines that will follow the self-deduced actions to take, to the platform that must sustain great flows of information, to even the simulation of processes and design and testing of prototypes based on intelligent and self-aware systems. Indeed, the science of artificial intelligence is settled in the same principles (statistical methods, machine learning, mathematical optimization, neural networks, probability, computational intelligence etc) that the ones that will help constructing the combination of physical and conscience worlds.

However, although the combination of hardware and software into an interconnected system is the base of industry 4.0, there are other variables that are important during this transition and can help to ease the different steps and conditions that a production chain from the fourth industrial revolution must reached in this new paradigm. These include autonomous robots that can work collaboratively with humans in safe conditions, simulation and virtualization tools to help during decision-making stages, or additive manufacturing with utilities not only in the fabrication of prototypes, but also as final product candidate. Next sections describe some of these tools in charge of receiving, gathering, managing, analyzing, interpreting and intercommunicating huge amounts of information proceeding from every part of the manufacturing system; executing decentralized decisions based on the

analysis of information; developing 3D models from objects previously designed with computer programs; and allowing the simulation of the entire supply chain and every process included in it in order to make efficient decisions.

3.1 The internet of things (IoT) and cyber physical systems (CPS)

Citing the Federal Ministry of Education and Research in Germany (BMBF): "Industry is on the threshold of the fourth industrial revolution. Driven by the internet, the real and virtual worlds are growing closer together to form the Internet of Things" [18]. Hence, IoT can be defined as the network system that supports the tools for communication between smart devices and their interconnectivity. In the configuration of IoT, two separate and equally important variables are required: On the one hand, a complete gear of sensors and tags in charge of capturing the information that the different stages and machines in the process generate; and on the other hand, communication software protocols to transfer this information to a central server.

The CPS is then the final responsible for the management and analysis of the information sent by these interconnected systems between its physical assets and computational capabilities; while the advanced connectivity network integrated in the IoT must ensure real time data acquisition from the physical world, as well as posterior information feedback from the cyber space [16]. These capacities allow self-comparison between present and past states, and assist in the decentralized decisions of recommended course of action, making machines self-configure and self-maintainable. The structure of a CPS can be divided in different levels to make machines self-aware and self-adaptive: Smart connection level (gathering of all information); data-to-information conversion level (extract the relevant information); cyber level (includes the virtualization hub to exchange information through other cyber interfaces); cognition level (where optimization decisions take place); and configuration level (for feedback deployment) [1].

Last fundamental concepts for this pillar of industry 4.0 are Big Data and cloud computing, which creates a medium that can handle all the managed information by CPS and IoT. Huge amounts of information is expected to be stored and processed, so later on can be accessible from anywhere at any time, thus, cloud computing constitutes an optimum solution for storage performance, as well as Big Data analysis aids in the management of the information. The capacity to support and control big flows of information is one of the most important applications of industry 4.0, which relies on the maintenance of artificial intelligence networks supported by digital product memories, translated into the collection of all data records for all data stages during the product life cycle, for posterior analysis, that could lead to newer and innovative methodical approaches for planning and development of products [6, 16].

3.2 Autonomous robots

Once the information has been received and analyzed; and decisions have been taken, another logical step consists in the actual execution of those measures. In the context of industry 4.0, self-learning and self-configurative robots are in charge of these actions, in complete collaboration with human workforce. Traditional concepts like proper design, operation performance, energy efficiency or maintenance are still important, with the difference that autonomous robots will have the capacity of managing some of this data to adjust and suggest changes by themselves to improve and predict their functionality and flexibility [3]. The proper combination of sensors, artificial intelligence and even robotic design are fundamental in this field. Technological enhancements have already improved robotics substantially over the past years, making robots suitable for almost every sector [8]. The increasing autonomy of robots will lead, however, to another consequence, which is the need for the establishment of safety protocols for operators working in the same area by, for example, changing to soft structural materials to minimize damage by impact, or the design of annulling mechanisms when a human enters in a sensitive area.

Nevertheless, machines from this new revolution and AI networks must ensure the means to support an infrastructure where robots are intended to work collaboratively with humans, facilitating their work instead of replacing them (adapting precisely to the human work cycle [19, 11]). Machine vision sensors, AI and learning software are the three most important variables that will allow the synergy between independent productive entities and shop-floor operators in a safe environment.

3.3 Additive manufacturing

Additive manufacturing or 3D printing, that is the capability to produce three-dimensional objects directly from virtual models, is another pillar of industry 4.0, with multiple possibilities, especially for designing and testing prototypes, where newer methods of modeling and reference models are continuously appearing without the need of moulds, so one machine can be used for the manufacture of different products, thus reducing production costs [6]. Even though so far, this application has not been broadly applied mainly because of slow production rates, few available materials and high prices, great developments are being made to solve these issues and improving the efficiency of producing individually customized products, allowing for rapid prototyping and highly decentralized production processes. One positive impact of this can be found in; for example, by counting with different printing locations; transportation, storage and other manufacturing costs would decrease, since the product model could just be sent off to the nearest printing site to the

customer [8]. This can be applied the other way around, enabling customers to print their own 3D designs, allowing them to discuss the possibilities of the desired final product with the manufacturer prior its construction [2].

3.4 Augmented reality, simulation and visualization systems

Every decision, whereas it is related with logistics, manufacturing or future changes must be sustained in well-founded arguments. Industry 4.0 technologies will ease the deployment of simulation scenarios where different configurations can be tried and tested before their actual implementation, thus allowing the implantation of more complex systems. Simulation of how changes can affect process behavior are a huge benefit towards the prediction of how these resources or services will impact final value added for end users [17]. Again, artificial intelligence can provide the means for simulation in every stage of the life cycle of a product (from model and design, to functionality prediction). One example of this application may consist in the development of newer methods of modeling and reference models, like integrated computational materials engineering (ICME), where the performance of design materials and dimensions can be tested before construction of the element [6]. However, this pillar of industry 4.0 does not only refer to product properties, being possible as well the implementation of virtualization technology that creates complete digital factories which can simulate the entire production process, in order to optimize layout disposition. This is especially useful for launching new products in already existing plants; by first simulating and verifying virtually the consequent impact in production and human-machine interactions; and only when the final solution is ready, the physical map is done, meaning that all software, parameters, and numerical matrixes are uploaded into the physical machines controlling the production.

4 Inclusion of ICT in embedded systems

The previous section has shown the most important tools for the development of industry 4.0. As it was said before, the combination of hardware and software into smart embedded systems will be greatly resting in AI applications. However, sometimes those smart devices are generally referred to as a whole scale with blurred barriers, where it is difficult to establish when one element ends and the other starts. For example, during the first years of development of this concept, IoT was proposed to refer just to uniquely identifiable interoperable connected objects with radio-frequency identification (RFID) technology. Later on, however, as the connectivity of these networks were getting bigger and including new technologies and concepts, the term was growing with it to include all these innovations, applied to measure, identify, position, track and monitor objects [10], referring now to IoT more as a dynamic global network where self-conscious objects connect with each other. In this new context where CPS can be considered the proper "brains" inside industry 4.0, one way to establish some frontiers can be to consider IoT as the global framework where identification and sensor technologies become integrated with interpretation technologies like CPS. In other words, CPS forms part of IoT´s new step towards its development [19] [2] [14], with the help of ICT elements to guide the autonomous communication between all of them. Next paragraphs will depict the most relevant parameters in this established network by IoT and CPS that constitute the embedded system; including sensoric equipment, their communication protocols through software architecture, standardization languages for the gathered information, big data management, cloud computing and middleware connectivity, or architecture guidelines for construction of CPS.

4.1 Sensors for IoT

It has been said that the integration of sensors/actuators, radio frequency identification (RFID) tags, and communication technologies served as the foundation of IoT, and consequently industry 4.0. According with that line of thinking; from a conceptual standpoint, sensors settle the principles for which smart objects are able to [17]: Be identifiable, communicate and interact with each other, with users and/or other entities within the network.

In the context of identification, sensing and communication devices, radio frequency identification systems (RFID) are the key components for industry 4.0 [22]. RFID systems are composed of one or more readers and several RFID tags. Tags are characterized by a unique identifier and can be applied to objects or people. The readers are used to trig the tag transmission by generating an appropriate signal, which represents a query for the possible presence of tags in the surrounding area and for the reception of their IDs. From a physical point of view, a RFID tag is a small microchip attached to an antenna (that is used for both receiving the reader signal and transmitting the tag ID) in a package which usually is similar to an adhesive sticker. Dimensions can be very low (0.4x0.4x0.15 mm) [22]. Depending on how the energy is supplied, we have to differ between passive RFID tags (energy for operation is supplied by the RFID interrogation signal itself), active tags (on-board power source feeds the on-board receiver and transmitter, allowing for an increased radio range), and semi-active or semi-passive, where on-board power source is used to feed the microchip, whereas transmission is either active (semi-active) or performed using back-scattering (semi-passive) [17].

4.2 Communication protocols for sensors

Along with RFID technology, other complementary devices for identification are sensor networks or wireless sensor networks (WSN) [16]. Sensor networks consist of a certain number (which can be very high) of sensing nodes communicating in a wireless multi-hop fashion [14, 15].

Usually nodes report the results of their sensing to a small number (in most cases, only one) of special nodes called sinks. Typically, a node, which is the WSN core hardware, contains sensor interfaces, processing units, transceiver units and power supply [15]. In fact, they can cooperate with RFID systems to better track the status of things, thus augmenting the awareness of a certain environment and act as bridge between physical and digital world, helping the exchange of information inside the network.

Some of the most common hardware and software available for WSN that serve as communication protocols with unique addressing schemes and standards are IPv6 (to connect unlimited number of devices) [22, 10, 15], WiFi and Wimax (to provide high-speed and low cost communication) or Zigbee and bluetooth (for local communication) [10]; or others like WLAN, M2M or RFID [14].

4.3 Standardization languages

Information can be provided from different sources in the network, whether it is measured by sensors, controllers or manufacturing systems such as ERP. One of the first steps in the development of the network is to acquire this data in a reliable way and interpret it. However, considering that the different sources may give different types of data, there is a need for a seamless method that can manage the acquisition of information, the connection between various types of networks through various communication technologies, and the transformation into a final uniform type of data to be sent to the central server. For this matters, specific protocols such as MTConnect, OPC or ROS are effectively useful [1] [3]. In the same way, machines and robots in industry 4.0 must ensure as well certain level of standardization and trustworthy in the management of information, so it can be integrated with other branches within the same industry, and with other industries and industry types; in order to extract conclusions from hypothesis of how is the best way to support different types of plant with information technology [6]. In conclusion, standardization methods must be sought and implemented to support heterogeneity at architectural and protocol levels [17], where artificial intelligence is necessary not only for the interpretation of different languages, but for the transmission of the main conclusions to user, in an accessible way.
This idea is linked with innovative concepts for manufacturing execution systems (MES) or enterprise resource planning systems (ERP) [6], that can complement the shift towards new business models, where products become modular and configurable in order to be able to adapt to specific requirements [12]. Nevertheless, this task would include a very complicated heterogeneous network, and that is why there is still lack of a widely accepted common platform that can embrace the large heterogeneity of communication technologies. In addition, the large traffic of data at the same time would also cause delay and communication issues.

4.4 Big Data, cloud computing

The great flow of information in these embedded systems needs for technologies that ease the automation, and Big Data can be a solution for that, by enhancing as well important variables like mobility, flexibility and energetic efficiency, providing a temporally and spatially independent access to them [12] [17]. In conclusion, the application of data mining serves to sustain the analysis, modeling, simulation, fusion and computation, and scientific prognosis for decision making [10].

Cloud computing is, on the other hand, basically a large-scale, low cost flexible processing unit, based on IP connection for calculation and storage. The need for cloud computing is founded in the fact that some relation must be established between identification devices towards a storage for huge amounts of information [16] and the need for a centralized infrastructure to support this storage and posterior analysis [15]. Some of the most relevant characteristics are the possibilities for ubiquitous network access; rapid elasticity by increasing or decreasing capacity at will (pay per use concept); and independent location of the resources pooling [10].

5 Conclusions

Industrial environments are currently setting the foundations for a new shift in the production and manufacturing processes, drawing away from static production chains, to a more flexible, individualized and efficient idea of production. The base of this revolution is settled in the combination of hardware and software components, towards a more intelligent, self-conscious, self-configurative and self-optimize structure that can foresee problems and launch preventive actions to minimize stopping times during production; and to understand the whole lifecycle of the production process in order to be ready to respond to new and continuously changing environments. Four main challenges rise in this revolution:

1. Energy and resource efficiency
2. Reduce time-to-market
3. Increase flexibility towards an almost individualized mass production
4. Gradually upgrade of existing infrastructure

There are different variables that configure the embedded systems that influence in this new, flexible, standardized, decentralized, heterogeneous, innovative form of production, where processes are more clients focused and more resource efficient. Those main tools are cyber physical systems, the internet of things, big data and cloud computing (with the entire required infrastructure to support them); and other

complementary elements like autonomous robots, simulation and virtualization models, and additive manufacturing. In this embedded systems, information gathered by different sensors and devices is interconnected and communicate with each other to, based on the compilation, process, and analysis of this great amount of data by smart objects, take decentralized decisions towards optimization of the production. Although this idea as a whole is still in its first steps of development, some possible applications are already being identified and tested in domains such as transportation, food supply chain, logistics, environmental monitoring, prototypes domain or industrial activities.

This situation gives form to a broth of several ideas and technologies where artificial intelligence may have a perfect niche for its thrive and implementation in the industrial environment, since its applications can give answers to different questions and possibilities within each one of the main pillars in which industry 4.0 will be structured.

6 References

[1] Lee, J. and Bagheri, B. and Kao, H., "A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems," Manufacturing Letters, vol. 3, pp. 18-23, 2015.

[2] FME, "Smart industry. Dutch industry fit for the future," 2014.

[3] Lee, J. and Kao, H. and Yang, S., "Service innovation and smart analytics for Industry 4.0 and big data environment," Product Services Systems And Value Creation: Proceedings Of The 6th Cirp Conference On Industrial Product-Service Systems, vol. 16, pp. 3-8, 2014.

[4] Kagermann, H. and Wahlster, W. and Helbig, J., "Recommendations for implementing the strategic initiative INDUSTRIE 4.0," 2013.

[5] Lee, J. and Lapira, E. and Bagheri, B. and Kao, H., "Recent advances and trends in predictive manufacturing systems in big data environment," Manufacturing Letters, vol. 1, no. 1, pp. 38-41, 2013.

[6] Lasi, H. and Kemper, H.G. and Fettke, P. and Feld, T. and Hoffmann, M., "Industry 4.0," BUSINESS \& INFORMATION SYSTEMS ENGINEERING, vol. 6, pp. 239-242, 2014.

[7] Baumgarten, R. and Bick, W. and Bleicher, F. and Butcher, C. and Fricke, K. and Geisel, J. and Hofmann, J. and Leppert, W. and Neuwirth, A. and Schönefeld, L. and Spaeth, D. and Spannblöchl, K. and Stading, C. and Stahr, M., "Industry 4.0. The fourth industrial revolution has begun: On the way to the factory of the future," Hoerbiger motion, 2014.

[8] Bechtold, J. and Lauenstein, C. and Kern, A. and Bernhofer, L., "Industry 4.0 - The Capgemini consulting view. Sharpening the picture beyond the hype," 2014.

[9] Hadim, S. and Mohamed, "Middleware: Middleware challenges and approaches for wireless sensor networks," IEEE distributed systems online, no. 3, p. 1, 2006.

[10] Bi, Z. and Da Xu, L. and Wang, C., "Internet of Things for enterprise systems of modern manufacturing," Industrial Informatics, IEEE Transactions on, vol. 10, no. 2, pp. 1537-1546, 2014.

[11] Bloem, J. and van Doorn, M. and Duivestein, S. and Excoffier, D. and Maas, R. and van Ommeren, E., "The fourth industrial revolution. Things to tighten the link between IT and OT," Sogeti. VINT (vision, inspiration, navigation, trends), 2014.

[12] Jazdi, N., "Cyber Physical Systems in the Context of Industry 4.0," 2014 IEEE INTERNATIONAL CONFERENCE ON AUTOMATION, QUALITY AND TESTING, ROBOTICS, pp. --, 2014.

[13] Dombrowski, U. and Wagner, T., "Mental strain as field of action in the 4th industrial revolution," Variety Management In Manufacturing: Proceedings Of The 47th Cirp Conference On Manufacturing Systems, vol. 17, pp. 100-105, 2014.

[14] Da Xu, L. and He, W. and Li, S., "Internet of things in industries: A survey," Industrial Informatics, IEEE Transactions on, vol. 10, no. 4, pp. 2233-2243, 2014.

[15] Gubbi, J. and Buyya, R. and Marusic, S. and Palaniswami, M., "Internet of Things (IoT): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, pp. 1645-1660, 2013.

[16] Abu-Elkheir, M. and Hayajneh, M. and Ali, N., "Data management for the internet of things: Design primitives and solution," Sensors, vol. 13, no. 11, pp. 15582-15612, 2013.

[17] Monostori, L., "Cyber-physical production systems: Roots, expectations and R\&D challenges," Variety Management In Manufacturing: Proceedings Of The 47th Cirp Conference On Manufacturing Systems, vol. 17, pp. 9-13, 2014.

[18] MacDougall, W., "Industrie 4.0. Smart manufacturing for the future," 2014.

[19] Campbell, I. and Bourell, D. and Gibson, I., "Additive manufacturing: rapid prototyping comes of age," Rapid Prototyping Journal, vol. 18, no. 4, pp. 255-258, 2012.

Use of Soft-Computing techniques to estimate the performance of the IBEX 35 Values

Arturo Peralta[1], Ricardo J. Rejas[2], Francisco P. Romero[1], José A. Olivas[1], Jesús Serrano-Guerrero[1]

[1]*Dept. of Information Technologies and Systems, University of Castilla La Mancha, Ciudad Real, Spain.*

[2]*Faculty of Law and Economy,Universidad Camilo José Cela, Spain*

Abstract – *The prediction of the stock-market performance is one of the main challenges of investors in variable assets. For this reason, there are currently a number of methods used to help with this. However, usually unforeseen events make any estimation based only on previous performance invalid. In this study a method is proposed based on the use of Soft-Computing techniques to aid in predicting the performance of a Value, taking into account not just previous historical performance, but also the opinion of the mass of possible investors or sector analysist experts.*

Keywords: Stock prediction, pattern recognition, clustering.

1.Introduction

As it is well known, in the financial market arena, the objective of every investor is to predict the stock-market performance. However, as the activity of the stock-market is influenced by a number of factors, predicting its exact performance means getting to know the individual psychology of every investor, and their investing style.

For this reason, there are a number of models which are designed to detect trends. However, the majority of them are based on studies which are purely technical or basic, leaving out aspects such as mass psychology, expert opinions or the possibility of unforeseen events which could cause instability. Resolving these weaknesses is the objective of this study. To do this, past performance patterns are considered, along with a combination of economic indicators and external variables which could represent the social feeling, which could provide a complete frame of reference and help in the making of decisions. With this objective, in this study it is proposed to use of KDD (Knowledge Discovery in Databases)[22] and Fuzzy Deformable Prototypes [20] [30], in order to extract dark knowledge from data and stock-market commentary by possible investors or sector experts, which should facilitate the decision making process.

2. Related Work

The prediction of stock-market performance is a key aspect for any investor. This is why there are a number of studies [9] [17] [3] [4] that try, through the use of techniques based on technical or basic analysis, to provide a solution to this problem. However, the uncertainty of the 'human factor' makes it worth looking into solutions that combine traditional techniques with others based on Soft-Computing, such as the fuzzy logic or clustering algorithms. Nevertheless, there are few studies which apply these techniques to make stock-market performance predictions, and they can be put into three big groups. Specifically, those based on the exclusive use of neuronal networks, those that use hybrid models which combine genetic algorithms with neuronal networks, and finally those that are based on the use of mathematical techniques, although improved by using a Soft-Computing tool to allow the representation of knowledge in a fuzzy manner.

Within the first of the groups, the most prominent studies based exclusively on neuronal networks are the following. In [5] a study is carried out about the application of neuronal networks and the calculation of some representative indicators, such as VAMA (Volume Adjusted Moving Average) and EMV (Ease of Movement) to make stock-market predictions of a group of selected companies belonging to the S&P index. In [21] [12] two studies are presented based on the use of neuronal networks and the analysis of a temporary series, to make a prediction in the very short term. Finally, [29] stands out, providing a fuzzy model based on neuronal networks to classify performance with the aim of predicting trends in the short term.

In continuation with this classification, the second of the groups are those which present hybrid models, combining neuronal networks and genetic algorithms. In [1] a model is proposed which, by using relative data from technical analysis, and expert knowledge, making use of neuronal networks and genetic algorithms, it tries to predict the closing price of a determinate value or a group of them.

With this same objective, [18] is a study of the combined use of both techniques, although based on the extraction of rules which could suggest the moment to buy or sell. Something similar is described in [8], although in this case it is more focussed on the daily ups and downs from an objective speculator.

Within the third identified group are those based on the mathematical or statistical methods along with some Soft-Computing techniques, one relevant study is [25]. In this study, a model is presented based on neuronal networks and complex statistical functions which are capable of calculating the RIS (Relative Strength Index) indicator, showing its application in relation to the IBEX 35 general index, and to a concrete value. In a similar way, in [15] a system is described which uses a group of financial indicators, making use of neuronal networks and statistical methods, to try to calculate a series of ratios to predict the stock-market performance of a group of companies from Standar and Poor's. [31] also stands out, it shows a model based on geometric series and mathematical functions where the indicators of volatility, risk and price are shown as fuzzy numbers.

In general, one of the main limiting factors of the above studies is that they usually use data based on previous performance, or calculable indicators from this data, not taking into account from the beginning that possible eventualities could change the predicted trend.

3. Proposal

The prediction of the stock-market performance, or the general detection of trends, is difficult to do. The combination of external factors (knowledge of information that could have a negative impact, percentage of unemployment, macroeconomic data, consumerism, IPC, PIB, etc.) and the inevitable 'human factor', means that in the majority of cases, initial predictions based on data and information of previous situations, is very different from reality. This is the context for the research into the use of Soft-Computing techniques, which uses the data provided by free databases with information on previous performance and stock-market commentary as a possible solution to improve these predictions, basing itself on the study and analysis of non-ideal situations which have been affected by external factors that could distort it.
To achieve this, a system capable of analysing historical data for pattern recognition is proposed. Figure 1 shows a diagram of the different phases that make up the proposed process.

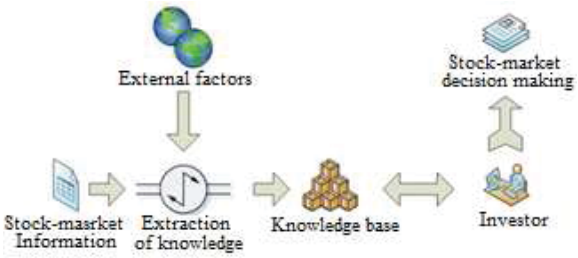


Figure 1. General Frame

The phases carried out are the following:

1. Stock-market information: Information Extraction from the free access databases, with the performance of each one of the IBEX 35 values from previous years. These registers contain data such as: Maximum, Minimum, % Variation, Volume of negotiation, BPA, PER, EV/Ebitda, date, Maximum and Minimum from previous week, etc. Furthermore, stock-market commentary from expert analysts and commentary which has been published in texts by general readers and obtained through the development of an IT procedure that allows the extraction of commentary that can be found on used web articles is also considered. To obtain microtexts from Twitter an interface has been developed which uses the Search API provided by Twitter. Obtaining microtexts from Facebook is carried out using Facebook's API3.

2. External factors: This is related to the elements that could affect a prediction obtained through the ideal ways, such as: knowledge of new macroeconomic data, percentage of unemployment, consumer index, IPC, etc. In addition, the occurrence of large unforeseen events causing instability in the markets, like the impact of wars or terrorist attacks, the publication of new governmental economic measures like the privatisation of public businesses, the granting of fiscal advantages to stock-market investors or a to certain business sectors, etc. These create a source of unforeseen factors, which make it difficult to predict trends and make decisions. So in this study, the opinion of sector experts and possible investors are taken into account, with the objective of learning and valuing the impression from society beyond what is extractable from economic data.

3. Extraction of knowledge: This is made up of the following phases:

3.1 – Data Cleaning, elimination of data that is not relevant, correction of typos and detection of possible data errors throughout. The different processes that should be carried out are listed below.

• Detection of sessions of erroneous data. Incongruent data is detected: impossible dates, illogical data, missing values, etc. Techniques are used to detect atypical values, like Grubbs Test [10], the Dixon Test as described by Li and Edwards [16], the Mahalanobis Outlier Analysis as proposed by Matsumoto [19] or the Robust Regression and Outlier Detection proposed by Rousseeuw and Leroy [28].
• StopWords Removal. The objective of this phase is to delete all the unnecessary information contained in the textual commentary associated with every Value. [14].
• Spellcheck. Standard automated dictionaries are used for correction. These dictionaries can be enriched through using additional and specific dictionaries for the context.
• Grouping together of stock-market commentary according to its level of positivity. In this phase a process is carried out to extract the feeling about textual comments, with the objective of grouping them according to their level of positivity about the evolution of a value. A clustering algorithm is used like that shown in [27] [26]. This is based on the use of functions of similarity that analyse the distance in terms of words and the total of the text, grouping them into groups of low distance and high similarity. Furthermore, *stemming* techniques are applied, like that of Korenius [13] to extract the lexical roots in order to group by concept.

3.2 -Transformation of data and calculation of indicators, to characterise the performance of days, weeks and of the typical values.

• Characterisation of days: For the characterisation of days, it is necessary to do an analysis of the data supplied by the historical databases on stock exchange prices for each value, like closing price, maximum, minimum, volume, etc. From this information, a combination of daily global indicators is established.

Indicator	Description
VA_P: Price Variation	Percentage of variation in the closing price obtained in previous session, that is, $VA\_P=((PC_i - PC_{i-1})/ PC_i * 100$
VA_V, VA_C: Volume Variation and Capitalisation	Percentage of variation in respect to that obtained in previous session, that is, VA_V y $VA\_C=(VOL_i - VOL_{i-1})/VOL_i * 100$
PER: Relation between Price and Profits.	The existing relationship between the price and the profit of the action, that is , *PER=(PC/Beneficios)*
BPA: Profit for the action	Profit obtained for each action, that is, *BPA=Beneficio/Nº Acciones*
RDIV: Profitability for Dividend	Relationship between the dividends and the price of the action, that is, *RDIV=Dividendo por acción/PC*

Table 1. Cleansing of stock-market commentary

• Characterisation of the weeks: One week is made up of days, so one week is characterised by the same indicators laid out above for days, taking into account the nature of a day within a week (working day, public holiday etc.). The values added by these indicators can also be taken into account.
• Characterisation of the Values: Making use of the previously specified indicators, it is possible to calculate the global indicators for the Values, which can represent and characterise its performance.

Indicator	Description
T_C: Short term trend	Prediction of trend, taking into account price data from the previous week. *T_C=(Movile_Average(last_week)-Last_Price(last_week)/ Last_Price(last_week))*100*
T_M: medium term trend	Prediction of trend taking into account price data from previous month
T_L: Long term trend	Prediction of trend according to price data from previous six months.
E: Prediction by experts	Prediction obtained from textual opinions by established experts
S: Social factor	Impression of the mass of investors, calculable by processing their commentary.

Table 2. Cleansing of stock-market commentary

3.3 -Selection of the most critical indicators. In the previous sections, indicators able to represent days and weeks of prices are described, just as their own values are. However, in the case of analysing indicators corresponding to a possible client base with a high number of Values, it can be complicated to choose which may be truly advantageous. For this reason, the use of a CART decision tree, such as that proposed by Breiman[2], or ID3 as described by Quinlan[23] is proposed. This way, using the data corresponding to the indicators calculated for each Value of each day and week and the evolution of the value (percentage of price variation) in this same unit of time, it is possible to obtain a graphic representation of which indicators are most critical.

4. Knowledge base: From the knowledge extracted and the use of fuzzy logic techniques, the calculated indicators and the obtained prototypes are stored as a support base for the stock-market estimation and prediction process.
5. Investor: Responsible for carrying out the corresponding stock-market actions, taking into account the knowledge base extracted in order to make decisions.
6. Stock-market decision making: To help make stock-market decisions, the investor is recommended by the extracted knowledge base on which of the value or values will be most beneficial.

Through the extracted knowledge, it is expected to gain valuable information, capable of helping a portfolio of clients.

4. Case Study

In the present study, there will be a description of the experience gained by applying the proposal to all the IBEX 35 values in order to extract the necessary knowledge to select the values which are estimated to be the most profitable for short term investment, using the prediction of the calculated performance for the following days as a basis. The proposal is specifically applied to three different scenarios: during a week of stable prices, during an unstable week due to external factors which could cause instability (like the recent terrorist attack on the 13th November in Paris), and during a week with negative stock-market performance.

4.1. Source of data

For the implementation of this proposal to the present case study, the stock-market data related to stock-exchange prices of the total of the 35 values belonging to IBEX 35 during the 2 months previous to the week which is to have its trend estimated.

The following table shows the division of the combination of data used for each one of the three established periods and the week which is to have its stock-market trend estimated.

	Stock-market information used	Trend to estimate
Scenario_1: stable week	16/08/2015-16/10/2015	19/10/2015-23/10/2015
Scenario_2: unstable week	13/09/2015-13/11/2015	16/11/2015-20/11/2015
Scenario_3: negative week	27/09/2015-27/11/2015	30/11/2015-04/12/2015

Table 3. Stock-market information used

The stock-market data is relative to the closing price of each value, the daily maximum and minimum, the volume of negotiation, the capitalisation and the value of the dividends.

In addition, a combination of stock-market opinions are relied upon which are offered by experts and the general public, that once grouped together are put into an Excel spreadsheet with 35 pages (one per each value), in which there is a chronological list of an average of 200 pieces of stock-market commentary per value. Specific software is developed to get these pieces of stock-market commentary, APIs in the public domain are used, for example the Search API supported by Twitter, or the API3 Graph offered by Facebook. Online editions of specialised press for the sector are used to collect the expert commentary.

4.2. Data Cleansing

The process of cleansing began with the data revision and commentary of the 75 days of stock-exchange used. Furthermore, a process of text cleansing was carried out for the stock-market commentary and subsequently they were shown as vectors in order to classify them into 4 groups according to the level of positivity, using a clustering algorithm as described in section 3.

4.3. Transformation of data and calculation of indicators

Through the cleansing of the data and the commentaries, the calculation of the indicators most able to represent a week of stock-exchange is made, for this the values of the first indicators were normalised and subsequently transformed into a scale of values between 1 and 5. To divide a series of numbers into 5 groups a divisive clustering algorithm was used, like that proposed in [6] for the combination of available data and the criteria of the expert to maintain a logical.

Value	Com	Max	Min	Inf	Ant	TC	E	S
Abertis	5	4	4	1	3	4	4	5
Acciona	5	5	4	1	4	4	4	4
Acerinox	5	4	4	3	2	3	3	3
Acs	5	5	3	1	3	4	3	4
Aena	5	4	2	1	3	4	3	4
Amadeus	5	4	4	1	5	5	4	4
Arcelormital	5	2	2	3	2	2	2	3
B. Popular	5	4	3	1	4	4	3	4
B. Sabadell	5	2	2	3	2	2	2	2
Bankia	5	5	4	1	3	4	4	5
Bankinter	5	4	4	1	5	5	4	4
Bbva	5	2	2	3	1	3	2	2
Caixabank	5	3	3	1	3	3	4	3
Dia	5	3	3	1	4	3	3	3
Enagas	5	2	2	3	1	2	1	2
Endesa	5	3	2	1	3	3	3	3
Fcc	5	2	2	3	2	2	1	2
Ferrovial	5	3	3	2	3	3	3	3
Gamesa	5	2	2	3	1	2	1	1
G. Natural	5	3	3	1	3	4	3	3
Grifols	5	4	4	1	3	4	3	4
Iag	5	5	4	1	5	4	4	5
Iberdrola	5	3	1	1	2	3	2	2
Inditex	5	5	4	1	3	5	4	5
Indra	5	4	4	1	3	5	3	4
Mapfre	5	4	3	1	4	5	4	4
Mediaset	5	3	3	2	3	4	3	3
Merlin prop.	5	3	2	2	3	4	3	3
Ohl	5	3	2	2	2	4	3	3
R. Eléctrica	5	2	2	3	2	3	2	2
Repsol	5	4	4	1	3	4	3	4
Sacyr	5	4	4	1	4	4	4	4
Santander	5	5	4	1	4	4	4	5
T. Reunidas	5	3	3	2	3	3	3	3
Telefónica	5	5	4	1	3	5	4	5

Table 4. Indicators for IBEX 35 values

4.4. Selection of the values with the best tendencies

The analysis of the above table by means of clusters consists of finding homogeneous groups of components. Using a function of similarity between vectors (in this case the Manhattan distance), it is possible to construct a matrix of squared proximity, where each component measures the similarity between the row and column component, and in that which the dark tones show great similarity between components and the light tones show a great difference. The similarity matrix is the basis of the cluster analysis.

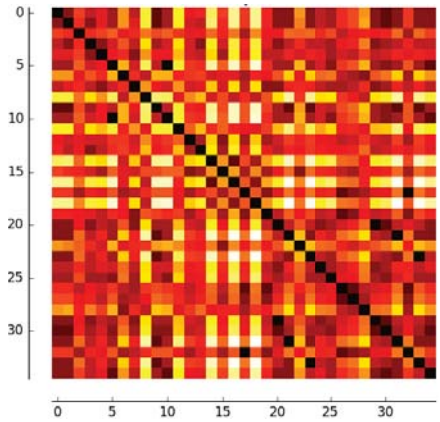


Figure 2. Similarity Matrix

By applying a hierarchical clustering algorithm, a dendrogram can be obtained, using the policy of re-calculus of distance between clusters "the most far away neighbour" to get homogeneous groups of values.

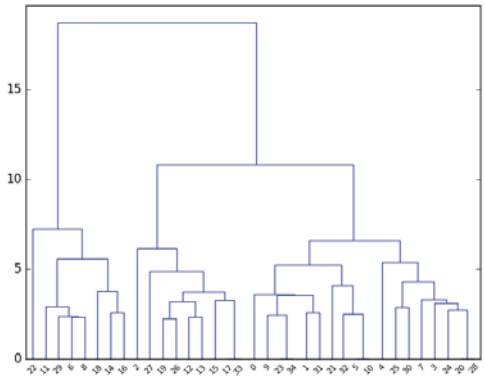


Figure 3. Dendrogram

According to the dendrogram, three quite homogeneous groups exist. Once the different groups have been obtained, the objective is to try to determine which of the indicators are most influential. With this intention, an analysis is carried out, based on decision trees to determine the most decisive variables to select the values which are most profitable in the short term. The following figure shows a decision tree taken from the classification obtained by the dendrogram. To calculate this, an extension of the ID3 [24] decision tree algorithm has been applied, establishing a maximum level of entropy of 0.1 for each leaf.

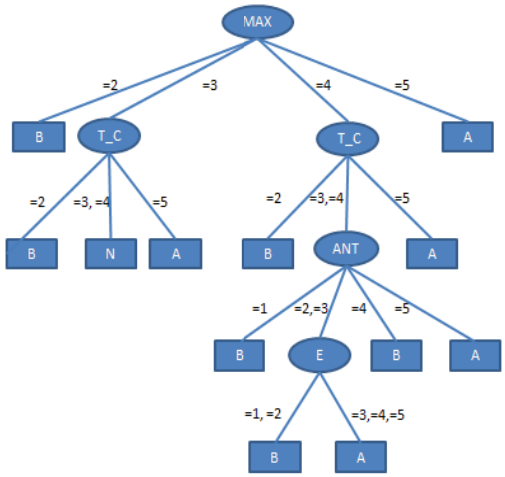


Figure 4. Decision tree with 0.1 entropy

With this representation, it can be clearly observed that the MAX and T_C indicators are the most influential. In a similar way, at the next level of influence, the ANT and E indicators are also relevant. According to this, the values that have a better score in these indicators should be those with the most favourable stock-market performance.

The following table (Table 5) shows the results in dark grey are those with the best scoring values for the indicators T_C and MAX. These values are those that should offer the best performance in the short term, meaning that they would be more beneficial in terms of investment. The results in light grey would be those values that would come in second place in terms of interest, having achieved slightly worse results (although high) in T_C, MAX, ANT and E.

Value	Com	Max	Min	Inf	Ant	TC	E	S
Acciona	5	5	4	1	4	4	4	4
Acs	5	5	3	1	3	4	3	4
Inditex	**5**	**5**	**4**	**1**	**3**	**5**	**4**	**5**
Santander	5	5	4	1	4	4	4	5
Telefónica	**5**	**5**	**4**	**1**	**3**	**5**	**4**	**5**

Table 5. Values selected as most profitable

This process to select the values that should be most profitable was carried out in the same way for the other two scenarios, this was in order to find the values of most interest during the period 16/11/2015-20/11/2015 and 30/11/2015-04/12/2015. For this, in the same way the data and commentaries from the previous 2 months were used.

5. Results and discussion

Once the values that should offer the best performance during the week 19/10/2015-23/10/2015 are selected, the real evolution of the said combination of selected values is shown.
This is for a stable week in the stock-market, without socio-economic occurrences of consequence that could interfere too much with the predicted performances.

The following table (Table 6) shows the closing prices of the three most interesting selected values in the short term (Inditex and Telefonica), in comparison to the typical IBEX 35 index during the analysed week.

	19/10	20/10	21/10	22/10	23/10	**Weekly Variation (%)**
Inditex	31,14	31,2	31,51	32,5	33,5	**8,84**
Telef.	11,01	10,79	10,82	11,38	11,68	**5,51**
IBEX35	10207,3	10100,6	10157,6	10365,4	10476,3	**2,39**

Table 6. Stock-exchange prices of selected values in the first assumption: stable week

The result of both values is higher than the IBEX 35 average, thus it can be concluded that the choice of the two values to make an investment in the short term would be very profitable.

In addition, with the objective of valuing the proposed model, the experiment is carried out very thoroughly, following the same steps indicated, with the intention of predicting the stock-market performance for two other selected weeks. The first of those spans over the days 16/11/2015-20/11/2015; one unstable week for the IBEX 35 due to external influencing factors. The second is over the days 30/11/2015-04/12/2015, which was a complicated week with steep stock-market falls.

Through applying the proposal over the week 16/11/2015-20/11/2015, the most recommended values were: Sacyr and Repsol. The performance of this selection of values in comparison with the IBEX 35 average was the following (Table 7):

	16/11	17/11	18/11	19/11	20/02	**Weekly Variation (%)**
Sacyr	2,20	2,28	2,35	2,38	2,32	4,98
Repsol	10,85	11,39	11,59	11,72	11,45	7,41
IBEX 35	10124,5	10363,8	10261,1	10354,7	10290,3	1,77

Table 7. Exchange prices of the selected values in the second scenario: unstable week

In a similar way, through the application of the proposal in the week 30/11/2015-04/12/2015 the selected values were: Bankinter and Gamesa. The results are shown in Table8.

	30/11	01/12	02/12	03/12	04/12	**Weekly Variation (%)**
Bankinter	6,80	6,94	6,87	6,84	6,89	2,84
Gamesa	16,57	16,28	16,36	16,11	16,34	-0,31
IBEX 35	10386,9	10379,2	10342	10092,9	10078,7	-2,25

Table 8. Exchange prices of selected values in the third scenario: low week

As shown in the data, in the three experiments, all the selected values offered a good performance in the short term. Considering that the average of the variations of the selected values is higher than the average of the IBEX 35, it can be concluded that the implementation of the proposed model has worked correctly, allowing the selection of the most profitable values.

6. Conclusions and future work

In this study there has been a presentation of a series of techniques defined by the discovery of knowledge that allow the modelling of the stock-market performance of a combination of values in a temporary distribution.

This has enabled the differences in exchange prices of days or weeks to be shown, or typical characteristics of the analysed values, through the use of techniques like clustering algorithms and Fuzzy Deformable Prototypes. Specifically, characteristics of the stock-market performance of determined values have been obtained, represented through linguistic variables and fuzzy numbers, with which and by following the model of prediction of performance, it is possible to make estimates and stock-market trends. Furthermore, the proposal has been applied, verifying its working in 3 different situations.

However, one of the possible areas to look at in the future is the reduction in the margin of error in making the estimate. For this, it is possible to explore the incorporation of other variables that could affect the stock-market trend on a global or local level, or the modification of the methods for the calculation of the indicators used.

Furthermore, with the objective of improving the adaption of the model to the needs of the investor, the possibility of incorporating variables which are specific to the investor's profile to the system could be looked at. This way, specific recommendations in the buying and selling of values could be given according to the investor's characteristics and capability to accept risk.

Finally, the other area which could be looked at in the future is the implementation of a fuzzy system capable of generating linguistic summaries from the analysis of the data from registers and stock-market commentary. In this way, there are currently diverse fuzzy estimations to generate linguistic summaries, which are usually made up of quantified propositions for different applications, like those proposed by Diaz-Hermida and Bugarin [7] or Kacprzyk [11] that could be adapted and used in the context of this study. Through these summaries a representation in normal language could be obtained, that puts together the general impressions of sector experts or society in general.

7. Referencias

[1] Armando G. (2005). "A hybrid genetic-neural architecture for stock indexes forecasting". Information Sciences, Vol. 170; 3-33.

[2] Breiman L., Friedman J., Olshen R., Stone C. (1984). "Classification and regression tree". USA. Wadsworth.

[3] Cai Y., Chou R., Li D. (2009). "Explaining international stock correlations with CPI fluctuations and market volatility". Journal of Banking & Finance, Vol. 33; 2026-35.

[4] Caldas G., Pires B. (2012). "Macroeconomic Environment, country risk and stock market performance: Evidence for Brazil". Economic Modelling, Vol. 29; 1666–1678.

[5] Chavarnakul T., Enke D. (2008). "Intelligent technical analysis based equivolume charting for stock trading using neural networks". Expert Systems with Applications, Vol. 34; 1004-1017.

[6] Chavent M., Lechevallier Y., Olivier B. (2007). "DIVCLUS-T: A monothetic divisive hierarchical clustering method". Comput. Stat. Data Anal, pp 687-701. DOI=10.1016/j.csda.2007.03.013.

[7] Díaz-Hermida F., Bugarín A. (2011). "Semi-fuzzy Quantifiers as a Tool for Building Linguistic Summaries of Data Patterns". IEEE Symposium on Foundations of Computational Intelligence; 45-52.

[8] Evansa C., Pappasa K., Xhafab F. (2013). "Utilizing artificial neural networks and genetic algorithms to build an algo-trading model for intra-day foreign exchange speculation". Mathematical and Computer Modelling, nº58, pp. 1249–1266.

[9] García M. C., Jalal A. M., López J. M. (2013). "Métodos para predecir índices bursátiles". Ecos de Economía, Vol. 37; 51-82.

[10] Grubbs, F. (1969). "Procedures for Detecting Outlying Observations in Samples". Technometrics, Vol. 11; 1-21.

[11] Kacprzyk J. (2010). "Computing With Words Is An Implementable Paradigm: Fuzzy Queries, Linguistic Data Summaries, and Natural-Language Generation". IEEE Trans. Fuzzy Systems, nº 18, pp. 451-472

[12] Karen E. (2007). "Automatic extraction and identification of chart patterns towards financial forecast". Applied Soft Computing, Vol. 7; 1197–1208.

[13] Korenius T, Laurikkala J., Järvelin K., and Juhola M. (2004). "Stemming and lemmatization in the clustering of finish text documents". Proceedings of the thirteenth ACM international conference on Information and knowledge management. 625–633.

[14] Korfhage R. R. (1997). "Information storage and retrieval". USA. John Wiley & Sons.

[15] Lam M. (2004). "Neural network techniques for financial performance prediction: integrating fundamental and technical analysis". Decision Support Systems, nº 37, pp. 567–581.

[16] Li D., Edwards E. (2001). "Automatic Estimation of Dixon's Test for Extreme Values Using a SAS Macro Driven Program". PharmaSug. 87-91.

[17] Liua H. C., Hungb, J. C. (2010). "Forecasting S&P-100 stock index volatility: The role of volatility asymmetry and distributional assumption in Garch models". Expert Systems with Applications. Vol. 37; 4928–4934.

[18] Mabua S., Hirasawab K., Obayashia M., Kuremotoa T. (2013). "Enhanced decision making mechanism of rule-based genetic network programming for creating stock trading signals". Expert Systems with Applications, Vol. 40; 6311–6320.

[19] Matsumoto, S., Kamei, Y., Monden, A., Matsumoto, K. (2007). "Comparison of Outlier Detection Methods in Faultproneness Models". Proceedings of the First international Symposium on Empirical Software Engineering and Measurement ESEM. IEEE Computer Society; 461-463.

[20] Olivas J.A. (2000). "Contribución al Estudio Experimental de la Predicción basada en Categorías Deformables Borrosas", Universidad de Castilla La Mancha, España. Tesis Doctoral.

[21] Oliveira F. A. (2014). "Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index – Case study of PETR4". Expert Systems with Applications, Vol. 40; 7596–7606.

[22] Peralta A., Romero F. P., Polo M., Olivas J. A. (2010). "Knowledge extraction of the behaviour of software developers by the analysis of time recording logs". FUZZIEEE. pp. 1-8.

[23] Quinlan, J.R. (1986). "Induction of decision trees". Machine Learning, Vol. 1; 81-106.

[24] Quinlan, J.R. (1987). "Simplifying Decision Trees". International Journal of Man-Machine Studies 27(3): 221-234

[25] Rodríguez-González A., García-Crespo A, et al. (2011). "CAST: Using neural networks to improve

trading systems based on technical analysis by means of the RSI financial indicator". Expert Systems with Applications. Elsevier.
[26] Romero F. P., Peralta A., Soto A., Olivas J. A., Serrano-Guerrero J. (2009). "Fuzzy optimized self-organizing maps and their application to document clustering". Soft- Computing. Vol. 14; 857-867
[27] Romero F.P., Peralta A., Olivas J.A., Serrano-Guerrero J. (2007). "Clustering documental basado en Mapas de Kohonen optimizados mediante Técnicas de Lógica Borrosa". CEDI. 97-105.
[28] Rousseeuw P., Leroy A. (1996). "Robust Regression and Outlier Detection". USA. John Wiley & Sons.
[29] Svalina I., Galzina V., et al. (2013). "An adaptive network-based fuzzy inference system (ANFIS) for the forecasting: The case of close price indices". Expert Systems with Applications, nº 40, pp. 6055–6063.
[30] Zadeh, L. A. (1982). "A note on prototype set theory and fuzzy sets". Cognition, Vol. 12; 291- 297.
[31] Zhanga W-G., Xiaob W-L., Konga W-T, Zhang Y. (2015). "Fuzzy pricing of geometric Asian options and its algorithm". Applied Soft Computing, Vol. 28; 360–367.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

KNOWLEDGE DISCOVERY AND MACHINE LEARNING

Chair(s)

Dr. Raymond A. Liuzzi
Dr. Roger Dziegiel
Dr. Todd Waskiewicz
Dr. Peter M. LaMonica

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Artificial Intelligence (AI), Big Data, and Healthcare

Gerard T. Capraro
Capraro Technologies, Inc., 401 Herkimer Road, Utica, NY 13502 USA
gcapraro@caprarotechnologies.com

Abstract

This paper provides a state of the art review of sensors in healthcare and a projection as to how Artificial Intelligence (AI) along with Big Data can improve and reduce the cost of healthcare. Reducing the cost of healthcare is dependent on leveraging 5 technologies, i.e. the Internet of Things (IoT), sensors, IPv6, AI, and Big Data. A IoT architecture is provided using the Cloud Standards Customer Council (CSCC) Cloud Components [15]. The described architecture will provide better and cheaper healthcare and allow researchers access to data that are not available today to further their understanding of the human body and its abnormalities.

Keywords: Artificial Intelligence, Internet of Tings, Big Data, Healthcare, Sensors

Section 1 Introduction

The US Administration of the Aging states that in 2013 the number of persons older than 65 was about 44.7 million or 14% of the US population and that in the year 2060 this number should approximately double [1]. Similar changes are occurring worldwide according to [2] "…by 2050 approximately 20% of the world population will be at least 60 years old". This increase in longevity can be attributed to our healthcare industry in general and the results garnered from the medical research community e.g. heart, cancer, and diabetes research. The numbers are staggering when considering dementia. According to [3] Alzheimer's patients will reach 16 million by 2050. Currently these patients are disabled for 9 to 20 years. However, it is projected that these numbers will increase to 40 to 50 years because of medical advances. The cost of medical treatment will also increase from $172 billion per year in the US to $1.08 trillion by 2050. Today's adults will spend more time caring for their parents than they have for their children.

Part of the increase in cost within the healthcare industry will be due to the raising of minimum wages, some to the increase cost of education, and some due to increases in the cost of medical services i.e. hospitals, nursing homes, prescription drugs and physicians [4]. To combat some of these projected cost increases the US government is changing the way doctors and hospitals bill their services. This new approach is called Capitation [5]. Capitation allows the government, which started with Medicare-Medicade, to pay one amount for each client. Doctors, nurse practitioners, hospitals, insurance companies, and other health providers formed groups called Accountable Care Organizations (ACO). These organizations, both government and private, take on the risk of collectively providing quality preventative healthcare since "…there is greater financial reward in prevention of illness than in treatment of the ill." The ACO approach will require all parties to accept a collaborative approach to "…focus on the whole patient or on populations of patients, encouraging and requiring teamwork among clinicians across specialties, as well as coordination among clinical care units and healthcare organizations of all types across the continuum of care (e.g., physician groups, hospitals, health systems, payers, and vendors)". This approach puts the direct risk in keeping healthcare costs down with the healthcare industry rather than the government.

Efforts are abundant in trying to maintain or reduce the cost of healthcare. The elderly are being cared for in their homes by family, friends, visiting nurses and aids. Independent living centers have been created to help those who can still help themselves and, for those who are in physical and mental need and require 24/7 care, there are nursing homes. For those that have medical emergencies there are health clinics, physician offices, ACOs and hospitals.

A goal of this paper is to provide a state of the art review of the sensors in healthcare and provide a projection as to how Artificial Intelligence (AI) along with Big Data can improve and reduce the cost of healthcare. Section 2 provides a description of some of the current sensors being used, studied and deployed for enhancing the care of patients. Section 3 provides our Internet of Things (IoT) architecture that will

reduce the cost of healthcare, provide state of the art healthcare to millions of people, and provide researchers the data they need to diagnose and prevent abnormalities that is not available today. Section 4 presents our summary and conclusions.

Section 2 Sensors and AI.

In the USA there is a commercial on TV where an elderly woman is shown on the floor and she presses a button on a device attached to a necklace, and she is heard saying "Help I have fallen down and I can't get up". There are many devices on the market that provide help for those individuals who wish to stay in their homes and have family members, visiting nurses, next door neighbors, friends etc. periodically check on their health status. To accelerate and enhance this capability there are numerous wireless body area network (WBAN) sensors that can be used to automatically alert 911, family members, or neighbors.

There are three types of sensors, i.e. ambient, physiological and biokinetic. These sensors can be partitioned into active or passive. An active sensor is sometimes called an actuator where based upon its sensed value it may take some action such as dispensing insulin for a diabetic. Ambient sensors record what is in the environment that the wearer is immersed within e.g. outside/inside temperature, humidity, chemical, biological, radiation and nuclear (CBRN) sensors, a wearer's location using Global Positioning System (GPS), audio, video, images, etc. Biokinetic sensors measure acceleration and angular rate of rotation in which one can determine human movement e.g. whether the wearer is walking or is at rest. Physiological sensors or biological sensors measure human parameters as shown in Table 1 [6]. Some of these sensors can be embedded in a wearer's clothing such as their vests, bra, etc.

Many applications use sensors embedded on smartphones that will record a person's heartbeat, GPS location, number of steps taken, etc. However, the accuracy of these applications have gotten mixed reviews lately as to how they compare to individual sensors. See [7] for the American Heart Association's take on mobile health technologies, mHealth.

There are newer sensors that have AI embedded within them. For example, there is the Wriskwatch [8], with a piezoelectric disk strapped to the wrist. It detects the arterial swelling at the radial pulse and can identify episodes of ventricular fibrillation, or v-fib, the most serious cardiac rhythm disturbance. Wriskwatch uses Bluetooth to send a signal when loss of pulse occurs. The device has AI software that integrates the information regarding pulse rate with a motion sensor to reduce the number of false alarms. A future instantiation will have the device emit an audible signal which will allow the patient to shut off the unit before summoning the paramedics due to a false alarm.

Table 1 Example of WBAN Medical Devices

Application	Data rate	Bandwidth (Hz)	Accuracy (bits)
ECG (12 leads)	288 kbps	100 1000	12
ECG (6 leads)	71 kbps	100 500	12
EMG	320 kbps	0 10,000	16
EEG (12 leads)	43.2 kbps	0 150	12
Blood saturation	16 bps	0 1	8
Glucose monitoring	1600 bps	0 50	16
Temperature	120 bps	0 1	8
Motion sensor	35 kbps	0 500	12
Cochlear implant	100 kbps		
Artificial retina	50 700 kbps		
Audio	1 Mbps		
Voice	50 100 kbps		

There are many types of sensors in the medical field that measure and record oxygen levels, blood pressure, heart monitors, etc. Research is occurring in the micro and nano chemical sensors that will provide continuous monitoring and medical diagnosis in near real time. One example of this type of research is being performed at University of Illinois at Urbana-Champaign [9]. Their research involves deploying a sheath around the heart. They have noted that current methods that sense one portion of the heart and activate defibrillators sometimes measure activity from other portions of the heart and thereby providing unnecessary electric shocks to the heart. With hundreds of electrodes in contact with the total heart via their new membranes they will be able to provide more accurate readings of heart rhythms and better-targeted electric shocks. They are also investigating: pH sensors (that can be used to predict sudden cardiac arrest,), how to power their membrane, how to store sensor data and how to upload the information to other devices.

Sensors for the medical community are being tested both within and out of living organisms. According to [10] the future will look like the following: "Present wearable health devices include heart rate monitors, ECG monitors, glucose monitors, pulse oximeters and blood pressure monitors. In the near future, they are expected to be complemented with micro and nano chemical sensors that will provide continuous medical

diagnostics. These miniaturized smart sensors will be able to detect additional chemical signatures, e.g. in breath and sweat, which can be translated into medical monitoring (e.g. diabetes patients, other metabolic diseases, skin diseases and drug pharmacokinetics)."

These future devices will power themselves using the human body. They will transmit their findings via protocols such as Bluetooth Low Energy (BLE) and Low Rate WPAN or IEEE802.15.6. In addition the FCC has approved a specific spectrum for wireless Medical Body Area Networks (MBAN) with 40 megahertz of spectrum in the 2360-2400 MHz with the 2360-2390 MHz range for indoor use and the 2390-2400 MHz range for outdoor use. These protocols and spectrum will allow access to these devices via nearby computers and the Internet allowing distant personnel to monitor patients in their own homes.

Studies are being performed to track patients in their own homes using passive Radio Frequency Identification (RFID) tags [11 and 12]. Algorithms have been developed that either distribute the sensors around a room and signal strength measurements are made throughout the room in a grid fashion. Based upon the signal strengths measured at all the RFID sensors for each grid point, one develops a pattern of signal strengths recorded at each RFID sensor and stores the pattern in a database. Another approach [12] uses less RFID sensors and employs a path loss prediction model along with measurements to determine a tag's location. The RFID tags may be inexpensive but the time and labor to map out a home may be expensive and the algorithm's accuracy will deteriorate over time if people rearrange furniture and/or add or remove furniture.

If we integrate multiple sensors together can we reduce the number of nurses and aides that are required to monitor elderly patients who wish to live alone? Pouke, et al [13] performed a study in Finland over a two month period with monitoring the healthcare of elderly patients with varying memory diseases. They were equipped with wearable sensors to measure the patients various activities and environmental sensors to determine their location in their facility. These sensors were connected as shown in Figure 1. The patients were asked to perform normal activities such as washing dishes, cleaning their living quarters, etc. During this period data were gathered and annotated in a database indicating what the patient was performing. The researchers divided their performance into 18 classes and created avatars to mimic these classes to be viewed at the central server. These classes include for example walking, getting up, and washing their hands. The extension of this study will lead to less people monitoring the care of multiple people living in multiple locations hence bringing down the cost of caring for the elderly living alone. There are some issues that will need to be addressed such as security, battery power of sensors, cost to install and maintain, privacy, speed in which to provide aid to the patient, and the minimization of false alarms. Monitoring numerous patients while reducing cost will require the use of AI to alert the medical staff whether a patient has fallen, stopped breathing, has slept way beyond their normal time, has been in the bathroom too long, etc. We need to extend the quality of our sensors, the signal processing, and the use of AI to ultimately perform the level of care that technology can bring to the elderly.

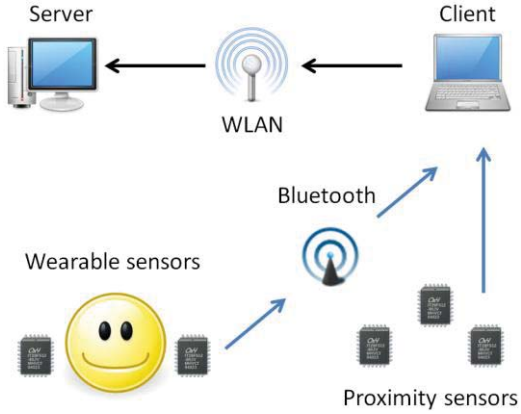


Figure 1 Abstract View of Sensor Network

Section 3 Internet of Things (IoT) and AI.

A recent paper [14] brings to light where we believe the future is headed by leveraging 5 technologies for enhancing healthcare. These are: the IoT, sensors, IPv6, AI, and Big Data. In [14] the authors develop a system design based upon the interfacing of several patient's Electrocardiogram (ECG) via a Raspberry PI (small single board computer) connected via a Global System for Mobile Communications (GSM) module which can send and receive text messages and/or voice calls. See Figure 2 for a description of their system. The system records the ECG for multiple patients, has software to analyze the ECG where it can compute heart rate, continuously monitor each patient, alert local staff via a buzzer if the continuous analysis indicates an issue with a patient, and send messages to remote personnel regarding a patient's status via the GSM module.

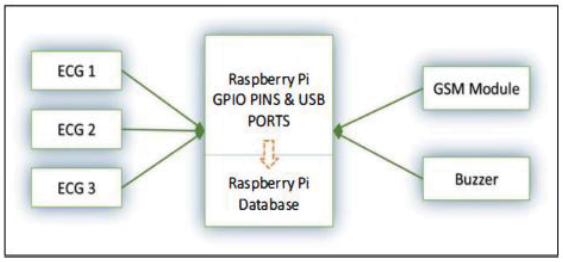


Figure 2 A Healthcare System

It is important to monitor heart patients in real time. According to [10] a monitored heart patient would have approximately 48% chance of surviving a cardiac arrest compared to a 6% chance for an unmonitored patent. What if the ECGs of these patients were analyzed in real time, beyond computing the patient's heart rate? The technology exists to retrieve these continuous readings from many sensors on a patient's body, provide alert mechanisms for extreme heart rate levels, temperature, blood sugar levels, etc. performed locally via a personal computer, similar to what was described above. However, with the advent of Internet Protocol version 6 (IPv6) we will be able to not only monitor every sensor connected to the Internet in almost real time but we will be able to activate some sensors, thereby providing healthcare via AI algorithms at various locations. (IPv6 has128 bits to identify a unique IP address or 3.4 $x10^{38}$ or 340 Undecillion available addresses.) In addition if we could capture sensor data from many people on a continuous basis we could not only look for triggers when sensor data are out of range but we could build sophisticated signal processing and AI tools to diagnose abnormalities, actuate sensors when and where needed (e.g. deliver medications) and mine data over time. This will allow one to gain more insight into solving known abnormalities but also discover new relations between measured data and patient ailments. Scientists will be able to perform cause and effect studies, analysis, monitoring, diagnosis, and prediction based upon Big Data analytics, where Big Data relates to volume, velocity, variety, data quality and provenance of data.

With the advent of the Internet, the exponential growth in deployed sensors and the need to process data in real time has caused IoT technologies to flourish. Amazon, Google, IBM and many others are building and deploying cloud-based platforms for IoT developments. There even exists a Cloud Standards Customer Council (CSCC) that is concerned about IoT and cloud computing in general. It is an end user advocacy group dedicated to accelerating cloud's successful adoption. The healthcare industry needs to leverage these technologies to bring down the cost of healthcare. We foresee the need to implement the IoT along with stream processing, AI, data mining, and distributed fault tolerant secure data storage.

In order to describe an architecture for the healthcare industry we will overlay our thoughts onto an IoT architecture. Figure 3 depicts a high level architecture acquired from CSCC [15]. This reference provides a description of all the key elements contained within a IoT architecture i.e. the User Layer, Proximity Network, Public Network, Provider Cloud and the Enterprise Network. In our design of the IoT the thing that we wish to physically monitor is the human/patient. It is the "thing" in Internet of Things. The sensors and actuators are the devices that sense or act upon the human. The instantiation of our architecture pertains to patients living for example, in their homes, day care buildings, nursing homes, and/or hospitals. We can easily extend our architecture for patients out in the environment e.g. in automobiles, ambulances, walking in the park, trains, planes, etc. (That will be covered in another paper.)

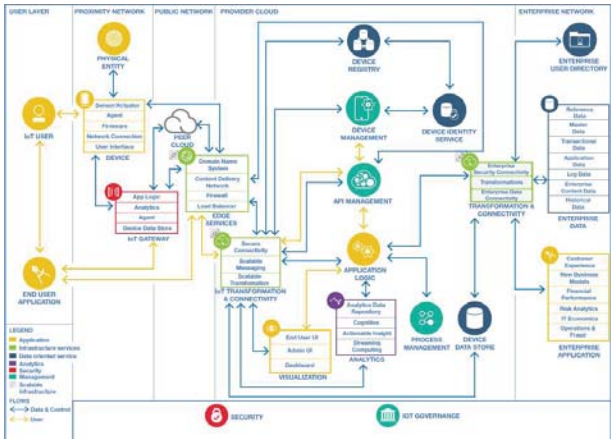


Figure 3 Detailed Components Diagram of a Cloud – based IoT Solution Architecture

In this architecture there are 3 cloud layers, the edge, the platform and the enterprise components layer.

(1)The Edge Layer: is composed of the Proximity and the Public Networks. This is where data are gathered from devices/sensors and transmitted to devices/actuators. Data flows through the IoT gateway and/or from/to the devices through the edge services to the cloud provider by way of the IoT Transformation and Connectivity. In our architecture this is where heart monitors, diabetic sensors, temperature sensors, X-Ray machines, MRI machines, etc. would interface e.g. via RF or Ethernet enabled devices to the local area network where processors will analyze results and pass it onto the cloud or process the data to determine

if the sensors are out of range and an alarm needs to be sounded for the local nurse to be summoned or call 911, etc. The local processing machine could be a local PC or a Raspberry Pi computer as described above. Here resides the first line of defense software and AI algorithms to assess the status of the patient's health. It may integrate more than one sensor or actuator to determine for example if a patient has fallen, requires more insulin, etc. In addition the computing device will have a user interface so that the healthcare provider can review past data, change settings, print reports, send messages and/or data, and manage the sensors and actuators monitoring the patient. The data collected from these sensors and nodes are passed on to the next cloud layer. This Edge Layer of processing is sometimes called edge or fog computing. This approach is relevant for a large nursing home or hospital.

For smaller institutions or for elderly patients living alone the architecture shown in Figure 4 from [2] may be more appropriate where alarms and diagnosis are performed in the platform layer as opposed to the Edge Layer. In either case the system should be built with fault tolerance and redundancy in mind, employing backup generators, multiple Internet connections and eliminating any single points of failure, regardless of whether the alarms and diagnosis are performed at the edge or in the cloud. The driving factor is best stated by [2] "Intervention response time is paramount, since it is well known that time is of the essence in treating falls or heart attacks. This is true not just to avoid fatalities, but to minimize the secondary damage that might be difficult to treat. It is also important to learn the nature of the incident as soon as possible: whether the person is conscious, their exact location, etc."

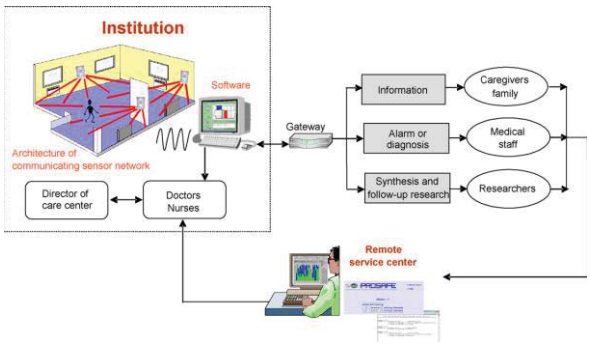


Figure 4. A IoT Architecture Design [2]

(2) The Platform Layer: is the provider cloud. This is where the data is received from the edge layer, processes and analyzes the data and provides application programmers interface (API) Management and Visualization for the total process. It also provides the connection between the public network and the enterprise network. In our design we foresee that the processing performed at the edge will also be performed in the platform layer for redundancy in case one fails since this is the crucial point or first line of care, alarm and diagnosis. This layer will provide real time analysis and diagnosis where the data from all the relevant sensors will be sent to numerous processors that will analyze the data using stream processing.

Conventional processing acquires data from sensors, medical devices, human input, or outside sources, stores these data in a database management system (DBMS) and then searches and processes these data to acquire information. However, given the severity of identifying and treating patients in real time we must process the data before storing it in a DBMS especially when considering that our architecture is monitoring multiple sensors on multiple patients in multiple homes, hospitals and/or nursing homes, or multiple ACOs. Stream processing is a byproduct of Big Data and the need to answer pressing questions in shorter time periods. Probably some of the first stream processing systems were developed for digital signal processing e.g. radar systems, where the data are continuously arriving and each return must be filtered, targets detected, tracks formed, targets classified, and identified. Imagine the healthcare domain where hundreds of patients have multiple sensors/actuators monitoring their every move 24/7. The data are encrypted and streamed to numerous processors or operators in the cloud where they are buffered slightly and operated upon and moved on to the next operator. The stream processing algorithms are operating on data from numerous and various sensors such as ECG sensors, location sensors, temperature, blood pressure, diabetes, etc. Our design is to stream all of these data and operate upon them such that we can detect if they require more insulin and how much, or whether they have fallen, or are having a heart issue and if so to take the proper action.

There are many research areas that need to be addressed to make this happen from optimization of stream processing [15], to pattern recognition in real time, to natural language processing, to diagnosing different heart issues, to AI algorithms for merging the signals from numerous sensors, to identifying that a patient has injured themselves in the kitchen while boiling water. Figure 5 shows the basic features of an ECG and images of T Wave abnormalities. By embedding AI, signal processing, and stream processing we should be able to determine T Wave abnormalities on the fly. Work is going on using AI, signal processing and knowledge bases where software can diagnose many heart ailments when a

physician is not available. The future is to automate this capability so we can outperform the physician by using stream processing on numerous heterogeneous sensors along with searching the history of the patient, contained in the enterprise network, all done in near real time. It should be noted that engineers have been embedding AI in signal processing for radar and communications systems for years under the names of cognitive radar and radio. As examples see [17 & 19]

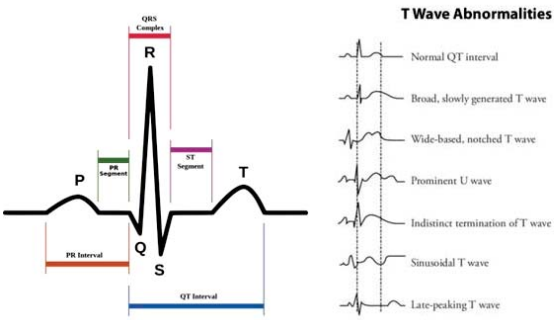


Figure 5. ECG Diagrams

3. The Enterprise Layer: consists of Enterprise Data, Enterprise User Directory, and Enterprise Applications. It is represented as the Enterprise Network. This layer communicates and exchanges data, such as the real time streaming data, from and to the Platform Layer via the Transformation and Connectivity component. In our architecture the enterprise layer contains the storage of all the data contained from all the sensors within our IoT along with structured and un-structured data related to the medical community such as audio and video files and connections to other structured files such as Electronic Medical Record (EMR) systems that reside on multiple computers in the cloud. These data are sometimes called the system's data warehouse (DWH). The data from the stream processing need to be stored for data mining and modeling. Selected data will be stored in a patient's EMR system e.g. OpenEMR, Centricity, CareCloud. The data from the stream processing are normally stored in Hadoop, an open source programming framework for the processing of large data sets in a distributed computing environment. Hadoop, or a variant, will be needed as a computational secure system for data mining and analytic processing. See [18] for an overview of different architectures for the processing and storing of Big Data arising from stream processing. We also need to be assured that the data are encrypted within the distributed system, as per Health Insurance Portability and Accountability Act (HIPAA), and may require the integration of Tahoe-LAFS, which is an encrypted fault tolerant secure method of storing files on distributed processors.

4.0 Summary/Conclusions

We have provided a motivation of our work via a description of our aging population and the rise in cost of our healthcare system within the USA. We have provided an overview of body area sensors and actuators being used and studied today. To help bring costs down and to enhance the quality of care we have described an IoT driven architecture including edge computing at the sensor level, stream processing, distributed computer processing, EMR systems, AI, and a encrypted fault tolerant secure method of storing files. The described architecture will provide better and cheaper healthcare and allow researchers access to data that are not available today to further their understanding of the human body and its abnormalities. The architecture presented here must be studied further and developed as a test bed for medical and engineering researchers.

.

References

[1] http://www.aoa.acl.gov/aging_statistics/index.aspx
[2] Chan, M, Esteve, D., Escriba, C., and Campo, E., "A Review of Smart Homes – Present State and Future Challenges", Computer Methods and Programs in Biomedicine 91 (2008) 55-81.
[3] https://www.jimcarroll.com/2011/02/the-future-of-seniors-care-assisted-living-big-trends-or-crazy-ideas/#.VzjkJ77mCUm
[4]http://www.aetna.com/health-reform-connection/aetnas-vision/facts-about-costs.html#breakdown
[5] https://www.healthcatalyst.com/what-is-an-ACO-definitive-guide-accountable-care-organizations
[6]Latre, B., Braem, B., Moerman, L. and Blondia, C., "A Survey on Wireless Body Area Networks", Journal of Medical Systems, August 2010 https://www.researchgate.net/profile/Ingrid_Moerman/publication/45709672_A_Comprehensive_Survey_of_Wireless_Body_Area_Networks/links/53e8ac820cf2fb74872448d4.pdf
[7]http://newsroom.heart.org/news/mobile-technology-may-help-people-improve-health-behaviors
[8]http://www.medpagetoday.com/Cardiology/Arrhythmias/27873
[9]http://www.the-scientist.com/?articles.view/articleNo/39264/title/Next-Generation--Sensor-Laden-Sheath-to-Monitor-the-Heart/
[10] Fernandez, F. and Pallis, G., "Opportunities and Challenges of the Internet of Things for Healthcare", National Conference on Wireless Mobile Communication and Healthcare, 2014.

[11] Ting, S., Kwok, S., Tsang, A., and Ho G., "The Study on Using Passive RFID Tags for Indoor Positioning", International Journal of Engineering Business Management, Vol.3, No. 1, 2011.
[12] Dao, T, Nguyen, Q., Ngo V., Le M., and Hoang, C., "Indoor Localization System Based on Passive RFID Tags", 2014 Fifth International conference on Intelligent Systems, Modelling and Simulation", IEEE Computer Society, 2014
[13] Pouke, M., and Hakkila, J., "Elderly Healthcare Monitoring Using an Avatar-Based 3D Virtual Environment", International Journal of Environmental Research and Public Health, 2013.
[14] Gupta, M., Patchavaa, V., and Menezes, V., "Healthcare Based on IoT Using Raspberry Pi:, 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015.
[15]http://www.cloud-council.org/deliverables/CSCC-Cloud-Customer-Architecture-for-IoT.pdf
[16] Hirzel, M, Soule, R, Schneider, S., Gedik, B., and Grimm, B., "A Catalog of Stream Processing Optimizations", ACM Coputing Surveys, Vol. 46, No. 4, Article 46, March 2014.
[17] Capraro G., Farina A., Griffiths H., and Wicks M., "Knowledge-based radar signal and data processing: A tutorial overview," IEEE Signal Processing Mag., vol. 23, no. 1, pp. 18–29, 2006.
[18] http://www.infoq.com/articles/stream-processing-hadoop
[19] Haykin, S. "Cognitive radio: brain-empowered wireless communications", IEEE Journal on Selected Areas in Communications, Vol 23, Issue 2, February 2005.

Predicting Insurgent Activity with Biased Lévy Flights

A. Wheeler[1] **and M. Winburn**[2]
[1]EHGO Analytics LLC, Winter Park, FL, USA
[2]3 Sigma Research Inc., Indialantic, FL, USA

Abstract - *This research seeks to test the validity of biased Lévy flights in the context of insurgent activity in Afghanistan during Operation Enduring Freedom from 2001 to 2011. A Lévy flight results in a sequence of frequent short moves with progressively less frequent longer moves. Location bias comes from both intrinsic attraction and from the broken window effect. We use poppy cultivation and distance to a major city to model intrinsic attraction. We use a cellular automaton to model the movement of insurgents over space and time. Results of our simulations show good agreement with the actual pattern of coalition casualties observed over the time period. Furthermore, biased Lévy flights perform better than random walks.*

Keywords: Lévy flight, cellular automata, crime modeling

1 Introduction

The U.S. military continues to improve the efficiency and effectiveness of their mission planning by identifying locations of interest with high probability of high value targets. This enhanced battlespace awareness will come through processing of diverse spatially referenced data sources (environmental, social, cultural), automated feature extraction, data normalization, space-time pattern recognition, and visualization.

Previous research [1] proposed to address this goal through a biologically inspired cellular automaton model of human search for attractive locations based on biased Lévy flights. This idea came from the application of crime analysis to insurgency and counterinsurgency [2] and work done by scientists at UCLA in crime mapping who 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 biased Lévy flights [3][4].

This research seeks to test the validity of biased Lévy flights in the context of insurgent activity in Afghanistan during Operation Enduring Freedom from 2001 to 2011. We also compare the biased Lévy flight model to a random walk model.

This research makes a number of significant benefits and contributions. First, it provides empirical support for biased Lévy flights as a model for some types of criminal activity. Second, it shows that insurgent activity has predictable similarities with domestic criminal activity. Finally, 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.

2 Background

The motivation for this research comes from previously published work [1] to model and predict insurgent activity using a mathematical model of search and foraging called Lévy flights. Lévy flights consist of sequences of independent, randomly oriented steps with lengths l drawn at random from a probability distribution function having a power-law tail, $p(l) \sim l^{-\mu}$ where $1<\mu<3$. A Lévy flight results in a sequence of frequent short moves with progressively less frequent longer moves. This pattern makes Lévy flights farther distributed from the origin than random walks (Brownian). Lévy flights perform better than random walks when searching for scarce and randomly located resources or targets. [5]

Patterns of Lévy flight movement appear in the behavior of at least 14 species of open-ocean predatory fish, including sharks [6]. Honey bees searching for their hive exhibit search patterns that fit those described by Lévy flights. This behavior manifests itself by honey bees moving quickly in one direction then searching for landmarks indicating the direction of their hive [5]. Scientists have found evidence of Lévy flights in human movement as well, such as human hunter-gathers using Lévy flights when searching for food [7] and criminals searching for opportunities to commit crimes [4].

2.1 Model

The bias in the cellular automaton model comes from the differential attractiveness of cells. The attractiveness of a site *s* results from and inherent attraction and repeated victimization (broken window effect) [3].

$$A_s(t) = A_s^0 + B_s(t) \quad (1)$$

We have the probability of an insurgent moving from site *s* to *n* as

$$q_{s\to n}(t)=\frac{A_n(t)}{\sum_{s'\neq s} A_{s'}(t)} \quad (2)$$

Dynamic attractiveness that comes with repeated victimization with ω a decay term, E(t) the number of insurgent events at time t, and θ the impact of insurgent events on site attraction.

$$B_s(t+\delta t)=B_s(t)(1-\omega\delta t)+\theta E_s(t) \quad (3)$$

Now we add the near-repeat victimization of the broken windows effect with z the number of neighbors and η the significance of the neighborhood effect.

$$B_s(t+\delta t)=\left[(1-\eta)B_s(t)+\frac{\eta}{z}\sum_{s'\neq s} B_{s'}(t)\right](1-\omega\delta t)+\theta E \quad (4)$$

New insurgents appear randomly at the rate Γ .

Lévy flights enter the model by including distance d in the attractiveness of sites [4].

$$a_{s\to n}(t)=A_n(t)\,d_{s,n}^{-\mu} \quad (5)$$

Now we have the probability of an insurgent moving from site s to n.

$$q_{s\to n}(t)=\frac{a_{s\to n}(t)}{\sum_{s'\neq s} a_{s\to s'}(t)} \quad (6)$$

Insurgents commit violent acts with probability

$$p_s(t)=1-e^{-A_s(t)\delta t} \quad (7)$$

The following steps describe the cellular automaton event loop [3].

1. Initialize the current state
2. Set $E(t)=0$
3. For each insurgent
   a. Compute the probability of committing an attack
   b. If an attack occurs then increment $E_s(t)$ and remove the insurgent
   c. Else compute q and move to a new site
4. For each site
   a. Update B_s
5. Place new insurgents
6. Repeat 2

3 Data and Methods

This section describes the data and relevant details of our implementation of the cellular automaton model of biased Lévy flight for identifying locations of interest. We use GRASS GIS [8] to create and manage the geographic data. Implementing the cellular automaton model, running the simulations, performing the analysis, and mapping are done with R [9].

3.1 Data

Data needed for our research includes Afghanistan province and district boundaries [10], locations of the sixty largest cities in Afghanistan [11], coalition fatalities for 2001-2011 by province [12], district populations for 2009 [13], and mean opium production 2001-2010 by district [14]. We cleaned the data to correct coding errors (e.g. negative values) and normalize differences in the spellings of provinces and districts. We compute derived data using GRASS GIS or R.

We use the average poppy cultivation over all years to address missing data for districts and specific years. Government eradication efforts target individual districts and sometimes production resumes later [14].

3.2 Methods

3.2.1 Cellular automaton grid

Instead of a uniform grid of cells typically found in traditional cellular automaton, we use administrative districts, equivalent to US counties, as cells. This makes physical sense in our case because we have data at the district level. We could rasterize the data to create a uniform grid of cells; however this would increase the number of cells to evaluate at each iteration without providing any advantage to spatial resolution.

Adjacent districts determine the neighborhoods, so cells can have varying numbers of neighbors. Furthermore, distances to each neighbor and to each cell in general fluctuate according to distances between centroids (Figure 1).

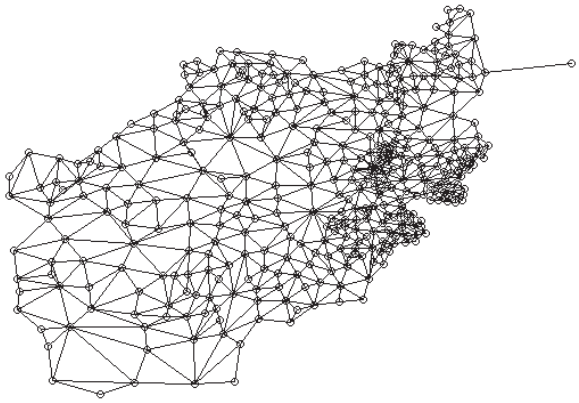


Figure 1. District centroids and adjacency.

3.2.2 Intrinsic site attraction

According to [2], insurgents showed the most activity in areas where they hoped to grow poppies to finance their organizations. Poppy cultivation proves most lucrative to

farmers far from markets (e.g. large cities) because poppy purchasers would travel to them [14][15]. Furthermore, areas far from large cities make it easier for insurgents to operate because these areas often have less national governance. Thus, we consider both poppy cultivation and distance to the nearest major city as potential predictors of intrinsic site attraction.

A linear regression model shows (Table 1) both poppy cultivation and distance to a major city significant. Furthermore, the coefficients have the expected relation to the number of casualties, with poppy cultivation negative and distance to a major city positive.

Table 1. Linear regression predictors of coalition casualties.

Variable	Coefficient	Std Err	t value	Pr(>|t|)
(Intercept)	102.103	8.715	11.716	<2e-16
Poppy cultivation	-18.282	8.738	-2.092	0.0371
Distance to city	64.627	8.738	7.396	8.47e-13

Based on the results of our regression model, we define the intrinsic attraction A_s^0 of each cell (district) s by a linear function

$$A_s^0 = aD_s + bP_s \quad (8)$$

with D_s the scaled distance from the district centroid to the nearest large city and P_s the scaled poppy production in district. We set the coefficient of our model to a = 65 and b = -18.

3.2.3 Model parameters

Interestingly, we found the parameter values similar to those suggested in the original studies [3][4] to work very well in predicting the coalition casualties.

$$\begin{aligned} \Gamma &= 10 \\ \delta t &= 0.05 \\ \eta &= 0.2 \\ \mu &= 3 - \frac{2}{\log 10\left(\frac{1}{\delta t}\right)} \\ \omega &= \frac{1}{15} \\ \theta &= 0.56 \end{aligned}$$

We iterate until we have 3028 casualties, the same as the total number of casualties in our data. This has the advantage of providing a large number of iterations, and gives a large enough number of casualties to compare proportional spatial distributions between actual and simulated casualties.

4 Results

So many factors contribute to actual coalition casualties, or to most types of insurgent activity, that predicting actual numbers does not seem reasonable. However, in terms of prioritizing coalition efforts, it suffices to predict the relative activity levels at locations. We use heat maps to visualize relative activity levels, with hot (lighter) indicating high activity and cold (darker) for low activity. A heat map of actual coalition casualties over the period 2001-2011 appears in Figure 2. We use this map to compare the performance of the random walk model and the biased Lévy flight model.

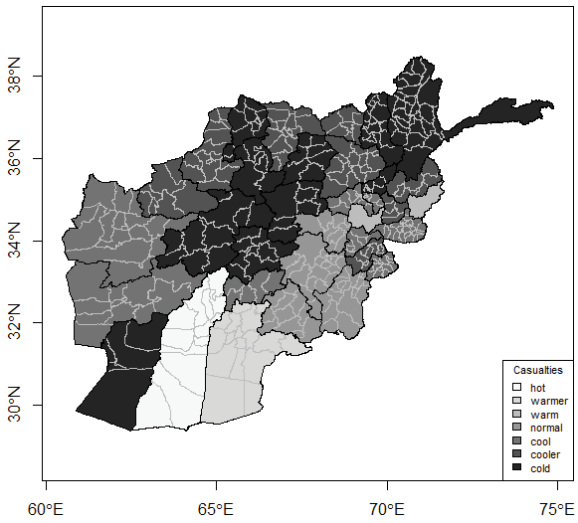


Figure 2. Total coalition casualties (2001-2011).

Figure 3 shows the results of a typical random walk simulation at the district (cell) level. For comparison with total coalition casualties (Figure 2) we aggregate district predictions to provinces and then generate a heat map (Figure 4).

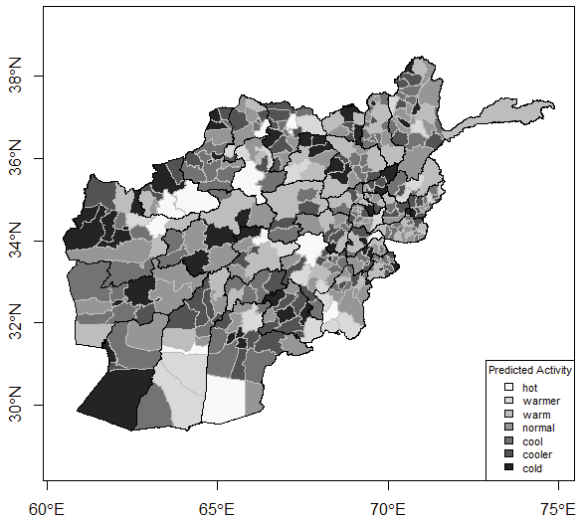


Figure 3. Predicted coalition casualties with random walks by district.

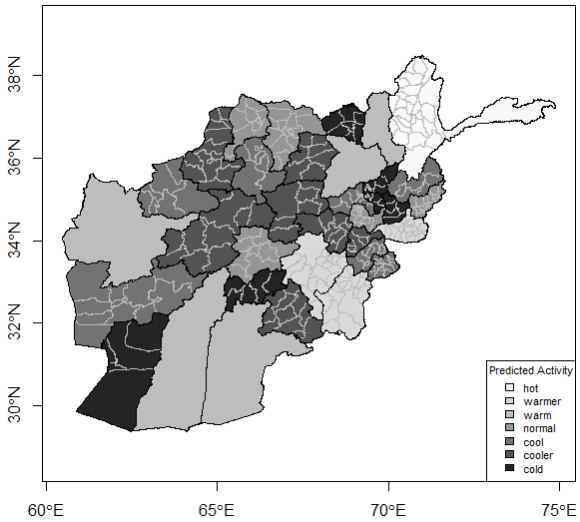


Figure 4. Predicted coalition casualties with random walks by province.

Figure 5 shows the results of a typical biased Lévy flight simulation at the district (cell) level. For comparison with total coalition casualties (Figure 2) we aggregate district predictions to provinces and then generate a heat map (Figure 6).

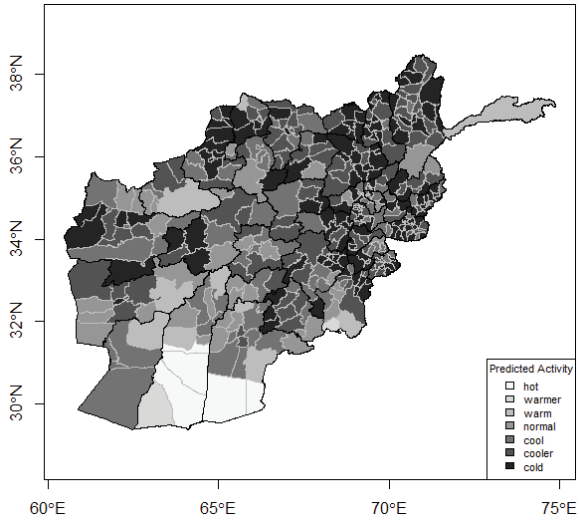


Figure 5. Predicted coalition casualties with biased Lévy flights by district.

We see by comparing the random walk model (Figure 4) and the biased Lévy flight model (Figure 6) with actual coalition fatalities (Figure 2) that the biased Lévy flight model appears better at predicting the provinces with the highest coalition casualties from district-level data.

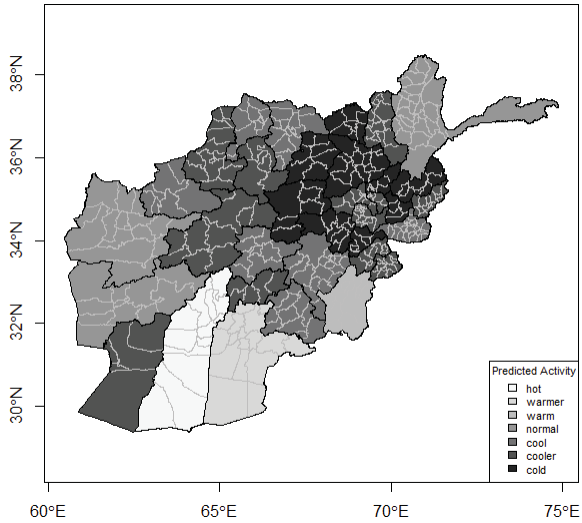


Figure 6. Predicted coalition casualties with biased Lévy flights by province.

5 Conclusions

We performed simulations of insurgent activity in Afghanistan with their movements governed by random walks or biased Lévy flights using cellular automata with cells corresponding to administrative districts. We compared heat maps generated at the province level with actual coalition casualties during the years 2001-2011 and found that the biased Lévy flight model better predicts the relative pattern of high activity. Furthermore, these results support the idea that insurgents travel locally but occasionally travel long distances to find targets.

Regarding future work, we hope to obtain more specific geographic data with which to calibrate our cellular automata models so that we can identify more specific geographic areas as locations of high levels of insurgent activity. We would like also to extend our models, which have assumed a free ranging pattern for insurgent activity, to incorporate central place behavior where insurgents operate from a fixed location and search out in all directions for targets.

6 References

[1] A. Wheeler (2015) Predicting Locations of Interest with Biased Lévy Flights. 17th International Conference on Artificial Intelligence (ICAI'15), 27-30 July 2015, Las Vegas, NV.

[2] 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

[3] 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

[4] 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

[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] 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

[7] D.A. Raichlen, B.M. Wood, A.D. Gordon, A.Z.P. Mabulla, F.W. Marlowe, and H. Pontzer (2013). Evidence of Lévy walk foraging patterns in human hunter–gatherers. PNAS, volume 111(2), pp. 728-733. http://www.pnas.org/content/111/2/728.full

[8] GRASS GIS. OSGeo Project. http://grass.osgeo.org/

[9] The R Project for Statistical Computing. http://www.r-project.org/

[10] Afghanistan Information Management Service. http://www.aims.org.af/ssroots.aspx?seckeyt=295

[11] (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

[12] Operation Iraqi Freedom and Operation Enduring Freedom Casualties. http://icasualties.org/

[13] Districts of Afghanistan. Statoids. http://www.statoids.com/yaf.html

[14] Afghanistan Opium Survey 2010. United Nations Office on Drugs and Crime. https://www.unodc.org/documents/crop-monitoring/Afghanistan/Afghanistan_Opium_Survey_2010_web.pdf

[15] Poppy growing season begins in Afghanistan after last year's record opium sale. The National, November 14, 2013. http://www.thenational.ae/world/poppy-growing-season-begins-in-afghanistan-after-last-years-record-opium-sale

Adaptive Workspace to Reduce Analyst Information Overload

Bruce McQueary[1], Anton DeFrancesco[1], Lynn Lehman[1],Roger Dziegiel Jr[2]
[1]Securboration, Inc., 1050 W. NASA Blvd, Suite 155, Melbourne, FL 32901
[2]AFRL/RIED, 525 Brooks Rd, Rome, NY 13441

Keywords: Adaptive Workspace, Cognitive Workspace, Data Immersion, Intelligent Offloading, Intelligence Automation

Abstract – *Unstructured open source data is ubiquitous and growing at exponential rates, far surpassing the growth of traditional structured (e.g. relational) data sources. This includes multi-lingual and multi-modal (e.g. video, audio, imagery) sources that have the potential to be exploited, whether it's to evaluate nuclear capabilities of a rogue nation, or to evaluate the likelihood of a company going public. However, the ability to exploit this information explosion has not kept pace. Current analysis tools still put the burden on the analyst to identify relevant data while serving more like information management systems instead of true cognitive contributors to the analysis process. This paper addresses the development of a system referred to as the Adaptive Data Immersion ENvironment (ADIEN), whose goal is to provide a true bi-directional human-IT partnership that includes flexible, adaptive processes like data reduction, 'what-if analysis' and other creative human activities that require 'thinking outside the box.'*

Keywords: analyst workspace, text analytics, task recognition, ontology

1 Introduction

The first recorded use of the phrase "information overload" was used by the futurologist Alvin Toffler in 1970, when he predicted that the rapidly increasing amounts of information being produced would eventually cause people problems[1]. Fast forward to the present day – anyone has the ability to generate information in many forms and distribute it instantaneously to an unlimited audience on the web. If one is an analyst in any number of professions this clearly causes challenges. No longer is the analyst just looking for the proverbial needle in the haystack – they must first find the *right* haystack in a near-infinite field of dynamic haystacks in which to execute their analytical process.

The tools and workspaces at the analyst disposal have not kept pace with the growth of data. There are tools to support multilingual Optical Character Recognition (OCR), Machine Translation (MT), Automated Speech Recognition (ASR), and natural language processing (NLP e.g. entity extraction) but the limiting factor is still the reliance on an already overburdened human analyst to discover the relevant or actionable information nuggets. The search and gather loop to 'connect the dots' is complex and requires time and effort to constantly build and revise mental models of the situation. This takes a cognitive toll on the analyst. Cognitive fatigue sets in and expectancy behavior[2] surfaces, in which analysts increase their tolerance for error, delay analysis, shed tasks and prematurely filter information, to arrive at conclusions and closure that fit their perceptions without considering alternate hypotheses. The work described in this paper aims to help reduce the cognitive load on the analyst and reduce information overload by through automation. This involves a continuous, adaptive interaction between ADIEN and the analyst, in which ADIEN is not simply learning and executing repetitive tasks from well-defined standard operating procedures (SOPs), or tactics, techniques, and procedures (TTPs). ADIEN is operating in a true human-IT partnership performing critical duties for the analyst; much like paralegals function as indispensable partners for lawyers in the legal community. As a result, ADIEN complements the analyst's abilities, increases their agility, alleviates cognitive fatigue, allows more data to be fully discovered within a timeframe that it is still valuable, while minimizing the analyst's skill-based slips, rule-based mistakes, and knowledge-gaps.

1.1 ADIEN Project Overview

Figure 1 depicts ADIEN's high-level function and benefits.

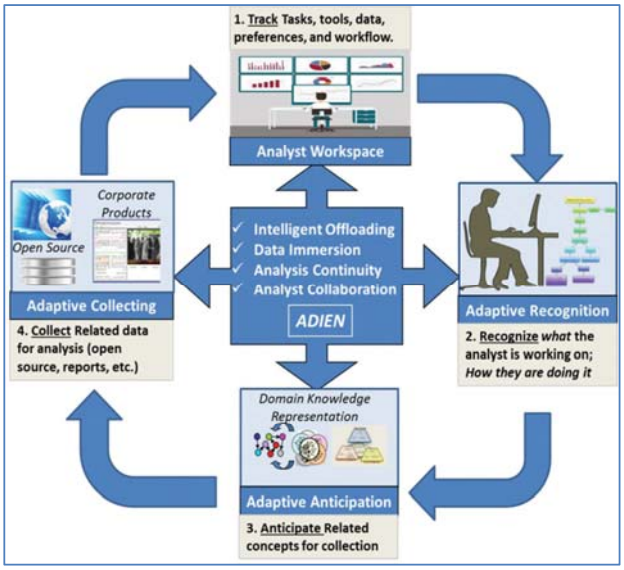


Figure 1. ADIEN Function and Benefits

As shown in Figure 1. ADIEN can be thought of in terms of the cyclic capabilities of: *tracking, recognizing, anticipating,*

and *collecting*; and in terms of the benefits: *intelligent offloading; data immersion, analyst continuity,* and *analyst collaboration.* Each are discussed as follows.

1. Track. ADIEN integrates with existing analyst' frameworks and tools to track the analyst's ongoing activities. This includes non-intrusive integration techniques, such as application extensions, and web services.
2. Recognize. This focus is to understand what the analyst is doing and why. As the system tracks analyst activity, it begins to learn the analyst direction and preferences. For example, the system can learn that when an analyst is evaluating a stock, the first thing they do is search for news articles related to increases in manufacturing. This supports the adaptive anticipation and collection.
3. Anticipate. Here, the system proactively broadens the scope of the analysis to bring in related concepts to the current analyst focus. For example, the system can broaden manufacturing and search for transportation related data. This brings in broader (outlying) domain knowledge to identify cascading related concepts that the analyst might be interested in exploring.
4. Collect. The identified related concepts and themes drive additional collection processes that reach into the global data space, which includes open source, existing products and corporate / enterprise data and resources as well.

This process involves a continuous, adaptive interaction between ADIEN and the analyst; resulting in the following benefits.

1. Intelligent offloading – this is enabled by the recognition to understand 'what' the analyst is doing and 'why' to proactively provide assistance to help satisfy intelligence requirements. This includes bringing data to the forefront based on an analyst's preferences, such as knowing that analyst 'A' is a heavy user of financial releases, and analyst 'B' generally reviews news feeds.
2. Data Immersion – provide analysts with a wide field of view of the data landscape, allowing them to navigate, test, and evaluate hypotheses and to discover data 'outliers,' which may lack in traditional statistical relevancy but warrant consideration. This capability is critical for next generation workspaces because "major intelligence failures are usually caused by failures of analysis, not failures of collection. Relevant information is discounted, misinterpreted, ignored, rejected, or overlooked because it fails to fit a prevailing mental model or mind-set."
3. Continuity – maintain the analysis state, allowing analysts to take necessary breaks and seamlessly continue where they left off at a later time. Analysts tend to achieve a 'Zen-like' state and are reluctant to disengage from the analysis process. The result is that analysts often become fatigued and produce analyses that are susceptible to errors. ADIEN continually works to learn analyst preferences, as well as the concepts or path the analyst is exploring, and maintains the state of the analysis to enable the analyst to continue where he/she left off. Additionally, since ADIEN is not simply reacting to the analyst input, but understands the analyst overall intent, it can to continue to identify, queue, and process relevant data on behalf of the analyst while they are offline.
4. Collaboration - innovate the means for explicit and implicit collaboration across analyst networks. ADIEN can be used as the basis for facilitating collaboration by tracking who is doing what and the progress that has been made to date. For example, implicit collaboration is established when analyst 'B' creates metadata for a video that is relevant to analyst 'A'. A link may be established among those analysts based on this shared information. This link may be exploited explicitly as well. For example, an analyst with a new business intelligence requirement can manually query to identify analysts working on related topics. Or the collaboration may be implicit, such as ADIEN identifying related video products and automatically queuing them for the analyst based on their profile and their current tasks.

2 Architectural Concepts

Figure 2 illustrates the high-level ADIEN architectural concepts.

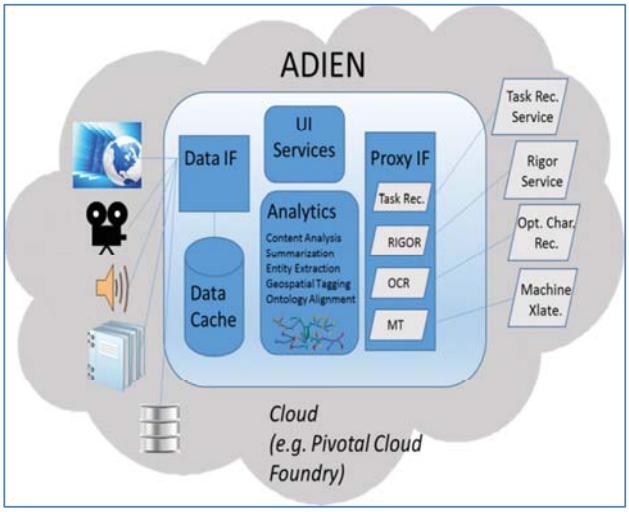


Figure 2. Architectural Concepts

Central to ADIEN's theme is leveraging third-party cloud services. There are a wealth of services that provide capabilities such as machine translation, automated speech recognition, and optical character recognition. Currently, the main services ADIEN leverages are those for Rigor Analysis[3] and Workflow Activity and Recognition[4]. Rigor Analysis can be thought of as the process of continually 'analyzing the analysis' for the purposes of ensuring that factors such as bias do not creep in. For example, if the user continually gets documents from a particular news source, an analytical rigor service can recommend diversifying the data sources. Rigor services provide recommendations that can include additional documents to search; formulating new hypotheses; and identifying gaps between what was actually done versus a

prescribed or "standard method". Workflow Activity and Recognition is the process of recognizing what the analyst is working on and the direction they are headed. The goal is to provide a proactive capability that anticipates the analyst's next-state information needs; to provide that information; and to do so in the manner and form that the analyst prefers (e.g. providing information that visual, like images vs. textual reports). As vendors provide more capability to cloud architectures, ADIEN will be in a position to leverage them.

3 Analyst Workspace Concepts

A primary goal for ADIEN is to support the cognitive processes that analyst go through when performing their job. Consequently, ADIEN is designed to be workspace centric. Each workspace represents a question or hypothesis(es) that the analyst is working on. For example, an analyst might be assigned to answer the question: *will the rocket and spacecraft manufacturer SpaceX go public by 2020?* Obviously, this consists of many hypotheses that must be addressed, and each will need to be supported (or refuted) by multiple document sources. Additionally, the analyst will create individual products consolidating their thoughts with respect to the hypothesis and the evidence presented throughout the process. While in theory this is straightforward, in reality it is a complex progression that includes detailed analyst checklist and invocation of multiple tools (e.g. entity extraction to detect Equity funding; speech translation to evaluate comments from SpaceX founder Elon Musk, etc.).

An illustrative object model depicting the relationships among classes of objects is shown in Figure 3. Each workspace will consist of a set of analyst-entered related hypotheses. The workspace is also driven by topics (Analyst areas of Interest) which serve to broaden and narrow the information retrieval process. Hypotheses, in turn, may contain sub-level hypotheses which further help to focus analysis. Ultimately, the workspace and hypotheses link to work products that the analyst generates; all while maintaining the critical linkage back to the originating artifacts that the analysts drew their conclusions from.

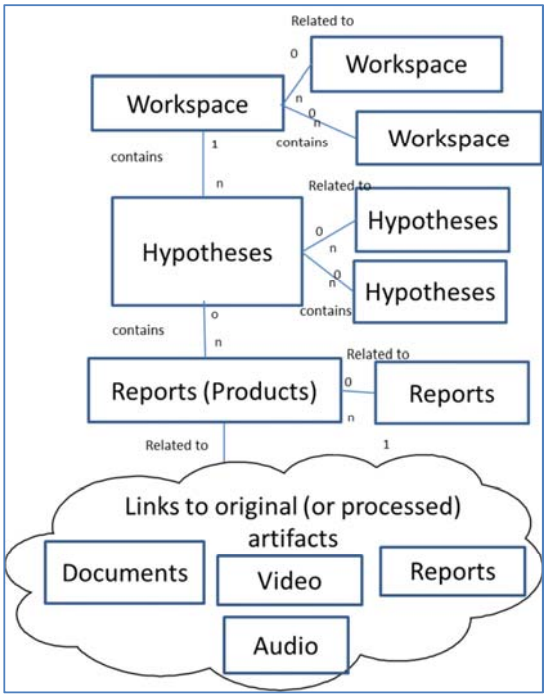


Figure 3. Workspace Object Relationships

4 Analyst Workspace UI

The Analyst Workspace and Architectural Concepts have been codified in the Workspace shown in Figure 4.

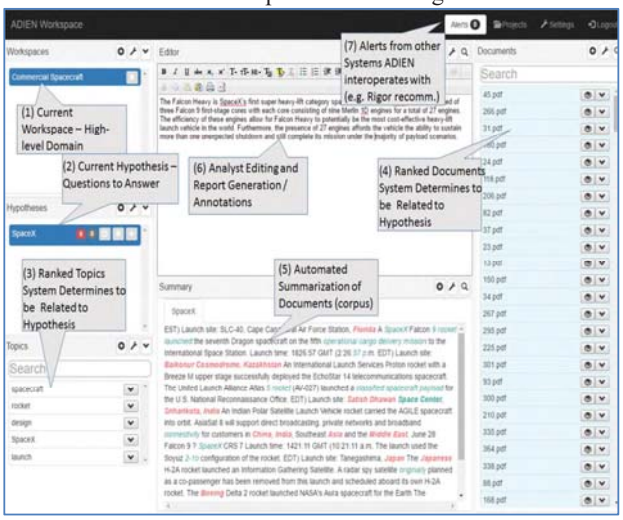


Figure 4. ADIEN Current Workspace

The standard workflow for an analyst using ADIEN is for the analyst to (1) create the workspace (2) create hypothesis(es) (3) when these are created ADIEN's text analytics processes the hypothesis and determines related topics from it, (4) which in turn ADIEN uses to perform and initial data query to find related documents (5) ADIEN then automatically summarizes

the corpus of documents (6) the analyst begins generating their report / notes based on the documents and summaries; and (7) behind the scenes, ADIEN is communicating with Rigor and Workflow services, these provide alerts to the analysts, such as Workflow providing the results of an automatically located and translated speech.

5 Workflow Recognition

As described the analysis process is complex. Analyst currently spend a large majority of their time performing mundane search and retrieval tasks; iteratively broadening their search using general concepts (e.g. Equity Funding in space initiatives) to the specific instances (e.g. Founders Fund, Draper Fisher Jurvetson and Valor Equity Partners). This is a process that lends itself well to proactive automation where by the system can understand key concepts the analyst is working on and stay a step-ahead in terms of accessing and preprocessing data for the analyst; and ranking it with respect to their context. For example, the system can locate speeches from Elon Musk; perform automatic speech translation; and process those for future business indicators – and present the results to the user. This approach falls into the realm of Workflow and Task Recognition. The Task Recognition Service that ADIEN initially leverages is based on SRI's Workflow Activity and Recognition Processor (WARP)[5,6]. WARP provides technology to recognize user intent in real time by matching observations of user actions and state to predefined models of workflows. WARP workflows are an extension of Hidden Markov Models (HMMs), which are a form of statistical model used widely for applications that require temporal pattern matching. HMMs (and, hence, WARP workflows) are described as graphs, composed of nodes and edges between them. These graphs contain two types of nodes, namely hidden states and observations (Figure 5).

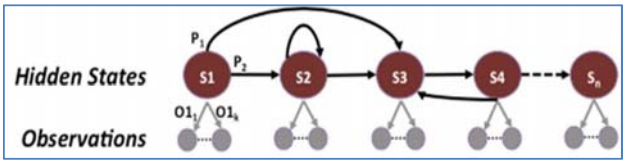


Figure 5. Workflow HMM Graph

The hidden states (also referred to as steps) are not directly observable—only the observations. The observations, which are generated as a consequence of the process being executed, are used to draw inferences about steps that have already been performed and likely next steps. This approach differs from more straightforward procedure monitoring, where an observation automatically determines whether a step has been executed. In particular, observations may be unreliable and possibly associated with multiple steps, thus complicating the determination of the step that has been performed. For example, an analyst performing a query that involves the term "low radar profile materials" could be indicative of a step to research stealth technology or of a step to investigate radar capabilities.

Directed edges between hidden states represent temporal ordering between states, i.e., that one state precedes another. Directed edges from a state to one or more observations indicate that the state is expected to generate those observations. Probabilities can be associated with both types of edges to reflect prior knowledge of distributions over those relations. In Figure 5, for example P1 and P2 correspond to the probabilities of the two possible transitions from state S1; similarly, O11 through O1k are the probabilities of the corresponding observations be generated when S1 occurs.

WARP reasons about observations to track progress through a workflow, using probabilistic inference to resolve ambiguities and uncertainties. In particular, WARP performs two types of inference:

- Filtering: what is the current state and how likely is it?
- Prediction: what is the most likely next state?

As with HMMs, WARP workflows exhibit the Markov property, which means that the possible next states and the set of observations depend only on the current state. WARP workflows differ from HMMs through their use of variables to track attributes and values associated with states (i.e., so-called Logical HMMs[7]).

6 Content Analysis

While ADIEN can leverage analytic capability by any service provider in the cloud, the primary analytic component within ADIEN revolves around providing the most relevant information to the analyst in a timely manner – minimizing their search and gather loop to the maximum extent possible. This is heavily dependent on automatically analyzing the content of documents, and performing different functions on them, such as ranking with respect to current topic; summarizing a corpus of relevant documents; extracting entities, etc. To accomplish this, ADIEN leverages Securboration's suite of Content Analysis tools, providing text extraction capability far beyond typical natural language processing (e.g. GATE) implementations. Content analysis is comprised of different components: Document Modeling, Topic Identification, Entity Extraction, Ontology Alignment, Geotagging, Topic Ranking, Clustering, and Summarization. Figure 6 is a 'metro map' depiction of the different tools and capability provided by the Content Analysis library and Figure 7 shows the workspace that is used to interactively compose these elements into a pipeline to support individual project needs.

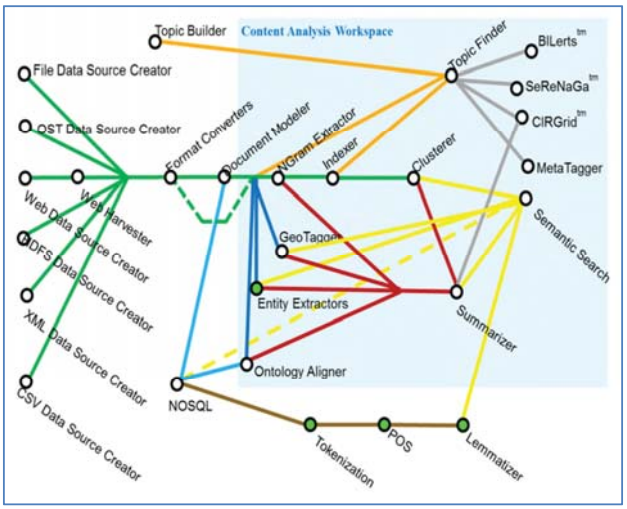


Figure 6. Content Analysis Tools

The workspace provides easy composition of text analytic solutions using modular building blocks. The text analytic capabilities (e.g. Phrase Extraction, Entity Extraction, Indexing, Clustering, and Ranking) can be easily dragged, dropped, and daisy chained to create a complete pipeline, that can then be attached to unstructured documents as well as structured data source.

Figure 7. Workspace to Compose Text Analytic Pipeline

6.1 Document Summarization

As an example, we discuss the Document Summarization since this is a portion of the capability depicted previously in Figure 4. Automated document summarization leverages many of the Content Analysis tools/APIs and in many ways represents the consummate application of these capabilities interoperating. Therefore the main components that contribute to document processing are discussed in the following sections. These steps work as a pipeline to take the corpus of documents and create a visual display that allows the analyst to quickly understand the important information from the raw data. The process, guided by the analyst's hypothesis, allows the pipeline to identify information that is relevant to the analyst's task providing a more targeted and useful summary.

The Content Analysis pipeline ingests documents of varying formats (.doc, .docx .pdf, .html, pptx, .xml,) and creates consistent representations of the material in xml format. The tool takes each document and identifies relevant metadata such as titles and authors as well as content in natural language text (sentences, headings, etc). In addition large documents (such as multi-page pdf files) are broken into single page document so that the overall length of a single document doesn't mask smaller focus areas found within. These document models are organized during the analysis and presented to the user as evidence supporting or refuting the analyst's hypothesis. When viewing the results the analyst can easily move between analysis of the data and the original data sources.

6.2 Entity Extraction

Named entity extraction is the task of identifying mentions of specific individuals by name in unstructured text. This provides the analyst with specific entities, such as people, organizations, dates, and locations, which were mentioned within the text. Entity identification is performed using methods that range from rule-based to supervised learning. ADIEN uses a hybrid approach that leverages rule-based system consistency with adaptability of supervised learning. Our approach uses an automatically updated table for the entity searches. The person and organization categorizes were generated with entity extraction. The supervised system utilized two available algorithms OpenNLP[8] and Mallet[9] to achieve the desired result. Training of the algorithms was performed with limited human intervention.

6.3 Topic Identification

Topic identification technology allows for groups of unknown documents to be clustered with their main topic points automatically identified, and subsequent presentation to the analyst to determine if this would be something worthy of further investigation. This task requires identifying statistically relevant topics within a document. This is accomplished through the use of an unsupervised n-gram modeling method. The benefit of this method is that the topics are not known a-priori but are identified based on the statistical distribution of the words within the corpus; and the algorithm runs in O(N) because only one pass through the corpus is required.

N-grams are defined as a contiguous sequence of *n* content words that occur in some proximity to each other in a corpus. For example, consider the phrase: *The Falcon 9 Space Launch Vehicle successfully completed.* Table 1 shows the 1-grams, 2-grams and 3-grams for this phrase.

1-grams	2-grams	3-grams
SpaceX	Falcon heavy	Space launch vehicle
Falcon	Falcon 9	Spacex successfully completed
mission	Space launch	
launch		

Table 1. N-Grams

6.4 Ontology Alignment

Ontology alignment is the task of aligning terms in unstructured text to concepts in an ontology (or semantic model). For example, in Figure 8 the terms *infrared signature* and *low heat signature* both align to the concept Infrared Signature in the ontology. The advantage of aligning terms to text is that search and navigation operations can orient a user to information about a concept without enumerating each possible term that could be used to describe that concept.

To align the documents, we use Securboration-developed lexicalization constructs in which the ontology classes have explicit links to linguistic object, word senses, surface forms, and part-of-speech information. These attributes are used to align the text within the documents to the ontology as seen in

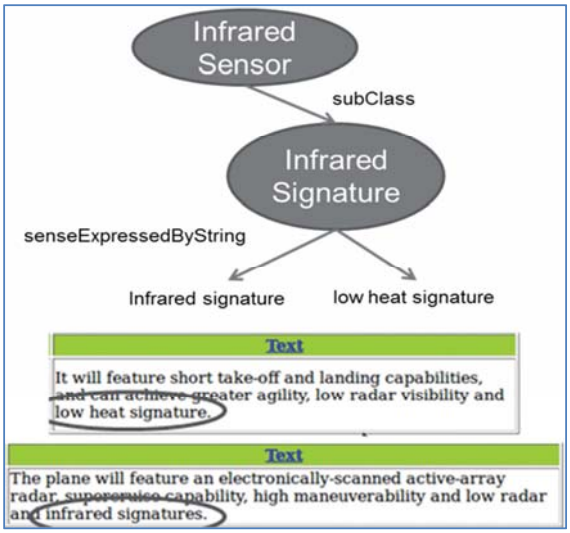


Figure 8. Ontology Alignment

Although not a required component Ontology alignment is one piece of the pipeline that elevates our approach over basic term search approaches.

6.5 GeoTagging

Geotagging is the task of automatically identifying locations in the text. This is different than entity extraction because geotagging tags a text with locations that may not specifically be mentioned within the document. For example, if the text is mentions the city Moscow, the geotag Russia may also be assigned to the document, if applicable, even though it was not explicitly mentioned. Geotagging allows the fusion of machine learning techniques with the declarative semantics of our geospatial ontology. Traditional machine learning uses a flat structure of features to learn classification rules. Gazetteers, on the other hand, are rich, multi-inheritance tree structures whose shape provides excellent disambiguation and classification evidence. To ignore this would be to waste the effort of constructing the gazetteer in the first place. In order to benefit from training data, we developed a weighted evidence consideration mechanism based on expressions likely to refer to geolocations and the familial relations of various geolocations within the gazetteer. Additionally, a threshold for all evidence is parameterizable. To train the geotagging model, we use a set of ground truth data and run Monte Carlo simulations of different evidence weighting schemes. The set of weights resulting in the best fit for the given ground truth is then used for similar corpora. The machine learning decisions can always be overridden by editorial changes to the gazetteer. This allows an important common sense override which many machine learning systems are resistant to.

6.6 Topic Ranking

Securboration's Topic Ranking organizes data sources, data assets and vocabulary by formally modeling natural language descriptions of topics and matching them against dynamically extracted features from source documents. Individual assets are collected from multiple data sources, along with descriptive metadata. Semi-structured and unstructured content is analyzed and features are extracted and indexed from the data assets. The index is optimized to perform weighted, structured queries pertaining to topic definitions. These queries are constructed based on a formal ontology of topics relevant to the topics. Once highly-relevant data assets are discovered via these queries, their original data sources are ranked based on how productive they were with respect to highly ranked assets. Further, features such as n-grams and named entities are ranked against the topics based on the frequency with which they are found in conjunction with formally modeled terms and concepts.

One measure of topic ranking is a tf-idf score. The *tf-idf* metric quantifies the degree of how relevant an n-gram is in a corpus based on the number of times it occurs in a specific document versus the number of times it occurs throughout all the documents in the document set. The basic idea of the tf-idf is that the more often a term occurs throughout the document set the less likely it can be used as a differentiator between documents. It is defined as:

$$tf\text{-}idf(n\text{-}gram,\ d,\ D) = tf(n\text{-}gram,d)\ x\ idf(n\text{-}gram,D),$$

where

$tf(n\text{-}gram,d)$ = frequency of the *n-gram* occurring in document *d*

$$idf(n\text{-}gram,\ D) = log\ \frac{|D|}{|\{d \epsilon D : n-gram\ \epsilon\ d\}|}$$

Within a body of work an overall score can be computed based on the tf-idf scores for the topics of interest, such as those from the hypothesis.

7 Conclusion

There is little doubt that analyst need new, innovative outside the box tools in their arsenal to combat the explosive generation, and access to, open source documents. Traditional techniques that focus on organizing knowledge the analyst has discovered are beneficial, but they neglect the information overload problem – i.e. a large amount of knowledge goes undiscovered. While still in the research stages, ADIEN has made significant progress innovating the analyst workspace to include capability such as task recognition and proactive data retrieval and information extraction; and leveraging state of the art services, such as analytical rigor. While ADIEN has proven important concepts, it is not yet at the level where it can be deployed into any analyst (domain) environment and automatically solve all their analysis needs (although our research indicated it would provide immediate benefit). Consequently, our near term areas of research for ADIEN involve expanding current capabilities described in this paper, as opposed to branching out and creating brand new capability (the challenges yet to be solved with personalized content analysis, task recognition, rigor analytics in the cloud is significant enough). The advent of cloud computing also facilitates rapid inclusion of leading edge analytic tools within ADIEN. For example, the Workflow capability can leverage image processing in the cloud (when it comes on line) to proactively extract all images of a certain type (e.g. engine) from a data corpus. Thus the goal for ADIEN is to continually evolve and incorporate capability as comes on line, and seamlessly leverage it to benefit analysts in all domains.

8 References

[1] http://www.infogineering.net/understanding-information-overload.htm

[2] Hueur, R. Psychology of Intelligence Analysis, Center for the Study of Intelligence, Central Intelligence Agency 1999.

[3] Patterson E., Zelik, D., Woods D. (2007) Rigor in Information Analysis. In S. M. Fiore & E. Salas (Chairs) Macrocognition Metrics: Meaningful Measures for Complex Processes. Panel Presentation at the 51st Annual Meeting of the Human Factors and Ergonomics Society. October, Baltimore MD.

[4] http://www.ai.sri.com/pal/PAL-software-downloads/PAL-zipfiles-and-docs/warp-doc/workflow_authoring.html

[5] Myers, K., Kolojejchick, J., Angiolillo, C., Cummings, T., Garvey, T., Gervasio, M., Haines, W., Jones, C., Knittel, J., Morley, D., Ommert, W. and Potter, S. (2011) "Learning by demonstration technology for military planning and decision making: A Deployment Story." Proc. of the 23rd Conf. on Innovative Applications of AI.

[6] Yorke-Smith, N., Saadati, S., Myers, K. and Morley, D. (2012) "The design of a proactive Personal agent for task management." International Journal on Artificial Intelligence Tools, 21(1): 90–119.

[7] Natarajan, S., Bui, H.H., Tadepalli, P., Kersting, K. and Wong, W. (2008) "Logical hierarchical hidden markov models for modeling user activities." Proc. of the 18th International Conference on Inductive Logic Programming.

[8] Baldridge, J., Morton, T., & Bierner, G. (2002). The opennlp maximum entropy package. Technical report, Technical report, SourceForge.

[9] McCallum, A. K. (2002). Mallet: A machine learning for language toolkit.

Discovering Expert Communities Online using *PSI4*

Steven Minton
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Neha Kansal
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Brian Amanatullah
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Kane See
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

Matthew Michelson
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

***Abstract*—The ability to find online communities with shared expertise or interests is a challenging and interesting problem. The problem involves collecting entity data, at scale, from a number of sources; linking data that refer to the same real-world entity across the sources; enriching the data to add the attributes that may define the community (such as interests); and finally analyzing the resulting data (for example looking at the social connectivity). Compounding these problems is the fact that in many cases the online community of interest is not well defined ahead of time, but instead only becomes crystalized through repeated cycles of data collection, refinement, expansion and analysis (what we call "interactive" sessions). In this paper we present PSI4, an end-to-end system that addresses these issues, and we walk through an interactive session to highlight its capabilities for community finding.**

I. Introduction

A challenging problem exists in finding online communities with shared expertise or interests, and their interrelatedness. For instance, a scientific funding agency may want to find out who the current experts are in a particular area, and how they related to one another in order to construct a review panel. Or a marketing professional at a company may wish to understand which journalists specialize in a particular area, and how they are connected socially, in order to strategize a Public Relations campaign.

However, there are a number of challenges in executing such a community-finding analysis. First, the data must be gathered, which involves collecting and normalizing large volumes of data from a number of different online data sources. Second, once the initial data is collected, it must be enriched to support advanced analysis and community finding. For instance, if the collected data involves user profiles (for instance, Twitter feeds) then it is useful to determine what that feed tells us about the person's interests and demographics. Or, we may want to geocode location information in the data to perform analysis such as geographical proximity. Further, social connections might be useful, so the data must be enriched with those features as well. Finally, the data can then be analyzed, for instance using social graphs (to understand communities), via queries and joins (to determine expertise across sources), or filtering the data (drilling down on experts in Los Angeles, for instance).

In practice, it can be very helpful if the community-finding process is *interactive*. Users searching for expert communities do not necessarily know exactly what they are looking for ahead of time, but instead it becomes clear during the search and analysis process. For this reason, a user should be able to collect, enrich, and analyze data, and then repeat any or all of that process using some refined subset (or expansion) of the data, until he or she has found the community of interest. That is, this is an interactive and online process, rather than a process where everything is known ahead of time, and data is sought to fit some preconceived requirements.

In this paper we describe the **PSI4**[1] system. PSI4 facilitates discovering these expert communities and addresses the major components described above: harvesting data, enriching data, and analyzing data. Further, all of the components are both interactive (supporting exploration and discovery) and extensible (so new sources can be harvested, new analytics can be plugged into the architecture, etc.). As we describe, PSI4 supports discovering expert communities of interest through interactive sessions.

The rest of this paper is organized as follows. In the next section we describe PSI4 overall, and detail a few of its components. After that, we describe an example interactive session to highlight PSI4's capabilities. Finally, we end the paper with our concluding remarks.

II. PSI4 Overview

At it's core, PSI4 is a large-scale data processing pipeline. It is able to consume huge volumes of Web data (including social media), process that data to enrich it (merging entities across data sources and inferring new features), and support search and filtering, as well as expanding the data set (such as bringing in social connections). The architectural components of PSI4 are shown in Figure 1.

As the figure shows, PSI4 supports the capabilities described above, and does so in a cyclical fashion that supports iterative

[1]PSI4 is an acronym for Probabilistic, Synthetic Inference for Intelligent Internet Investigation.

ISBN: 1-60132-438-3, CSREA Press ©

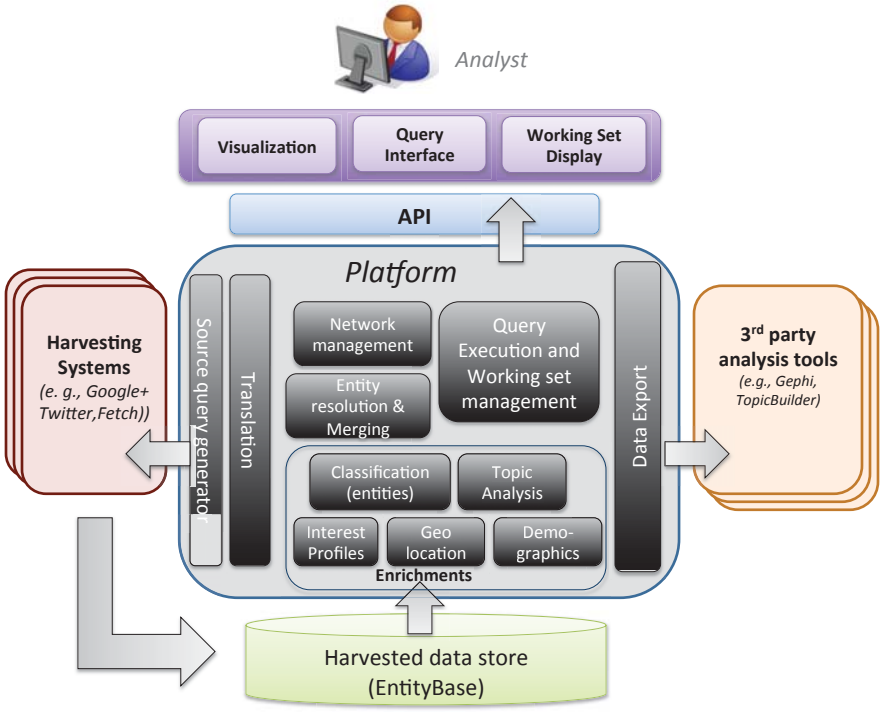


Fig. 1. PSI4 Architecture

sessions for discovery. The far left of this figure shows the Harvesting component. This is the main data collection mechanism for PSI4, which serves as the input for community discovery. We describe this component in more detail below (Section II-A), but to summarize its functionality, it serves as both a gathering and storage component for PSI4 and is able to ingest data from both social media and websites. Once data is collected by the Harvester, it is passed through our entity resolution system, EntityBase [1], [2]. EntityBase is able to link multiple references to the same real-world entity that may be described differently by different data sources. For instance, it may determine that James McNally on Twitter is JMcNally on Google+. Entity resolution is an important aspect for PSI4, since we intend to find communities of *individuals*, even though the various features about those individuals may come from different sources. (For instance, occupation information and interests may come from Twitter but geographic information is provided by Google+).

Once the entities are merged, they are expanded by the Enrichment component. The Enrichment component infers features about entities using an extensible set of inference mechanisms, which can employ inference rules, machine learning, as well as additional data sources. For instance, the Enrichment component calles inference mechanisms that infer people's interests and demographics based upon what they say about themselves on social media [3], [4]. In this manner, an entity can be enriched with demographic and interest information. Other enrichments may include topic analysis (for instance over a set of social media posts) or classification (to infer an occupation, for instance). We discuss the Enrichment component in more detail in Section II-B.

Finally, data is passed to the user for visualization, interactive queries to expand or drill down on the data (via the Query Interface), and filtering (via the Working Set Display). This all supports the notion of interactivity, as the user can cycle through PSI4's capabilities. In this way, a PSI4 user can interactively discover the expert community of interest. In the next subsections we delve deeper into two of the critical components, the Harvesting component and the Enrichment component.

A. Harvesting Component

The Harvesting component's main purpose is to collect data from various sources. It can collect and store data, at scale, from the major social media services (such as Twitter and Google+) using its own customized hooks into those services' APIs. In Figure 2 below, as user has initiated a social media search on Twitter account for JAIR (the Journal of Artificial Intelligence Research) and the system has collected the Tweets (which it then displays). We note that PSI4 is especially focused on social media collection and analysis. Since we want to not only discover who is in an expert community, but also their social interconnectivity we felt it was natural for PSI4 to harvest data from these social media sources.

The harvesting capabilities of PSI4 are quite rich. It provides a simple and unified query interface to social media APIs for harvesting entities (e.g., users from keyword search, followers of accounts, etc.) and data for those entities (e.g., social media

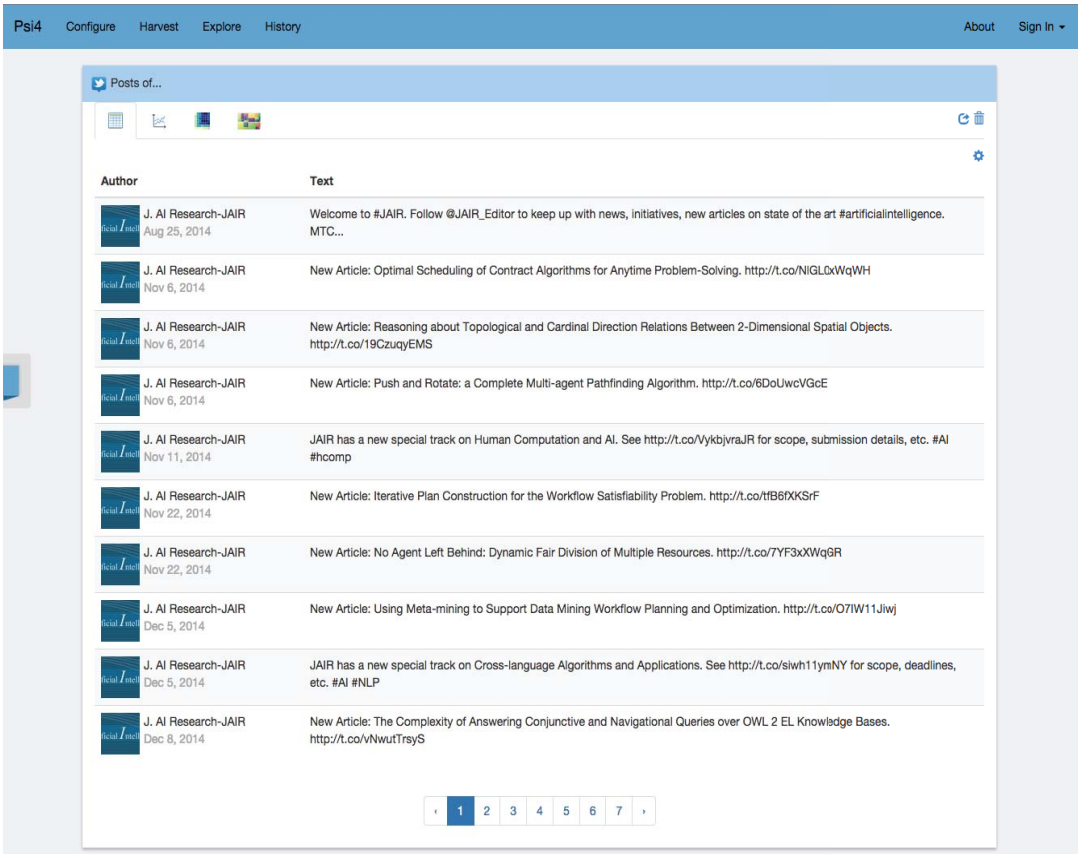


Fig. 2. Harvesting Social Media

posts or other information). In fact, we also realized that there is useful data outside of social media sources as well. Therefore, PSI4 supports the ability to ingest data from our own information extraction tools.[2] These tools are capable of turning semi-structured websites into structured records. For instance, imagine taking a directory website of an academic institution, such as the one shown in Figure 3 and turning this list of faculty members into stored data with the name, link to their faculty page, email and phone number which can then be included and linked to other entities within PSI4.

Finally, we note that not all data of interest is necessarily in English. Building upon our previous work in analyzing foreign language social media [5], our Harvesting Component can apply third-party tools for translating text, so that discovery can occur across multiple languages.

Name	E-mail	Work Phone
Andrade, Joe A	joe.andrade@usc.edu	(213) 821-0573
Arciniega, Mark	marcinie@usc.edu	(213) 821-0915
Baklarz, Mark Bucky S	baklarz@usc.edu	(213) 740-7383
Balles, Kristine	KBalles@dps.usc.edu	
Barker, Dennis	DBarker@dps.usc.edu	
Bell, Chris	cbell@dps.usc.edu	(213) 740-6000
Benzel, Terry	tbenzel@isi.edu	(310) 448-9438
Breckenridge, Stella M.	sbreckenridge@dps.usc.edu	(213) 740-5519
Carr, Rick	carrr@usc.edu	(213) 740-1278
Castellanos, William	wcastellanos@dps.usc.edu	(213) 740-6000

Fig. 3. A faculty page to turn into Harvested data

B. Enrichment Component

We now describe the Enrichment component in more detail. Importantly, we note that the Enrichment component deals with *entities*, rather than single records. The focus is to enrich entities with information that can be inferred from the raw data (and linked across sources, since we are dealing with entities). As an example, two records, one from Twitter and one from Google+, may be linked together to form a single entity. Then, each of these social media accounts will pass through our social media analyzer, along with other enrichments, to add features to the data. And since each social account belongs to the same entity, enrichments will attach at the entity level. This means, for instance, that we may infer demographics and interests based on Twitter, and occupation based on Google+ and the entity will contain both sets of enrichments.

[2]https://github.com/inferlink/extraction

We have a number of enrichment capabilities in PSI4 currently. First, by analyzing what people say about themselves on social media we are able to infer both demographics and interests. Figure 4 shows part of the PSI4 interface displaying an enriched record for the Twitter account of an expert on Web technologies. Based on his social media account, PSI4 infers demographics including that he is a "dad" and a "pet owner." His interests include "Web-related conferences" and "technology." This is useful information for community finding, since for instance, if we wanted to find the experts in Web technologies, the analysis of his social media profile would allow PSI4 to include him in the analysis.

dems	interests
PARENT	technology
PET_OWNER	communication
HUSBAND	society
DAD	automation
	digital technology
	computer networking conferences
	computing
	applied sciences
	web-related conferences

Fig. 4. An enriched account with interests and demographics

Another compelling enrichment supported by PSI4 is its topic analysis capability. Given either a series of social media messages from one account, or a group of accounts, PSI4 is able to cluster those messages topically, allowing a user to drill down into various topics and their sub-topics. For instance, it can analyze the Tweets of an individual to discover his or her main topics of discussion, but also do so for a whole community (once found). We will give an example of this analysis and its utility below. Finally, we note that PSI4 is able to geocode accounts as an enrichment, based on the social media profile.

III. Interactive Analysis: An Example

In the previous section we described two of PSI4's primary components, the Enrichment and Harvesting components. Here we put PSI4 to use, walking through an example of community finding. Our goal in this example is quite specific. We aim to find the expert community at the intersection of Artificial Intelligence and Web technologies, and uncover the topics they are currently talking about. Since we must start somewhere, the first thing we do is direct PSI4 to harvest from the Twitter account of the Journal for Artificial Intelligence Research (JAIR). The system will return a set of Tweets as shown previously in Figure 2. However, we realize what we really care about are the social media users JAIR is following, since those are more likely to be accounts of vetted computer scientists. So, once we tell PSI4 to harvest the followers, this data can be displayed as shown in Figure 5.

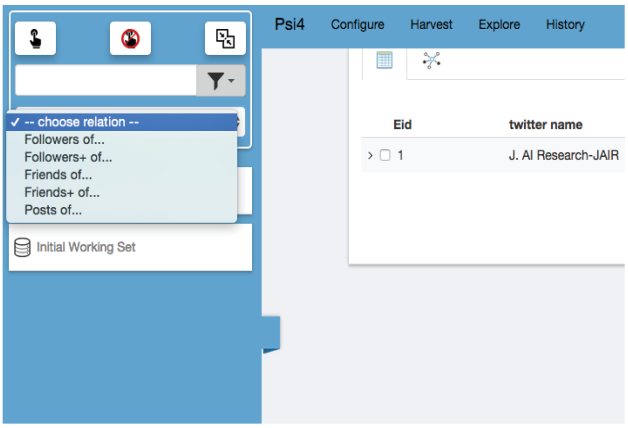


Fig. 5. PSI4 data expansion via social connections

Note that, as shown in the figure, PSI4 supports a configurable set of options for harvesting and displaying social media data. For instance, the set of capabilities includes:

- **Posts of**: the posts authored by an entity
- **Followers of**: the Twitter followers of an entity
- **Followers+ of**: the Twitter followers of an entity, as well as the entity itself
- **Friends of**: the accounts that the entity is following
- **Followers+ of**: the accounts that the entity is following, as well as the entity itself

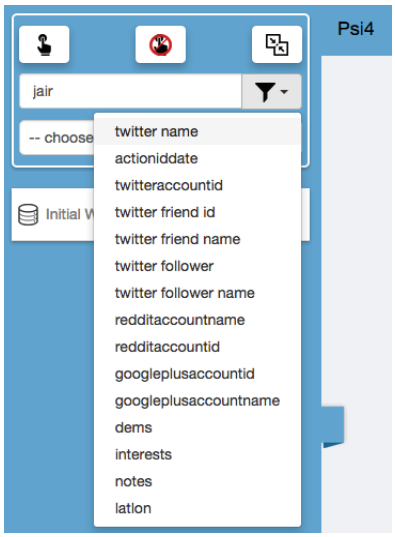


Fig. 6. Filtering a working set

Given the set of users that JAIR is following, we then enrich all of them. We find users such as the one shown in Figure 4. We note that a number of users have "Web-based conferences" and "Automation" as interests, so we use PSI4 to filter down on just users that share those interests, since that will define our expert community. Figure 6 shows part of PSI4's interface for filtering the entities of interest. As shown, we can filter by interests, demographics, LatLong, and any

other attributes defined over the entities, including user-defined attributes (such as "notes").

Given this community, we want to understand what they are talking about. So, again we have PSI4 harvest the posts for all of the users in our filtered set, and we run a topic analysis on their aggregated Tweets. This is show in Figure 7. As the figure shows, these users are talking about "Big Data," "Net Neutrality," and "Summer School." The Summer school topic sounds interesting, so we drill into that to see the details of the Tweets that compose that cluster.

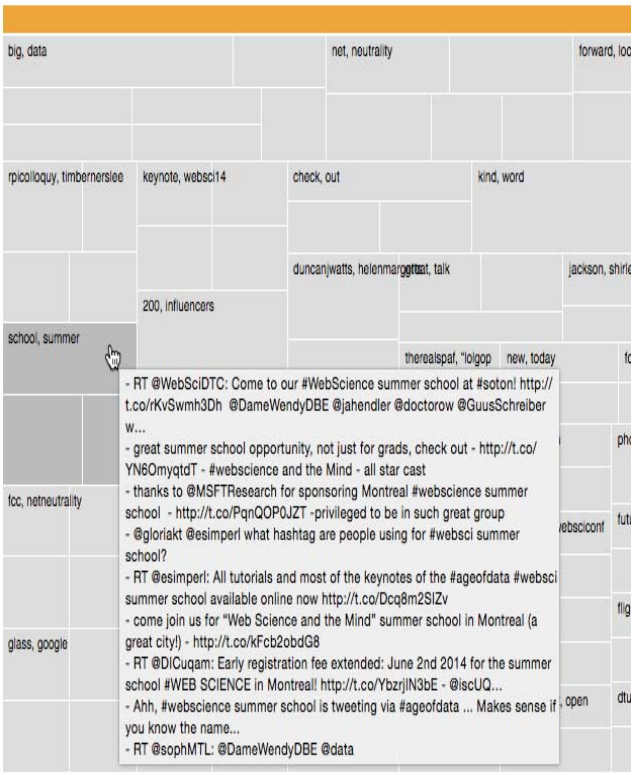


Fig. 7. Topic analysis for Tweets

Finally, we run an analysis of this group's posting behavior to see what the best time to reach out to them might be (to form our expert review panel). So, we direct PSI4 to produce a visualization of posting behavior across day of the week and time of the day. This is shown in Figure 8. The visualization shows more activity during the week (darker blue squares), and during working hours, and less during the weekend (yellow and white). Interestingly, for this group, activity mostly stops on Sunday morning, resuming towards the afternoon.

IV. Conclusion

In this paper we discussed PSI4, an end-to-end system for gathering, enriching and analyzing social media and online data in order to uncover expert communities of interest. The core capabilities of PSI4 involve its robust collect capabilities its deep enrichment of the collected data, and its ability to visualize and refine the data in various ways. Further, PSI4 supports interactive sessions, so users can discovery the communities they care about, even if they are not well defined ahead of time. While there are many future refinements for PSI4, we hope that a tool such as this will spur further research into internet-based analysis and improves the capabilities for many applications within retail, marketing and public relations.

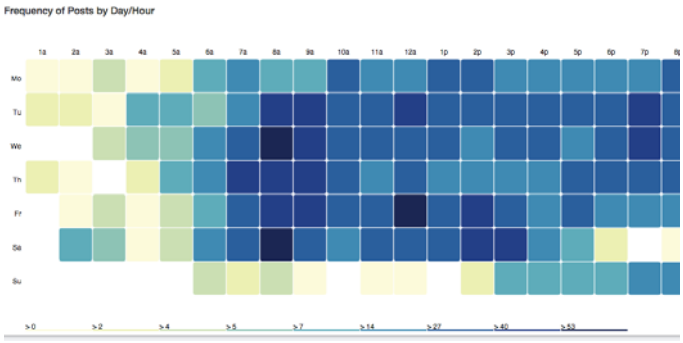


Fig. 8. A heat map of social media activity

V. Acknowledgment

This material is based on research sponsored by the Air Force Research Laboratory, under agreement number FA8750-13-C-0142. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the U.S. Government.

References

[1] C. A. Knoblock, J. L. Ambite, K. Ganesan, M. Muslea, S. Minton, G. Barish, E. Gamble, C. Nanjo, K. See, C. Shahabi, and C.-C. Chen, "Entitybases: Compiling, organizing and querying massive entity repositories," in *IC-AI*, 2007, pp. 15–21.

[2] S. N. Minton, S. A. Macskassy, P. LaMonica, KaneSee, C. A. Knoblock, G. Barish, M. Michelson, and R. Liuzzi, "Monitoring entities in an uncertain world: Entity resolution and referential integrity," in *Proceedings of the 25th National Conference on Artificial Intelligence (AAAI-11)*, San Francisco, CA, 2011.

[3] M. Michelson and S. A. Macskassy, "Discovering users' topics of interest on twitter: A first look," in *Proceedings of the Workshop on Analytics for Noisy, Unstructured Text Data (AND)*, Toronto, Canada, 2010.

[4] ——, "What blogs tell us about websites: A demographics study," in *In Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM)*, 2011.

[5] B. Amanatullah, G. Barish, S. Minton, K. See, and M. Michelson, "Analyzing foreign-language social-media reaction to televised speeches: Lessons learned," in *Proceedings on the International Conference on Artificial Intelligence (ICAI)*. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2015, p. 201.

Distributed Information Gathering, Exploration and Sense-making Toolkit (DIGEST)

P. Benjamin, K. Madanagopal, M. Erraguntla, and D. Corlette
[1]Knowledge Based System, Inc., College Station, TX, USA

Summary – *This paper describes a semantic toolkit for extracting information from multiple text and social media sources, and fusing the information so as to facilitate enhanced situational awareness and decision making. This innovative new capability assists information analysts to rapidly reason social media data to support enhanced situational awareness while significantly reducing the time and effort needed to convert data to action-enabling information. The semantic toolkit (DIGEST) provides the following capabilities: (1) rapid-sensing of multi-source text data to extract actionable knowledge – including emerging themes, trends and patterns in sentiments, social-influence and information flow structures; (2) exploratory data analytics and knowledge discovery methods to support collaborative information analysis – including social network analysis and event extraction and analysis; and (3) using the stored results from the exploratory information analyses to generate a variety of knowledge products.*

Keywords: Data Integration, Knowledge Extraction, Ontology, Big Data

1 Motivations

Information analysts are drowning in a flood of information. Modern sensors and reporting systems produce unprecedented amounts of data; modern processing capability is transforming, churning, and translating that data at an ever-increasing rate. Yesterday's analysts had to be experts at interpreting the sparse data they could obtain; tomorrow's analyst will need different expertise: discovering, corroborating, and interpreting actionable information from the rising flood of data. Evolving and emerging information technology provides the opportunity to equip tomorrow's analyst for this challenge. Improvements in communications capabilities make collaboration in near real time and at continental distances a reality today. Semantic information processing provides the analyst with access to a deeper understanding of data—bringing data closer to information even before the analyst sees it. And modern visualization techniques allow for rapid assessment of large amounts of data and enhance the analyst's ability to communicate observations, hypotheses, and findings. A key challenge is to provide significant reductions in 'data-to-decision' time through the use of advanced methods and tools.

1.1 Key Challenges

A recent estimate reports that more than two billion people currently use the Internet and that number is expected to rise in the future. There are an estimated 750 million Facebook users and an estimated 100 million users of the Twitter® social media web site. Forty percent of Twitter® site users read other people's tweets and do not tweet themselves; around 230 million tweets are posted daily.

This widespread adoption of social media provides a means for communication previously unavailable to large segments of the world's population. Disaffected populations that previously lacked the means to shape political debate or effect social change have now been given a significant voice. The use of social media may already have played a critical role in several recent rebellions including the Moldavian revolution, the student uprising in Iran, and the Arab Spring. There have been reports that the State Department asked Twitter® to delay a scheduled maintenance event so that Twitter® feeds coming out of Iran would not be disrupted during the Iranian student uprising of 2009.

A research group in Dubai claimed that they were able to detect indications of the Arab Spring uprising before it made headlines by analyzing 10 million social media conversations in the days before the event. There is a strong belief that social media can be an important source of political and cultural intelligence. Yet, there is no consensus that correct insights, much less actionable intelligence, are being drawn from the flood of social media content. The analysis of social media is very much an open research area. The potential value of social media as intelligence grows every day. Analysts need effective capabilities with which to mine social media for indicators of upcoming events, and to plan courses of actions to influence the social networks reflected in the social media. Policy makers and social-media experts identified the following social media challenges during a recent meeting at the U.S. Institute of Peace:

- Analysts are faced with a deluge of data generated by social media sources. Solutions must scale to the size of the social media ocean.
- In addition to the size of the data, the content of the data is complex. The identities of the persons (entities) involved in a community are often hidden or anonymous, sometimes deliberately so. The complexity of the data

must be mitigated by effective entity resolution or disambiguation.

- Social media are "reshaping human language" through the use of tags, idioms, symbols, acronyms, etc. This places new burdens on existing natural language processing techniques.

Dynamic situational awareness and, consequently, effective course of action planning can only be achieved through adaptive information processing techniques that allow for coherent sensing and monitoring of the "pulse" of the affected community or group. The pulse here encompasses the sum total of ground realities and facts as they occur, both physical and tangible measurements (for example, satellite images or traffic camera feeds in urban crisis situation), as well as more factual and softer sensed measurements such as participating entities, the information-flow and influence structures. There is ample evidence to suggest that soft sensors such as social media offer the greatest promise, at the same time posing the greatest challenge to providing situational awareness for a variety of knowledge-centric application situations.

One of the challenges that face information analysts is the disparate sources of data ranging from news articles, reports, RSS feeds, and the more recent source of data: social media sites such as Twitter®, Facebook, blogs, etc. Recent experiences suggest that the leverage of information is the critical technological chasm for accurate SA – and this is true for the full gamut of the situation-types ranging from socio-political upheavals to disease outbreaks and natural disasters. The massive quantity and diverse types of information (especially in social media such as Twitter®, Facebook, etc.) that overlay and emanate as threat situations unfold make it infeasible for humans to effectively filter it, much less organize, make sense of, and act on it. For example, hundreds of millions of tweets are generated on a daily basis making real time monitoring very challenging. In summary, the task of automatically generating useful, reliable and actionable information from multi-source, multimodal data is a daunting information comprehension and integration challenge.

2 Overview of Solution Concept

2.1 Distributed Information Gathering, Exploration and Sense-making Toolkit (DIGEST)

DIGEST uses a variety of different text analytics methods to rapidly process multi-source text and social media data (Twitter®, YouTube, blogs, media news sources, etc.) in order to generate accurate and reliable information products for enhanced situational awareness and decision support. DIGEST is a significant force-multiplier, enabling analysts to work more efficiently through the use of pre-defined information product templates and an ontology-based semantic data tagging process. DIGEST's plug-and-play architecture means that the application may be tailored easily and fielded within a wide variety of customer application contexts. In order to address the scalability, replication, fault-tolerance and parallel processing, we have carefully designed each of the components of DIGEST using Big Data technological solutions. The DIGEST conceptual architecture is shown in Figure 1. The DIGEST components are grouped under four main layers that are summarized in the following.

- The **data collection layer** consists of the components that are required to ingest the data from external source to the DIGEST pipeline. The collection components are designed to support downloading of data from heterogeneous sources/formats like web, stream, enterprise data, etc. The source data is also stored as-is to support a source provenance trail.
- The **data enrichment layer** consists of components that help to enhance, refine or otherwise improve raw data. Another way that the data enrichment layer would work is extrapolating data; i.e., produce more information from the given raw data set.
- The **data processing layer** consists of components that assist in producing meaningful information from the raw data. For example, sentiment extraction, sense identification, and topic extraction were some of the data processing activities. Some of the data processing layers use the large scale data processing

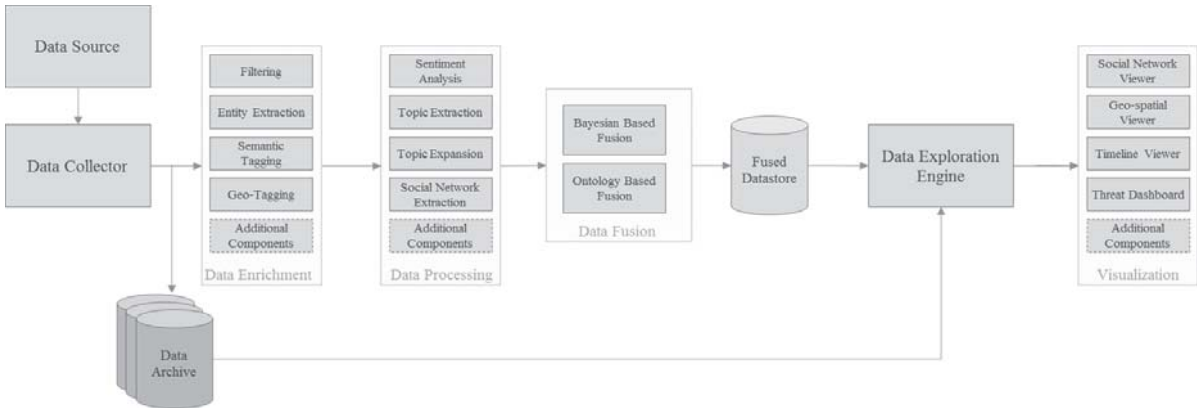


Figure 1. DIGEST Conceptual Architecture

capabilities provided by the big data infrastructures such as Hadoop, Oozie, etc.

- The **data fusion layer** consists of components that integrate information derived from the various sources so that the different feeds complement and reinforce each other. A 'reference ontology' is used to support the fusion of information derived from text and other types of information-seeking sources.

In order to provide efficient mechanisms to explore the information, the fused information is indexed in a key-value document store which provides near real-time indexing and querying. We have also built custom user interfaces to enable the information analyst to view the search results in an efficient manner.

3 DIGEST Analytics Overview

DIGEST "Community Pulse" analytics provide analysts the capability to detect topic and sentiment trends over time within a region or community. More specifically, "Community Pulse" analytics allow analysts to analyze topic frequency for a given data source, record overall sentiment average, and detect sentiment shifts over time. The information provided by this capability is particularly valuable when planning an operation or developing organizational Courses of Action (COA) options. Figure 2 shows active visualization graphs from the "Community Pulse" sub-menu in the DIGEST main window (clockwise from top-left): (1) "Topic Over Time," (2) "Sentiment Over Time," (3) "Sentiment Summary," and (4) "Sentiment."

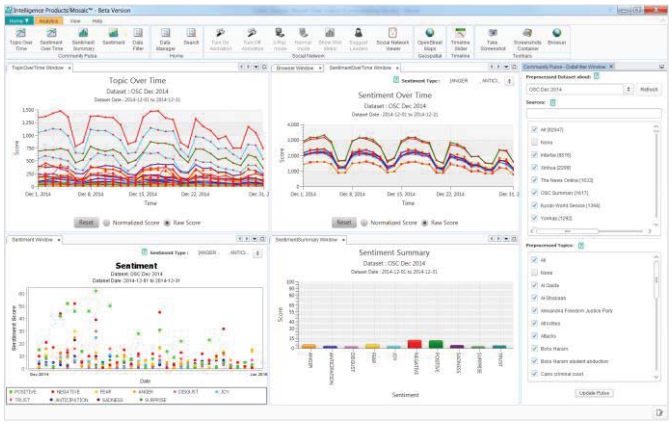


Figure 2. Example DIGEST Community Pulse Charts

3.1 Semantic Tagging

Semantic tagging refers to the activity of 'labeling' data with 'tags' that provide 'meaning' in the context of an application about the information contained in the data. Two types of data are tagged (within the scope of this project): text and images. Text data tagging is done using an ontology-based natural language processing (NLP) capability. This includes the capability to generate Resource Description Format (RDF) semantic labels from multi-format raw text data. The DIGEST image data tagging (a) leverages previously image-processed data and (b) exploits the information provided within the image 'metadata' tags. The image processing includes object detection, tracking, and background/foreground segmentation. A unique aspect of our semantic tagging approach is to use an ontology to increase the semantic quality (depth of meaning represented) of the resulting text and image tags. To illustrate, the semantic tagging activity for an example sentence is shown pictorially in Figure 3.

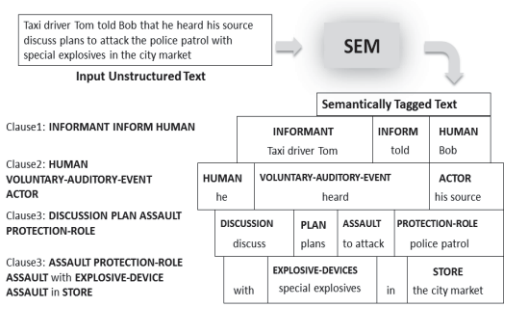


Figure 3. Semantic Tagging Induced Semantic Labels to the Clauses in Every Sentence

The 'heavy lifting' that is necessary to convert the raw text into meaningful information is provided by the KBSI NLP 'pipeline,' shown in Figure 4. The process shown in the figure uses a combination of NLP techniques and is capable of scaling to handle large text data collections in multiple formats.

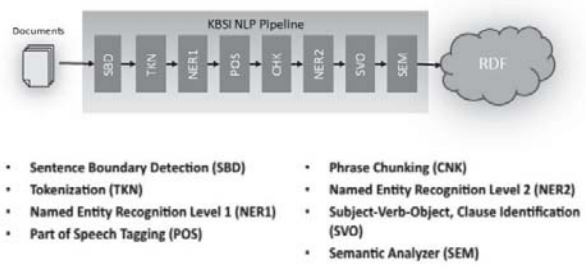


Figure 4. Dataflow Diagram of the KBSI NLP Pipeline

A unique innovation is the ability to use an ontology to increase the richness of the semantic tags. Each block in the pipeline is labeled according to the set of tags that are added to the input text after the input text has passed through the block.

3.2 Topic Modeling and Monitoring

Topic modeling is a text mining technique that analyzes large volumes of data for patterns of words that collectively represent a topic. Characteristics (semantically aligned words) of the topics evolve over time and it's required to periodically model the topic and monitor for any abnormal events. The information analyst have to go through a non-linear workflow to finally arrive at the topic model. Figure 5 shows the set of analysis steps that are involved in the non-linear workflow followed by the analyst to solve the analysis problem. Minor tweaks could be made to optimize the workflow for the chosen

dataset. The set of steps resulting from exploration represents a deployable autonomous analytic. Deep knowledge of the mechanism of each data source, analytic, or visualization was not required by the analyst. The set of analysis steps taken in the non-linear workflow could be saved, shared, and optimized.

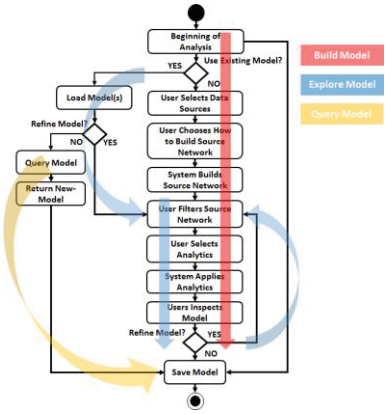


Figure 5. Non-linear Workflow of Topic Modeling

Figure 5 provides a data flow diagram outlining T-SAT's non-linear workflow capability. The Figure 6 below demonstrates the direct application of natural language processing algorithms to the visualizations. Begin with Step 1 in the upper left of the figure where the user is provided a top-level topic model derived from the processing of a collection of news articles spanning the period of one year. The topic-model was generated using a combination of an LDA Topic Modeling algorithm [4] and KBSI's Natural Language Processing (KNLP™) Pipeline.

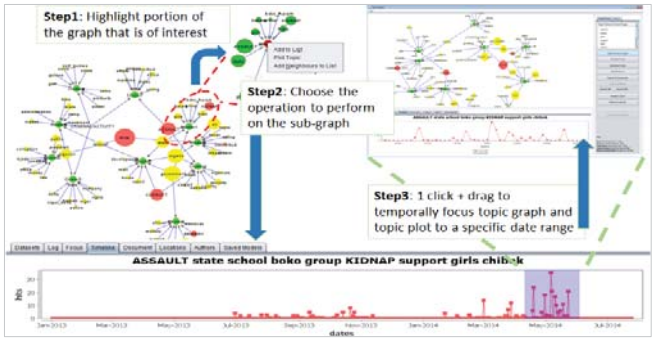


Figure 6. DIGEST Topic Modeling User Interface

Green nodes represent the topic centroids, yellow nodes represent plain lexical items (e.g., words or phrases), and red nodes represent event related concepts in our ontology such as ASSAULT or KIDNAP. Such concepts are generated using the semantic tagging capability of the KNLP™.

At Step 1 the user selects a portion of the graph that is of interest. Selection of the sub graph triggers the appearance of a menu displaying the relevant NLP algorithms for the given analysis context. In this example the portion of the topic model highlighted in red deals with the kidnapping of school girls by Boko Haram in Chibok. In Step 2 the user elects to plot data related to the selected sub portion of the graph. The timeline updates to reflect the temporal frequency of the components of the selected portion of the topic model. In Step 3 the user elects to apply the LDA Topic Modeling algorithm to a sub selection of the plotted data. The topic models that are extracted are then applied to the new data and monitored for abnormalities. The screenshot given below shows the topic over time interface that is implemented in the web application. It has additional charts like topic count, document frequency by week and document frequency by days. All the charts are coordinated charts; i.e., all the charts can be filtered by clicking on any of the chart displayed.

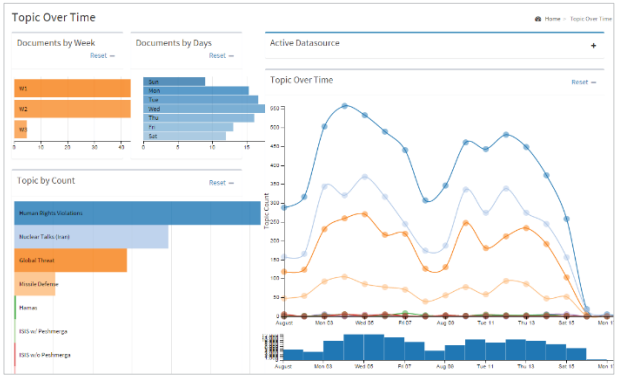


Figure 7. Topic Over Time Interface of DIGEST Web Application

Additionally, the analyst can click on the bubbles in the line chart and see the source documented for the selected date and topic.

3.3 Dynamic Topic Over Time

One of the drawbacks of our topic over time approach is its static nature; i.e., the topics are prepopulated and the documents are tagged in the preprocessing stage. It doesn't allow the analyst to tweak any topic and see the results. With the primary intent to make the detection document for a topic in real time we implemented a dynamic topic over time. The dynamic topic over time interface shows pre-defined topics and also allows an analyst to create a new topic. The topic definition page is split into four semantic containers they are people, location, organization and event types. When the update chart is clicked, a smart search query is constructed and executed against the selected data source.

3.4 Sentiment Tagging

For every topic evidence that is tagged through the topic over time tagger, sentiment tagging is performed to measure the nuanced aspects of sentiment. In our sentiment mining approach, the initial choice of the affected dimensions were joy, trust, sadness, anger, disgust, anticipation, fear, and surprise, along with overall positive and negative sentiments. Our approach creates an amalgamated lexicon that fuses information from various open source sentiment lexicons and manually curated lexicons (with the intent to include SentiWordNet after further validation, as well as a customized lexicon that can store domain specific tokens).

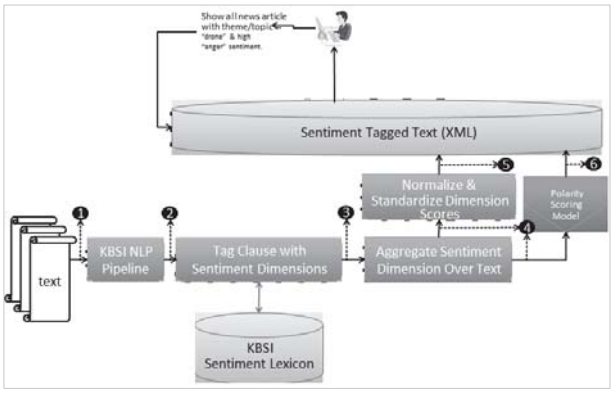


Figure 8. Sentiment Tagging Pipeline

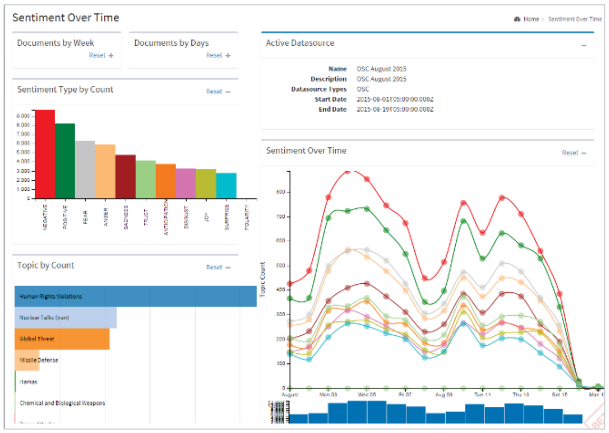


Figure 9. Example Sentiment Over Time Display

The text that is to be sentiment scored is parsed using the KBSI KNLP™ Pipeline tool where the parse tokenizes the text and tags it with common syntactic elements – such as part of speech tags, clauses, concepts, etc. The screenshot given above shows the sentiment over time interface that is implemented in the web application. It has additional charts like topic count, sentiment type count, document frequency by week and document frequency by days. All the charts are coordinated charts; i.e., the charts can be filtered by clicking on any of the chart displayed.

3.5 Analytics Support for Relation Extraction and Analysis

The DIGEST text analytics components are used to automatically extract different types of relationships including social networks and event networks. This 'deep knowledge extraction' capability enables DIGEST to answer questions such as what are the objects or who are the persons of interest in the emerging situation, who are the key leaders and what are their social networks, where are the events taking place, how are the events being carried out, why are there operations of interest, etc.? Answering these questions involves augmenting human subject matter expertise with fused threat assessments derived from multiple sources (including social media) while taking into consideration subject matter expert knowledge. Text analytics-based algorithms are used to extract social networks and activity profiles. The analyst can define a list of initial sources and metadata filters using the DIGEST frontend application to explore the extracted social networks. An example DIGEST social network analysis display is shown in Figure 10.

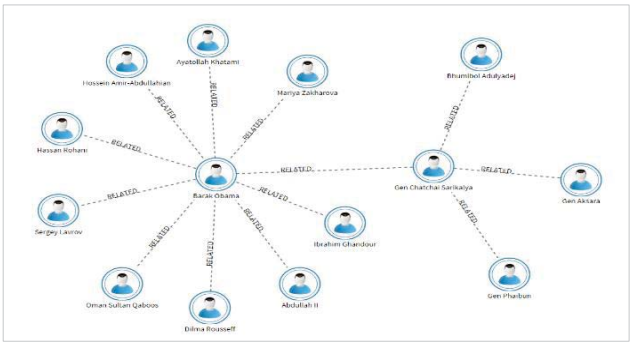


Figure 10. Example DIGEST Link Exploration and Analysis Display

The DIGEST system supports a fluid and creative environment for the analyst where the burden of data source, text analytics-based algorithms, and visualization details are removed giving the analyst more time and energy to expend on data exploratory efforts.

4 Putting it All Together: DIGEST Automated Support for Product Generation

This section describes the DIGEST information product generation capability. The information product generation is often time- and expert-intensive work, where analysts sift and cull information from multiple data sources and compile and integrate the information into different types of reports.

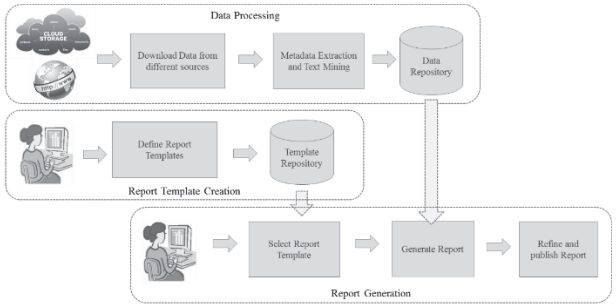


Figure 11. DIGEST Automated Report Generation CONOPS

In todays' 'big data' landscape, the number of data sources and the quantity of information we get from each data source pose a significant challenge. The increasing velocity of data updates further compound this 'information overload' situation. The DIGEST capability that addresses this problem and helps information analysts by providing automation support for the information product generation process. The DIGEST tool enables teams of collaborating analysts to access multiple data sources and rapidly generate information products using an interactive and intuitive user interface.

This concept of operation of automated report generation in the DIGEST platform is now outlined. There are three parallel tasks that happen in the DIGEST tool with different stake holders, as shown in Figure 11.

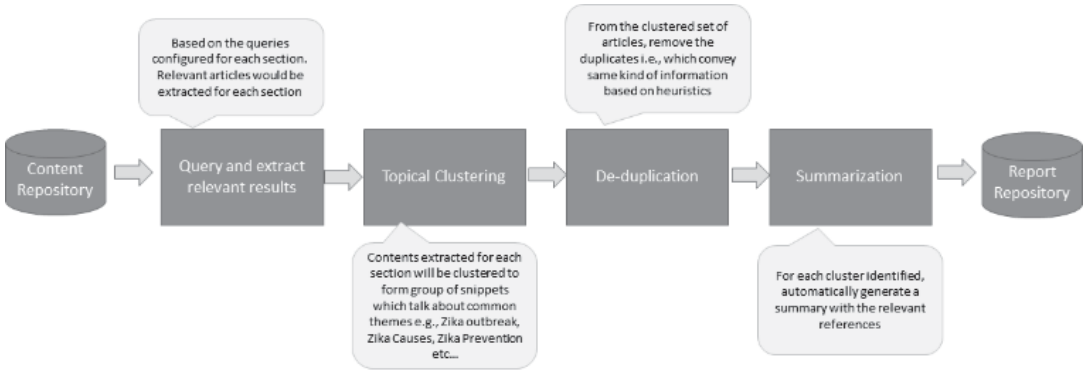


Figure 12. Content Extraction Process

1) **Data collection and processing**: The DIGEST tool is a role-based access controlled web application. An analyst can configure a set of web sources (RSS) to be downloaded into their profile which will be monitored for data download. The downloaded information will be processed and stored for the data source owner to perform various analysis such as topic exploration, sentiment exploration, model-based analysis, geospatial analysis and finally generate products/reports. In this example the data collection was configured to extract disease outbreak information from news articles and disease outbreak bulletins from health organizations such as CDC and WHO.

2) **Report template creation**: The DIGEST tool's product generation is based on the fact that a report consists of various sections which summarize different details with respect to the common theme (report objective).

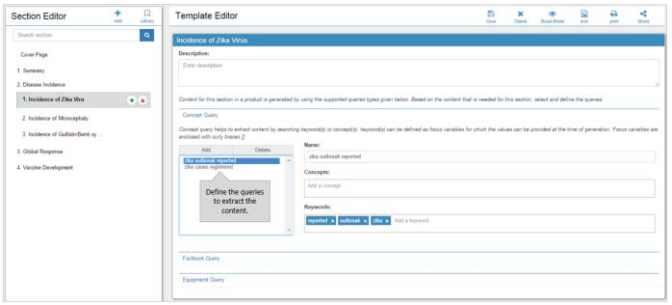


Figure 13. Template Editing in DIGEST

The data for each section comes from different sources. An analyst creates a report template by first adding all the section headers that will be available in the final report and then configures each section header with a set of queries to extract the data for each section. For example: for a section about population of a region/country, the analyst can configure it to extract it from CIA World Factbook and for a section like recent news articles about a disease outbreak configure a concept query. The report template can be saved and shared. Figure 13 shows the template editing page of how the concept query is added to a section.

3) **Report generation**: Once a template is created, reports can be automatically generated from the templates. The templates page of the DIGEST tool shows all the existing templates for the currently logged-in user. Every template has "Edit" and "Generate" options. If the user wants to edit the template before generating the report, they can click on "Edit." To generate the report, the user clicks on "Generate." Reports can be generated immediately or scheduled for automatic generation at a single or multiple future dates.

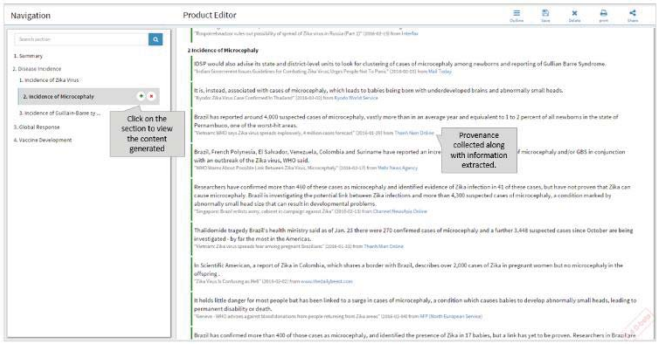


Figure 14. DIGEST Report Editing

The report editing page (Figure 14) of the DIGEST application contains a section navigator on the left to navigate to different sections of the report with ease. The extracted content for each section will show as individual snippets with the source information below them. The user can edit/remove the extracted content and also add his own insights into any of the sections. Finally, the completed report can be published directly from the product page of DIGEST. The DIGEST technology also contains various other analytics that help to explore the collected data.

The DIGEST tool also supports the creation of word clouds from search results. The word cloud uncovers the common topics that are discussed in the search results. For example, the geospatial viewer shown in Figure 15 displays the search results on the geospatial map with automatically

extracted facets on the left. The user can click on those facets to filter the results in detail.

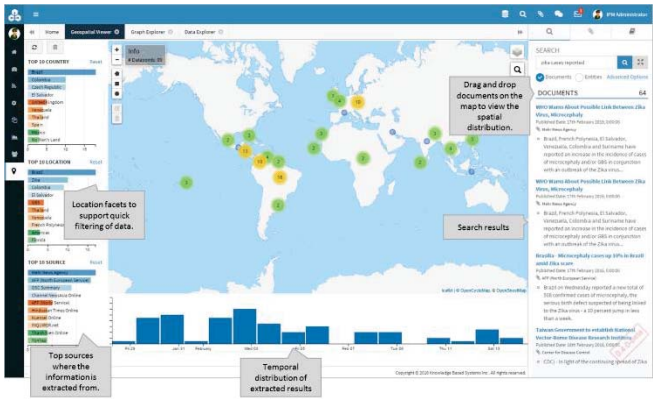


Figure 15. DIGEST Geospatial Viewer

For example, the word cloud for "Zika" showed some interesting topics like: (1) Aeds Aegypti mosquito is suspected to be the vector for Zika outbreak in Brazil, (2) Pregnant women and newborns are affected, and (3) Microcephally and Gullian-Barre syndrome are related to Zika.

5 Summary and Opportunities for Further R&D

This paper described a framework for extracting information from multiple text and social media sources and fusing the information so as to facilitate enhanced situational awareness and decision making. The problems that motivated the research were described. The Distributed Information Gathering, Exploration and Sense-making Toolkit (DIGEST) solution architecture was described. The semantic information analysis capabilities of DIGEST including topic modeling, sentiment analysis, and social network analysis capabilities were outlined. Finally, the ability to compile and automatically publish the semantic data analysis and fusion results in the form of information products and reports was outlined. Promising areas for future R&D include (1) automated event extraction, (2) information fusion from multi-source and multi-modal data, and (3) automation support for collaborative information analysis and knowledge discovery.

6 References

[1] Benjamin, Perakath, et al. "A Framework for Multi-source Semantic Information Extraction & Fusion for Collaborative Threat Assessment (SIFT)." *Proceedings on the International Conference on Artificial Intelligence (ICAI)*. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013.

[2] Kerne, Andruid, and Steven M. Smith. "The information discovery framework." *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*. ACM, 2004.

[3] Fayyad, Usama M., Gregory Piatetsky-Shapiro, and Padhraic Smyth. "*Knowledge Discovery and Data Mining: Towards a Unifying Framework.*"*KDD*. Vol. 96. 1996.

[4] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." *the Journal of Machine Learning Research* 3 (2003): 993-1022.

[5] Daniel M. Russell, Mark J. Stefii, Peter Pirolli, Stuart K. Card, "*The Cost Structure of Sensemaking*", 24-29 April 1993 INTERCHI'93.

[6] Wright W., Schroh D., Proulx P., Skaburskis A., Cort B., "*The sandbox for analysis: concepts and evaluation. In Proc.*", ACM CHI 2006. 2006.

A Privacy Approach for Crowd-Source Analytics Based on Internet of Things Sensor Data

Dean C. Mumme, Robert M. McGraw, Richard A. MacDonald
RAM Laboratories, Inc.
San Diego, CA

Abstract—This paper discusses an effort to provide mobile applications and crowd-based analytics that protect user privacy information. The paper discusses an approach for abstracting personally identifiable information and location information both at the device and in cloud-based analytics. The paper provides a walkthrough of several clustering analytics examples. As an example, crowd-sourced alerts related to air pressure measurements for weather alerts are discussed.

Keywords-Big Data, privacy, Privacy by Design, analytics, clustering, geo-spatial clustering

I. INTRODUCTION

As the ubiquitous nature of networked devices drive the size of the Internet of Things (IoT) to larger and larger scales, the opportunity for applications that use sensor data produced by those devices grows exponentially. A challenge in using this data, however, is that there are significant privacy concerns associated with individual sensors and/or sensor readings involving locations and individuals. Analytical approaches are needed to leverage sensors found on the IoT while also suppressing data that facilitates assimilation of actionable information pertaining to individuals. This paper details a proof-of-concept for an analytical process that utilizes reporting and clustering analytics in a manner that suppresses user-specific information while generating actionable information potentially useful for characterization air pressure-related weather alerts as shown in Figure 1.

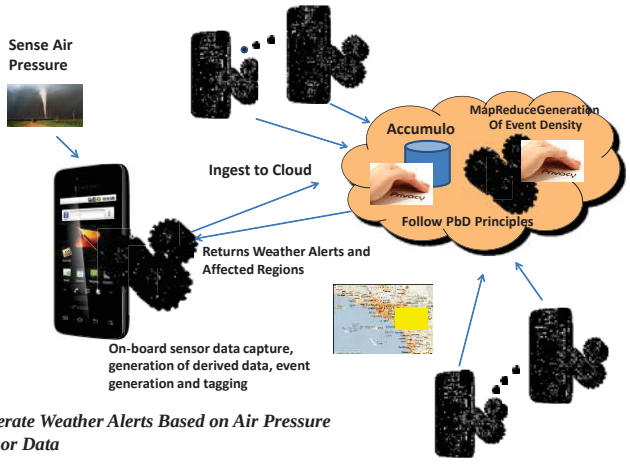


Figure 1: Mobile Devices Sense Environment Data and Deliver Event Reports to Back-end Analytics in Cloud to Generate Alerts. Personal Information, such as Identity is Suppressed

II. PROBLEM BEING SOLVED

Owner installed mobile device applications leverage data extracted by mobile device sensors residing on the Internet of Things (IoTs). Such applications typically provide high-level, actionable information about environment events having natural or artificial cause based on crowd-source device sensor data. These applications must be accurate in their assessments and provide them in near-real-time while also preserving the privacy of device owners. A key facet of this research was designing these reports and back-end analytics that use them with privacy in mind. To accomplish this, the design of our analytics followed Privacy by Design (PbD) principles.

II.1 Privacy by Design

In examining the privacy aspect for mobile communications, including the privacy aspect of applications operating over mobile communication devices, we turned to the seven foundational principles of PbD [1]:

1. Proactive not Reactive; Preventative not Remedial,
2. Privacy as the Default Setting,
3. Privacy Embedded into Design,
4. Full Functionality – Positive-Sum, not Zero-Sum,
5. End-to-End Security – Full Lifecycle Protection,
6. Visibility and Transparency, and
7. Respect for User Privacy.

Within this context, researchers in the PbD spectrum put out a Roadmap for PbD in Mobile Communications [1]. These researchers outlined the PbD concerns for the device manufacturer, the operating system/platform developer, the network providers, application developers and users. As the application developer on this project, we were concerned with the use-cases outlined in the roadmap that pertain to the application development and data processors. These use cases are:

1. Abide by the protections of the Global Privacy Standard,
2. Employ notice and informed consent,
3. Utilize and document appropriate security practices,
4. User privacy-protective default settings,
5. Ensure end-to-end protection of data, and
6. Design applications with privacy in mind.

Of these, cases 5 and 6, ensuring end-to-end protection of data and designing applications with privacy in mind had the most impact on the design and implementation of our crowd-source analytics.

II.2 Privacy Challenges for End-to-end Protection of Data

In order to provide privacy and end-to-end protection of data, the solution must ensure data integrity, authentication, and non-repudiation of data. A common technique in providing this end-to-end protection is to use Public Key Infrastructure (PKI) that utilizes keys generated from device identity for signing and encryption functions as shown in Figure 2. While the use of PKI provides an effective tool for protecting data in motion, the sender's authentication process may result in that user/device losing its privacy in the event that an eavesdropper has access to the authenticating party due to the fact that the user's Identity information is typically passed along with the message. Techniques that separate this Identity information from the event reporting are a desired solution when ensuring user privacy.

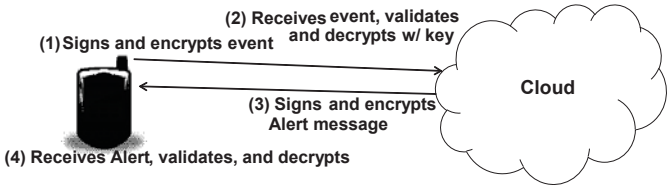


Figure 2: Typical Messaging Process for IoT Devices and Their Back-end Analytics

II.3 Privacy Challenges for Back-End Analytics Design

In addition to ensuring end-to-end protection of data through separate authentication and data processing processes, our apps and analytics were designed with respect to privacy principles. When examining our initial phases of event generation and event reporting, it becomes apparent that a potential shortfall may exist in the process of generating or extracting the sensor data. For instance, a common technique that can be used to extract information via applications on mobile phones is to have those devices constantly poll their sensors and report their data and position at a constant time interval. A shortfall of providing constant or consistent position updates at a given point in time is that tracks can be assembled for the mobile device in the vehicle in question and used to subvert the user's privacy by identifying all the locations to which he/she traveled.

III. Design for End-to-End Protection of Data

To address privacy protection in the messaging process, the project examined several techniques to obfuscate or suppress privacy information in the event reporting process. One solution to alleviate this problem is the use of random identifiers. Using random identifiers, the vehicle tracks can be identified but the randomization process keeps that information from being quickly correlated with a specific user.

Despite the use of randomized identifiers, data collected by application can still be used to track the users. For instance, consider a traffic monitoring application. In this case, for any track, the data can be traced until that specific track was present or initialized at a known home or work location that could be correlated with a specific user. To address privacy subversion in this manner, the analytics process must be reconfigured from the data capture through the analytics processing. For instance, instead of relying on a poll-based solution that uses user/vehicle identifiers, sensor location data and timestamps, the concept of Virtual Trip Lines (VTLs) was introduced [2][3]. The use of VTLs enabled the traffic monitoring application to abstract location by changing their reporting structure from the polling based system to an event based system (with the event being the time when the vehicle/mobile device reached a specific GPS coordinate). Additional enhancements of abstracting the timestamps and aggregating VTLs across a number of tracks were further used to ensure the privacy of specific devices' location and time.

A second more successful approach entailed implementation of a two-step authentication and messaging process as described in Hoh et al [4]. This technique de-couples the data processing and authentication steps being performed by the app. In this event, a solution utilizes an authentication based proxy server for connecting to the app, and a second analytics server for handling the messaging (or in cloud terms, the ingest and processing).

On this project, a third approach was explored that enhances scalability while also preserving privacy. It entails creation of "summary nodes" immediately upon receipt of device sensor data as shown in Figure 3. Each summary node represents a geographically located cluster of a statistically significant number of devices (say 30). For each sensor type, the sensor readings are summarized as a mean and standard deviation for the cluster. This is also done for derived data, particularly "rate-of-change". For example the acceleration, location, and pressure data are each summarized in this way, and the summary node renders this data to the downstream analysis stages, thereby obscuring the individual data values. The issue remaining is the method for determination of the clusters.

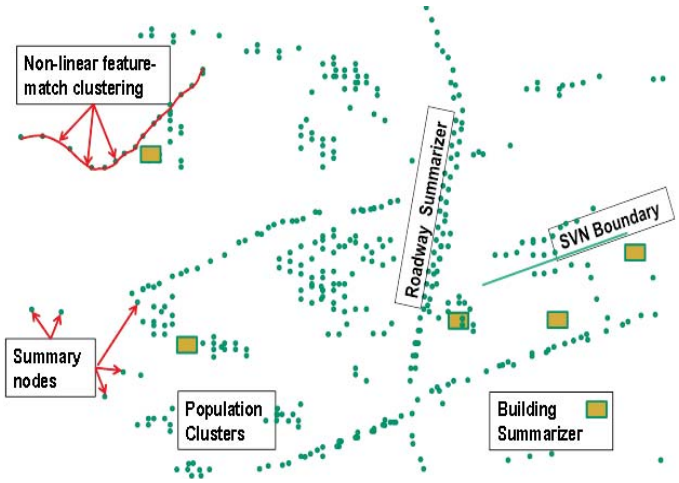


Figure 3: Summary Nodes and Other Summarizer Entities

The fourth approach examined was to granularize location reports as shown in Figure 4. A multi-resolution grid (multi-grid) is constructed so that areas of high device-density are divided into smaller latitude-longitude "cells" and vice-versa. This allows each cell to contain a statistically significant number of devices without over-gridding sparsely populated areas.

Finally, another approach was to use "trip wires" for nodes moving along a roadway rather than tracking their movement.

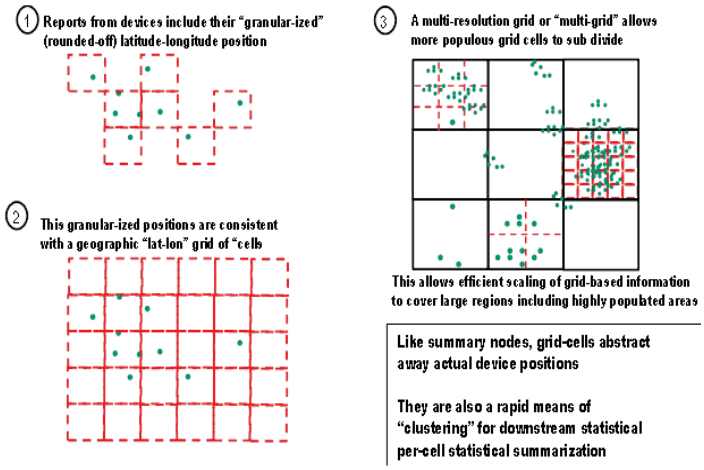


Figure 4: Multi-resolution grid (Multi-grid) Uses Low Resolution Latitude / Longitude Grid-cells for Classification of Devices and Summary Nodes

IV. Design of Cloud Analytics

A key objective in developing the privacy-based crowd-source analytics was to design and implement several types of analytical approaches to determine which was best at providing actionable information while suppressing privacy information.

Once the events were generated, we were concerned with a density of events occurring within a common area. Identifying this area enabled us to map out alerts for the region of interest. We investigated the use of clustering techniques that combined knowledge of geographical features such as rivers, roads, buildings, and processing facilities with sensor events and derived rate of change measurements reported within a given area or region.

IV.1 Clustering and Classification Algorithm Background

This project investigated clustering and classification algorithms for implementing our solution. As background, three types of these solutions, k-Means Clustering, Self Organizing Maps, and "clustering around geographical features" are discussed below. Techniques employed on these algorithms guided our research toward the use of k-Means [5] and/or k-medoid [6] and hierarchical geo-feature based clustering and classification on this project.

IV.1.1 k-Means Clustering

This project investigated the use of k-means clustering to define geographical alert regions, as well as determine the location of summary nodes. The algorithm takes the events generated from the raw sensor readings, GPS location, timestamp information and the derived data as inputs. It outputs the location of the centroid of the event cluster.

The k-means algorithm uses iterative refinement and Euclidean distances to define the clusters. It starts with a given number, *k* of random candidates for the cluster centers and then moves the centers while seeking to minimize the intra-cluster variability. The goal of using k-means clustering was determination of geographical sub-regions that possess a large rate of change in air pressure, or that exhibit distinctly different air pressure values from the average over a large overall region.

The k-medoid algorithm (Figure 5) works similarly to k-means but always identifies a particular data-point (rather than a mean location) as the "most central" for its cluster.

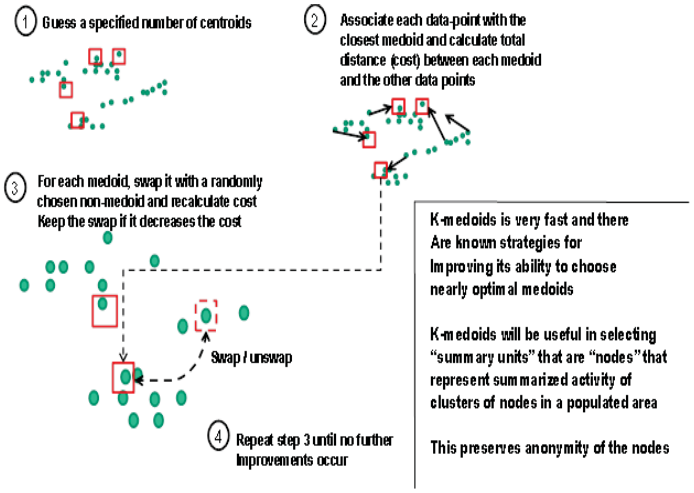


Figure 5: Diagram of the Mechanism by Which k-medoid Works

IV.1.2 Self-Organizing Maps

The Self-Organizing Map (SOM) is a type of unsupervised neural network that uses an iterative unsupervised learning algorithm to map a set of vectors from a high dimensional space onto a low dimensional space while preserving the spatial geometry [7]. The SOM is arranged as an n-dimensional grid whereby the location of a node in the grid corresponds to a specific region of the higher dimensional state space that is being clustered. The neural network node is activated when the network is presented with a vector that excites the node more than its peers (winner-take-all). The node's weight vector is then updated slightly to more closely align with the vector. This particular solution requires that vectors be presented in random order from the grid. The SOM does not require that the analyst or analytic "know" the exact number of clusters beforehand.

IV.1.3 Geometric Figures

In combination with multi-grids, a pattern detection algorithm can be applied to detect curvilinear shapes within the locations of the reporting nodes with the objective of "organic" detection of features and boundaries that characterize devices experiencing anomalous sensor values as shown in Figure 6. This has not yet been tested but leads naturally to the use of geological features to get the same or better information.

IV.1.4 Geological Features

Summary nodes can be classified according to their proximity to (or within) various geological structures of interest for detection of events (Figures 7 and 8).

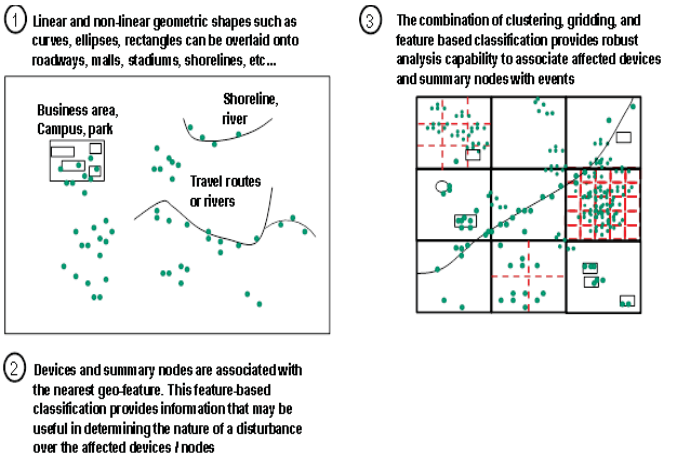


Figure 7: Process for Implementing Non-linear Geo-feature Clustering Geographical Clustering Analytic

For instance, summary nodes that can be associated with an averaged velocity vector can be considered as nodes moving along a roadway. Stationary nodes may be associated with various geographically located entities. As such, these nodes can be grouped at a higher level as linear structures such as roads and shorelines, while others may be associated with buildings including hazardous processing plants and the like. This association enables the system to relate novel events on these nodes with a particular entity and thereby infer the most likely real-world event.

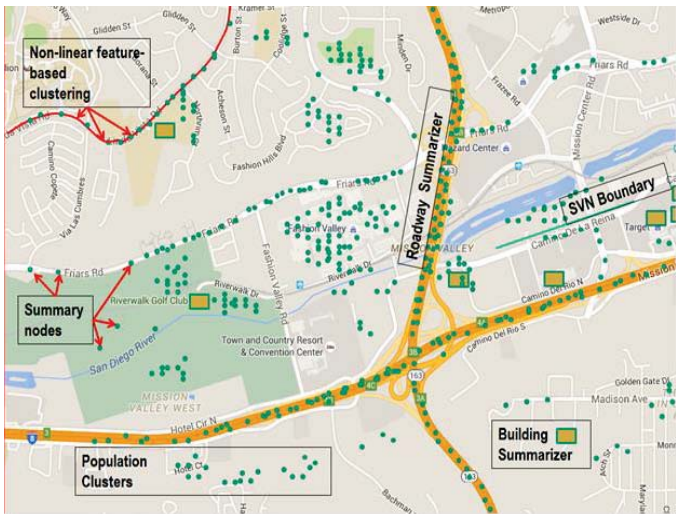


Figure 8: The underlying map for Figure 7. This is the local area for the Mission Valley business district in San Diego. RAM Laboratories resides in the building just to the right of Hwy 163.

Since the summary nodes are not segregated based on linear distances from each other, this is a non-linear classification method and captures information not available from Euclidean distance based classifiers alone.

Due to limited time, these mechanisms were not tested but were considered as proof-of-concept regarding how they would be used to assimilate low level events into high-level actionable information for the location and characterization of weather alerts.

IV.2 Algorithm Evaluation

To evaluate the effectiveness of the clustering algorithms, the project applied control data sets to each of the inputs. The goal of the proof of concept evaluation was to generate the detections and event densities and compare the regions identified through each approach with the actual occurrence of a weather event that causes a depression of pressure over a wide area.

IV.3 Clustering and Classification Analytics

The clustering methods we tested were a representative suite of algorithms to be used for data reduction on the Cloud. Figure 7 shows the "summary nodes" as "dots, many of which correspond to a roadway or geographical entity. Each one represents a statistically significant number of potentially reporting devices. The devices are segregated into these nodes via one of the geographical clustering methods listed above. For instance, k-means / k-medoid can be used to segregate nearby clusters that correspond to no obvious geographical boundary. Referring back to Figure 7, the nodes can be segregated into a grid of cells at larger scales, each representing nodes within a range of latitude / longitude coordinates.

The segregation into nodes and then into grid cells allows reduction of the data by converting it into the percentage of devices reporting within the node / cell.

The resulting "summarizers" greatly reduce the data flows within the cloud since only summary statistics are required over the entire summarizer rather than for each device within it. Most importantly, the summarizers represent real-world entities that may be associated with an event and so will play a part in the high-level analysis and characterization. For example, the analytics can be applied to detecting traffic jams due to accidents or road blockage (e.g. mud-slide or collapsed / washed-out bridge).

V. Simulation Testbed for Analytics Workflow

The proof of concept was demonstrated in part by the implementation of a simulation program that simulates devices with sensor data streams that could be scripted to change the statistics of their data outputs over time. Primary features of the simulator were:

- Configurable
  - Device locations and report rates
  - Per sensor scripting
  - Report criteria
  - Simulation time line
- Generate of events from devices at script-specified moments in time
- Model device-sensor events, reports, and summaries
  - Device level
  - Cloud level
- State-ful report tracking
- Classification via hierarchical grid cells for determination of affected areas

The simulator and analytic tools created during the project were comprised of about 4,000 lines of C++ source code embodied in eight packages including a mathematics package, statistics / time series package, as well as six other packages; the latter as shown in Figure 9. For brevity, the math and statistics packages are not shown.

Figure 9: Primary software packages and the major C++ classes they contain. These depend on custom math and statistics packages developed for this project. The latter packages (not shown) constituted a major portion of the code.

In what follows we detail the data flows for the architecture, but first show our means of live data visualization (Figure 10) that can be inserted at any point in the simulator's processing chain. The figure is a view of an open source utility called Live Graph developed by the Centre for Intelligent and Complex Systems at Monash University, Australia [8]. The tool plots data from a file specified by the user. By streaming data to the file from a separate application, the data is plotted in real-time by the tool. This allowed additional verification that proper distillation of statistics was occurring the various stages of the simulation.

V.1 Simulator Structure

Figure 11 shows a high-level block diagram of the device / sensor simulation and downstream processing stages. The generation of sensor data and subsequent report generation runs in its own thread. Each sensor is simulated as a normal random variable with specified mean and standard-deviation that outputs a value several times a second (the rate is configurable). Only when a stream from a sensor deviates significantly (as determined by a configurable threshold from a historical baseline stored for that sensor) does the device report the anomaly. The report consists of a granular (rounded) timestamp, a granular GPS coordinate, and the detected deviation from the mean in terms of number of standard deviations.

Concurrently with this, the main thread reads a script file that specifies a change in the mean and standard deviation for a given sensor's data stream. The script also specifies the number of seconds before the change is to be made. When the specified time elapses, the update is made to the sensor which subsequently will generate data according to the new distribution.

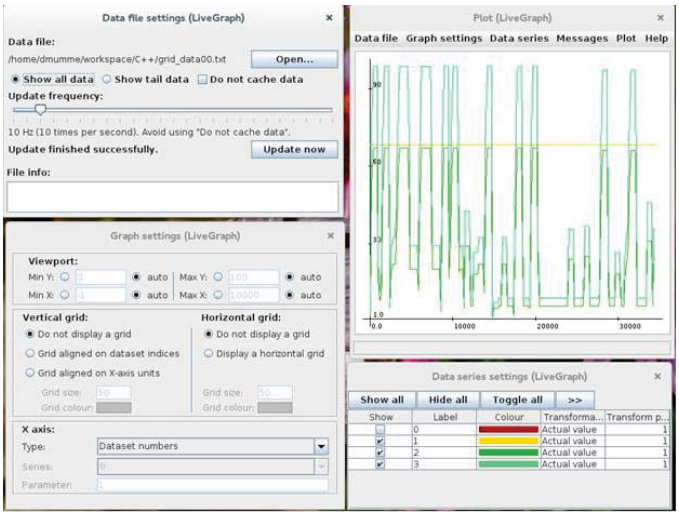


Figure 10: The simulator leverages LiveGraph, an open source public domain tool for live visualization of data

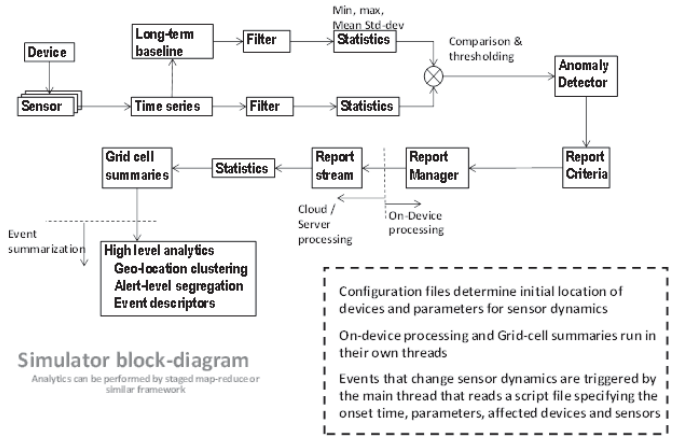


Figure 11: Simulator block diagram. For clarity, the file parsers that read the configuration files and the device script are not shown.

A third thread simulates the downstream analytics performed in the Cloud. This thread receives reports streaming from the sensor thread and uses the granular time stamp and position as a key on which to segregate the reports for further processing. This processing is also performed by this thread and consists of summarizing population counts and percentages for each grid cell.

V.2 MapReduce-Based Processing

As proof of concept for deployment in the cloud, we broke out the simulator's processing pipeline into successive MapReduce stages (Figure 12). The refactoring of the processing to accommodate a map-reduce frame work was performed to demonstrate the flexibility of the architecture and one potential way forward for development. Examples of mapping clustering algorithms to mappers and reducers are round in [9][10]. The

MapReduce pipeline uses Hadoop Streaming to pipe data from one stage to another [11].

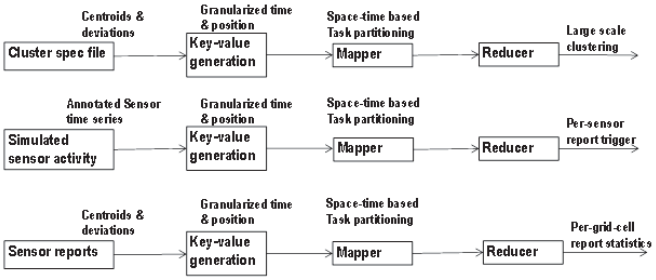


Figure 12: Decomposing the Simulator Pipeline into MapReduce Processes

The topmost pipeline in the diagram shows how the time stamp and position act as a key to the mapper so that it may segregate the summarization task to multiple reducers, each one performing clustering of the devices within a particular cell. The result of a larger number of reducers is a "large scale" global clustering of all the devices over a wide area.

The second pipeline is similar but is for the determination of which devices have sensors that actually trigger a report. From the analytics perspective, this pipeline will most like not exist since this processing is done by the on-device application installed on each individual device. The third pipeline represents processing on the Cloud. Here the mappers again segregate devices into grid cells and pass the reducer the reports from a particular cell's devices. The reducer then summarizes the number and percentage of reporting nodes.

Other computations may be decomposed into such a parallelized approach. This includes generating keys from the clusters produced by a clustering algorithm (such as k-medoid) so that statistical summaries for the cluster can be generated by the reducers.

V.3 Cluster Generator

For testing clustering algorithms such as k-medoid we developed a cluster generator (Figure 13) that provides a large number of randomly generated device coordinates centered on numerous "cluster centers", each cluster having a pre-specified variance from its center. The data was also used to generate clusters lying in the local metro-area of RAM Laboratories' primary San Diego office.

The generator reads an input file specifying the cluster parameters, each cluster specified on a single line. The specification includes the number of devices in the cluster (cluster size), the location representing the mean and the variance for the randomly generated positions. This specification acts as the "ground truth" to test against the output of any clustering algorithm with this positional data as input.

Further development of the test bed would generalize the cluster generator to change the positions of devices over time according to a script that changes speed and direction for each device and changes in the device's off / on state. The changes over time would be piped into the simulator in real-time to test its classification and summarization capabilities in a dynamic setting.

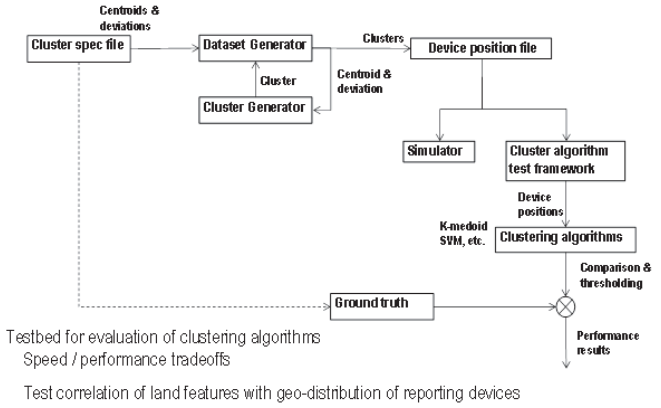


Figure 13: Cluster generator populates a geographical area with device coordinates that are clustered around pre-specified medoids with a pre-specified variance.

VI. SUMMARY

The project successfully proved the feasibility of the privacy-based analytics. On the project, several classification methods were implemented that anonymized assimilation of sensor reports including: (1) K-means clustering, (2) Grid-cell classification, and (3) Classification according to association with geographical-features. Additionally, a cluster generator was used to assist us with our proof-of-concept. This cluster generator (1) generated positional coordinates for devices that are clustered around pre-specified medoids with a pre-specified variance around it, and provided the results in a file format that is readable by the simulator. Likewise, a data generation simulator was developed that simulated sensors on devices, generating a data streams, acquiring the locations of these sensors from the cluster generator, and simulated impacts on sensor data stream statistics over time. The simulations demonstrated successful recovery of the ground-truth clusters. The proof-of-concept included incorporation of a tool for live visualization of summary data.

Scalability of our solution was successfully demonstrated using our testbed. Based on our analysis, this approach can potentially scale to 10^9 or more devices for e.g. a single nation or continent. The solution also provides added scalability by using data summary over granularized latitude-longitude coordinates and provides a capability for handling local geo-event storms by requiring fine-grained grids

VII. ACKNOWLEDGEMENTS

This work was performed under a Phase I Small Business Innovative Research project funded by the Department of Homeland Security, Contract HSHQDC-15-C-00020. The authors would like to thank Stephen Dennis of DHS for his valuable feedback on the project.

VII. REFERENCES

[1] The Roadmap for Privacy by Design in Mobile Communications: A Practical Tool for Service Providers and Users. Arizona State University Privacy Design Research Lab. December 2010

[2] Cavoukian, A., Bansal, N., Koudas, N. "Building Privacy into Mobile Location Analytics (MLA) Through Privacy by Design," March 2014.

[3] Work, D., and Bayen, A. "Impacts of the Mobile Internet on Transportation Cyberphysical Systems: Traffic Monitoring Using Smartphones." National Workshop for Research on High-Confidence Transportation Cyber Physical Systems: Automotive, Aviation, and Rail. Washington, DC 2008

[4] Hoh, B. et al. "Virtual Trip Lines for Distributed Privacy-Preserving Traffic Monitoring." International Conference on Mobile Systems, Applications and Services, 2008.MacQueen, J. "Some Methods for classification and Analysis of Multivariate Observations". Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297, 1967.

[5] Kanungo, T.; Mount, D. M.; Netanyahu, N. S.; Piatko, C. D.; Silverman, R.; Wu, A. Y. (2002). "An efficient k-means clustering algorithm: Analysis and implementation" (PDF). IEEE Trans. Pattern Analysis and Machine Intelligence 24: 881–892

[6] Kaufman, L. and Rousseeuw, P.J. (1987), Clustering by means of Medoids, in Statistical Data Analysis Based on the L_1–Norm and Related Methods, edited by Y. Dodge, North-Holland, 405–416

[7] Kohonen, T. Self-Organizing Maps. Third, Extended Edition. Springer, 2001.

[8] http://www.live-graph.org

[9] Zhao, W., Ma, H., He, Q. "Parallel k-means Clustering Based on MapReduce," Lecture Notes on Computer Science, 1st International Cloud Computing Conference 2009.

[10] Weichel, C. "Adapting Self-Organizing Maps to the MapReduce Programming Paradigm" Science Technology Education Partnership Conference, 2010.

[11] Hadoop Streaming. https://hadoop.apache.org/docs/r1.2.1/streaming.html

Biography

Dean C. Mumme is the Chief Scientist at RAM Laboratories. His interests include cyber security, optimization algorithms, and machine learning. He is currently developing cyber security mechanisms for nodes via hardware virtualization that renders these protections out-of-band to attack. Dr. Mumme received his B.S. in Aeronautics and Astronautics from MIT, his M.S. in mathematics from Idaho State University, and his Ph.D. in Computer Science from the University of Illinois.

Robert M. McGraw co-founded RAM Laboratories, Inc. in 1997 and is now its Vice-President and Chief Technology Officer. Dr. McGraw's interests lie in the areas of multi-resolution modeling, cyber security, and distributed information management. He is currently overseeing RAM Laboratories efforts in developing an Application Security Analyzer for MDA and enhancing the CARET Agent Framework. Dr. McGraw received his B.S. in Physics and Electronics Engineering from the University of Scranton, and received his M.S. and Ph.D. in Electrical Engineering from the University of Virginia.

Richard A. MacDonald co-founded RAM Laboratories, Inc. in 1997 and has been serving in the capacity of President and Chief Executive Officer since the company's inception. With twenty years experience in high technology and defense businesses, Dr. MacDonald has overseen the strategic planning and growth of RAM Laboratories into a recognized defense technology leader with a national customer base spanning coast-to-coast, supporting multiple branches of the Department of Defense including the Army, Navy, Air Force, Coast Guard and the Department of Homeland Security. Dr. MacDonald earned both a Ph.D. and Masters of Science (M.S.) in Electrical Engineering from the University of Virginia, and a Bachelor of Science (B.S.) degree in Computer Engineering from the University of Michigan.

SESSION

POSTER PAPERS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Generic Object Recognition using Bag-of-Features and Image Concept-Base

[1]Hirokazu Watabe, [2]Misako Imono, [3]Eriko Yoshimura and [4]Seiji Tsuchiya
[1,3,4]Dept. of Intelligent Information Engineering and Science,
Doshisha University, Kyotanabe, Kyoto, 610-0394, Japan
[2]Dept. of Information Systems, Daido University, Nagoya, 457-8530, Japan
[1]hwatabe@mail.doshisha.ac.jp, [2]m-imono@daido-it.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 recognition method (bag-of-features) and word-concept association techniques to perform generic object recognition .*

Keywords: generic object recognition, bag-of-features, degree of association, Concept-Base

1 Generic object recognition system

Our goal in this research is object recognition based on an image database and a image-concept association system. The procedural flow is as follows (fig.1):

(1) Construct the image KB(knowledge-base).
(2) Construct the image CB(concept-base).
(3) Input the image and derive BoF of the image.
(4) Calculate the degree of association(DoA) between BoF of input image and BoFs in the Image KB.
(5) Output the object name whose DoA is the largest.

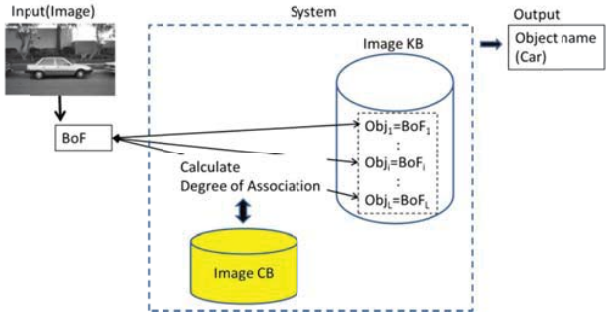


Fig.1: Proposed system

2 Construct the image knowledge-base

We use a method for recognition of categories of objects called bag-of-features[1](BoF). The main idea consists of finding keypoints, visual words, which are cluster centers of the affine invariant descriptors of image patches, such as SURF[2]. A certain object in an image Obj has a pair of sets of visual word VW_i and weighting (rate of frequency) w_i, as the equation 1 and figure 2.

$$Obj = \{(VW_1, w_1), (VW_2, w_2), \cdots, (VW_M, w_M)\} \quad (1)$$

The object recognition method is that distribution of visual words in a input image is compared to visual words found in the image knowledge-base.

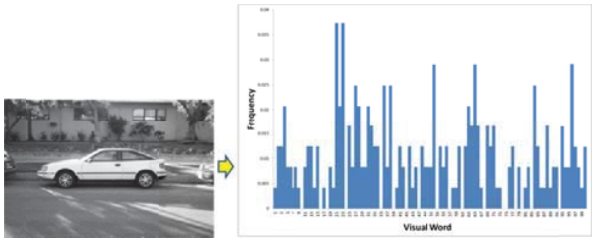


Fig.2: BoF

3 Construct the image concept-base

3.1 Word concept association system

Concept-Base

A concept-base (CB)[3] 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.

$$A = \{(a_1, w_1), (a_2, w_2), \cdots, (a_m, w_m)\} \quad (2)$$

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.

Degree of Association

The degree of association(DoA)[3] 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. To find the degree of association, the relevance between two concepts is calculated as a numeric value based on the value found by calculating the degree of matching(DoM) of the secondary attributes of the concepts.

$$DoM(A,B) = \sum_{a_i = b_j} \min(u_i, v_j) \quad (3)$$

$$DoA(A,B) = \sum_{i=1} \{DoM(a_i, b_{xi}) \times (u_i + v_{xi}) \times (\min(u_i, v_{xi}) / \max(u_i, v_{xi})) / 2\} \quad (4)$$

3.2 Image Concept-Base

As shown above, the object in an image is represented by the equation 1, and the word concept is represented by the equation 2. The two equation have the same formation. Therefore, we constructs image concept-base(ICB) and using the ICB, we calculates the degree of association between objects in images. However to calculate the DoA, we must conceptualize all visual words (VW). Figure 3 shows the method to conceptualize the visual word. Each weighting of the attribute of the visual word derived from *tf-idf* method.

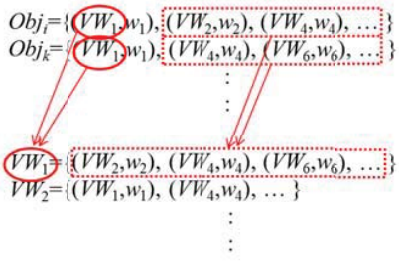


Fig.3: Conceptualize visual words

4 Evaluation

We then evaluate the proposed method by precision comparing with conventional methods, histogram intersection(HI) and support vector machine(SVML, SVM). We use an image dataset, 30 objects x 50 images from Caltech-256[3]. Each 5 images is used as a input image for evaluation and using the rest each 45 images, the image knowledge-base is constructed.

Figure 4 and 5 show the results in case of the number of visual words is 100 and 500 respectively. In both cases, the best result is the degree of matching(DoM).

Figure 6 shows the precisions against the number of visual words in case of DoM, HI and SVM. The best performance is at that the number of visual word is 700 by DoM.

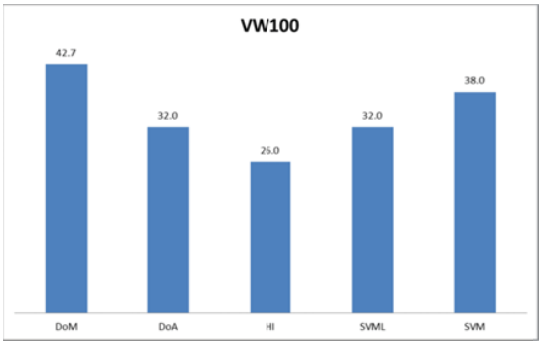


Fig.4: Experimental results (VW100)

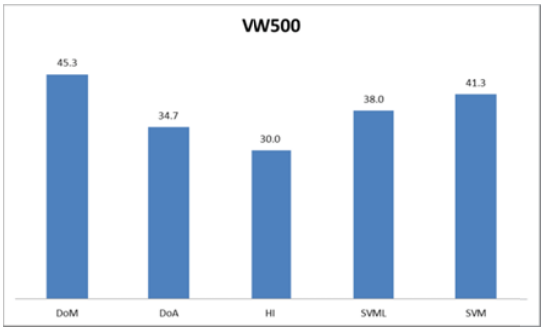


Fig.5: Experimental results (VW500)

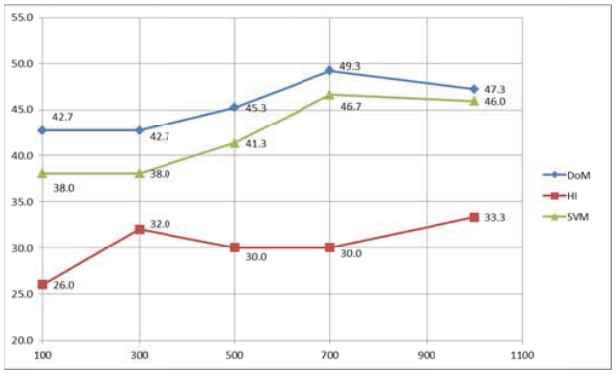


Fig.6: Experimental results (number of VW)

5 Conclusion

In this paper, we proposed combined method by existing image recognition method (bag-of-features) and word-concept association techniques to perform generic object recognition .

Acknowledgements

This work was partially supported by JSPS KAKENHI Grant Number 16K00311.

References

(1) Csurka, G., Bray, C., Dance, C. and Fan, L., "Visual Categorization with Bags of Keypoints", *Proc. ECCV Workshop on Statistical Learning in Computer Vision*, pp.1-22, 2004.

(2) H. Bay, A. Ess, T. Tuytelaars, L. V. Gool: SURF: Speeded Up Robust Features , *Computer Vision and Image Understanding*, Vol. 110, No. 3, pp.346-359, 2008.

(3) H. Watabe, M. Imono, E. Yoshimura, S. Tsuchiya: Calculating Degree of Association Incorporating Viewpoint Using a Concept-Base, *Proc. of ICAI2012*, Vol.I, pp.191-197, 2012.

(4) Caltech 256 image dataset: http://www.visioin.caltech.edu/Image_Datasets/Caltech256/

(5) Cyganek, B.: Object Detection and Recognition in Digital Images, Theory and Practice, John Wiley & Sons, Ltd, 2013.

Judging Emotion from EEGs Using SVM and Principal Component Analysis

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 from EEGs using SVM. Accuracy of emotion judgment from EEG features using only SVM was 37.8%, and using SVM and principal component analysis was 49.2%. 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 from EEGs using SVM and principal component analysis.

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).

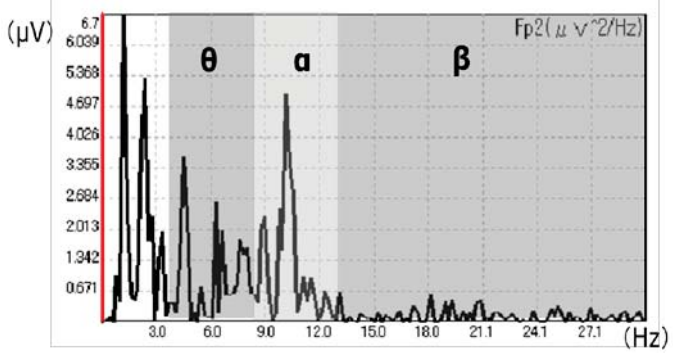


Fig.1 Spectrum analysis of the source EEGs

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 anger, sadness, fear and pleasure.

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.

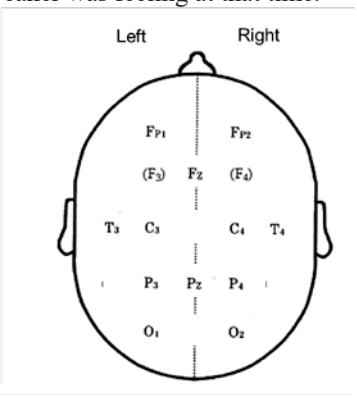


Fig2. 14 locations to measure EEG conforming to the International 10-20 system

Twenty 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:

$$Linear\_al_{ij} = al_{ij} + \left\{ \left(\frac{q_1 - q_2}{p_2 - p_1} \times l + q_1 \right) - \left(\frac{q_2 - q_1}{p_2 - p_1} \times l + q_2 \right) \right\} / 2 \quad (4.1)$$

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 Principal Component Analysis

The principal component analysis is a technique to integrate the data of the multivariate, and to produce the overall new index. In a word, small number of synthesis variables are made applying weight to a lot of variables. As a result, multi-dimensional data can be processed to the data of the low level. Moreover, the overall index (principal ingredient) produced newly is mutually orthogonalized, and can be expected to obtain the solution efficiently even by the data of the low level.

In this paper, EEG features of six dimensions compressing 42 dimensions' EEG features (three bandwidths and 14 locations) by principal component analysis were used.

6 Evaluation Experiment

6.1 Experimental Method

The method of evaluations 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 2945 EEG features obtained by excluding outliers from the total of 3075 EEG features. The emotions of the 2945 EEG features used in this study comprised 629 anger features, 857 sadness features, 979 fear features, and 480 pleasure features.

The side-by-side test was done in the following three techniques: at random judgment, judgment using only SVM ,and judgment using SVM and principal component analysis.

6.2 Evaluation of Accuracy

Figure 3 shows the result of the emotion judgement from EEGs. Accuracy of emotion judgment from EEG features using only SVM was 37.8%, and using SVM and principal component analysis was 49.2%. As a comparison, the result of doing the emotion judgment at random is accuracy of 25.0%. The method proposed herein thus appears valid.

However, performance accuracy remains low, and continued development is required through further development of methods for both reducing noise mixed in with EEGs.

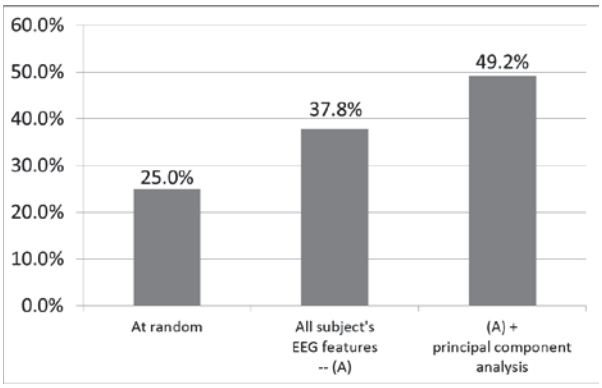


Fig.3 result of the emotion judgement from EEGs

7 Conclusion

We have presented a technique for gauging the emotions felt by a person from EEGs. Accuracy of emotion judgment from EEG features using only SVM was 37.8%, and using SVM and principal component analysis was 49.2%. The method proposed herein thus appears valid. 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.

Deep Recurrent Neural Network and Psychoacoustic Modeling for Speech Enhancement

Yu Yong Jeon[1], **Gyu Seok Park**[1], **Jang-Woo Kwon**[2] **and Sang Min Lee**[1,3,*]
[1] Department of Electronic Engineering, Inha University, Incheon, 22212, South Korea,
[2] Division of Computer Engineering and Information, Inha University, Incheon, 22212, South Korea,
[3] Institute for Information and Electronics Research, Inha University, Incheon, 22212, South Korea,
nicejyy@gmail.com, gyuseok.park@gmail.com, jwkwon@inha.ac.kr, sanglee@inha.ac.kr
[*] Contact Author
Type of the submission : Extended Abstract/Poster Paper

Abstract - *Monaural speech enhancement is one of the important topics in signal processing because it can be used for many real-world applications. In this paper, we propose a monaural speech enhancement method that is combination of deep recurrent neural network (DRNN) to separate clean speech from noisy speech, and time-frequency psychoacoustic modeling for speech enhancement. For separating clean speech, 2 schemes of deep recurrent neural network with 3 hidden layers are used in this study. One is with fully temporal connection called stacked RNN and other is with temporal connection at a specific layer. Results of each network are compared each other and to result without psychoacoustic masking.*

Keywords: Deep Recurrent Neural Network, Speech Enhancement, Time-Frequency Masking, Psychoacoustic Modeling

1 Introduction

Recently, many approaches using deep learning have been proposed to separate the clean speech from noisy speech. These approaches have enhanced the accuracy of automatic speech recognition (ASR) [1, 2] and increased the quality of speech in noisy environment [3, 4]. However, separation is still difficult and separation results are far from human auditory capability. Huang and et al [5] tried to separate two speeches from mixed signal using DNN and RNN. They proposed the joint optimization of the network with a soft masking function base on posteriori SNR. Moreover, they proposed a discriminative objective function that takes into account the similarity between the prediction and other source, and between the prediction and the current target. In our previous work [6], we studied speech enhancement algorithm with the psychoacoustic modeling. In general, loudness of speech in noise environment is degraded by noise. So that is why we tried to separate speech from noise and reinforce the loudness of speech as much as degraded in noise environment based on partial loudness model. In this paper, we explore the use of a deep recurrent neural network (DRNN) for speech separation and propose a joint optimization using psychoacoustic modeling for speech enhancement extended from our previous work. Figure 2 shows a flow of proposed approach. Speech and noise are separated by DRNN which is trained with training dataset and speech is enhanced with joint optimization.

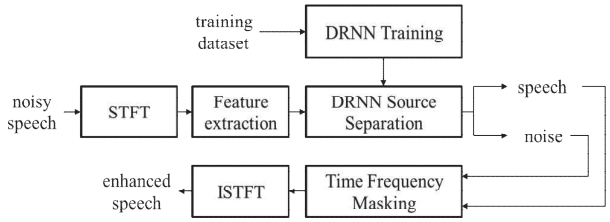
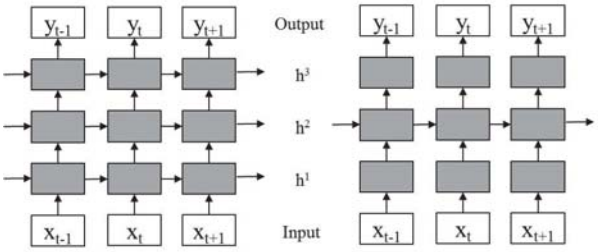


Figure 1. Block Diagram of Proposed Approach

2 Proposed Method

2.1 Recurrent Neural Network

To separate clean speech from noisy signal, we used two schemes of deep recurrent neural network (DRNN) as shown in Figure 2. Figure 2 (a) shows a 3 hidden layer DRNN with full temporal connections call stacked RNN (sRNN) and (b) is a 3 hidden layer DRNN with temporal connection at the 2nd layer. We denote this network as $DRNN_2$. At time t, training input of network is denoted x_t which is spectral feature from a mixture of speech and noise within a window, and trained



(a) 3-layer stacked RNN (sRNN) (b) 3-layer $DRNN_2$

Figure 2. Deep Recurrent Neural Network architectures

output feature is denoted y_t. In an RNN, hidden layers output is calculated based on current input and output of previous time step.

2.2 Psychoacoustic modeling

In our previous work [6], we tried to enhance the speech quality in noise environment using psychoacoustic modeling called partial masking. Although an algorithm proposed in previous work derived pretty good result, psychoacoustic model only in frequency domain was considered. To enhance speech quality, in this paper, we extend our previous work and propose a time-frequency masking approach using psycho-acoustic modeling called auditory masking effect. Auditory masking effect is divided into temporal masking and simultaneous masking. Temporal masking is occurred when a masker sound make other sound which are present preceding or following the masker inaudible. And simultaneous masking is occurred when a masker sound make other sound which are close to masker sound in frequency domain inaudible. Actually, these phenomena are used in many applications, but rarely used in speech enhancement. In this work, we explore joint optimization using temporal and simultaneous masking effect for speech enhancement.

3 Experiment

3.1 Network Setting

To evaluate the performance of the proposed approaches, we used TIMIT corpus from five male and five female speakers. Eight TIMIT sentences from each speaker are used for training, and one sentence from each speaker and another one is used as development set and test set respectively. Test dataset is mixed with babble noise from NOISEX92 at 0dB.

The sRNN and DRNN has 3 layers and each layer have 160 nodes. Dimension of input and target is 513 because speech is framed to 1024 samples and transformed to frequency domain by 1024-point short-time Fourier transform.

3.2 Metric

The evaluation of speech separation is measured by Source to Interference Ratio (SIR), Source to Artifacts Ratio (SAR), and Source to Distortion Ratio (SDR), according to the BSS-EVAL metrics [7]. Higher values of each evaluation mean better separation quality. And evaluation of speech enhancement is measured by Perceptual Evaluation of Speech Quality (PESQ). Higher PESQ value means better speech quality.

4 Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2010-0020163) and the MSIP(Ministry of Science, ICT and Future Planning), Korea, under the C-ITRC(Convergence Information Technology Research Center) (IITP-2016-H8601-16-1003) supervised by the IITP(Institute for Information & communications Technology Promotion)

5 References

[1] Sak, Haşim, et al. "Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition." arXiv preprint arXiv:1507.06947 (2015).

[2] Graves, Alan, Abdel-rahman Mohamed, and Geoffrey Hinton. "Speech recognition with deep recurrent neural networks." Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013.

[3] Weninger, Felix, et al. "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR." Latent Variable Analysis and Signal Separation. Springer International Publishing, 2015. 91-99.

[4] Xu, Yong, et al. "A regression approach to speech enhancement based on deep neural networks." Audio, Speech, and Language Processing, IEEE/ACM Transactions on 23.1 (2015): 7-19.

[5] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, Paris Smaragdis. "Deep learning for monaural speech separation." Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014.

[6] Jeon, Yu-yong, and Sang-min Lee. "A speech enhancement algorithm to reduce noise and compensate for partial masking effect." Journal of Central South University of Technology 18.4 (2011): 1121-1127.

[7] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," Audio, Speech, and Language Processing, IEEE Transactions on, vol. 14, no. 4, pp. 1462 –1469, July 2006.

Dual Sub-swarm Interaction QPSO Algorithm Based on Different Correlation Coefficients

Tao Wu[1], Xi Chen[2*], Xi Wu[1]

1. Department of Computer Science, Chengdu University of Information Technology,Chengdu, 610225, China
2. School of Computer Science and Technology, Southwest University for Nationalities, Chengdu, 610041, China
Address correspondence to Xi Chen, No. 16, 4th Section of 1st Ring Road South, Chengdu, Sichuan, China,
Email: wut@cuit.edu.cn; cx@swun.edu.cn; xi.wu@cuit.edu.cn
Type of the submission：Extended Abstract/Poster Paper

***Abstract*—A novel quantum-behaved particle swarm optimization (QPSO) algorithm, the Dual sub-swarm Interaction QPSO Algorithm Based on Different Correlation Coefficients (DCC-QPSO) is proposed by constructing the master-slave sub-swarms with different potential well centers. In the novel algorithm, the master sub-swarm and the slave sub-swarm have respective functinons during the evolution through different information processing strategies. The master sub-swarm is conducive to maintaining the population diversity and enhance global search ability of particles, otherwise the slave could accelerate the convergence rate and strengthen particles' local search ability. Experiment results show that DCC-QPSO outperforms the traditional QPSO algorithm regarding optimization of multimodal functions.**

Keywords*—particle swarm optimization; quantum-behaved particle swarm optimization; sub-swarm; interaction

BINARY CORRELATION QPSO ALGORITHM

According to the analysis of particles' orbits in the PSO algorithm done by Clerc and Kennedy[1], a δ potential well is established in QPSO[2] at the local attraction point $p_i = (p_{i1}, p_{i2}, \cdots, p_{iD})$ to impact particles in the population.The point of p_i take advantage of the particles personal optimum *pbest* and the global optimum *gbest* mainly depends on the acceleration factor $c1$、$c2$ and random factors $r1$ 、$r2$.

As the important parameters of standard PSO algorithm, there are a lot of related studies about how to value the acceleration factors $c1$ and $c2$[3,4]. However, they ignored the impacts of the random factors $r1$ and $r2$ on algorithm performances. It is necessary to analyze the random factors in order to further study the impacts of utilization of particles' own experiences and community sharing information on the performance of the QPSO algorithm respectively.

To analyze the connection between $r1$ and $r2$ in QPSO, reference [5] suggested the conception of the binary correlation factors and proposed the Binary correlation QPSO algorithm, referred as BC-QPSO algorithm. The BC-QPSO algorithm constructed the relations between $r1$ and $r2$ using the bivariate normal Copula function: $\Phi_\rho(\Phi^{-1}(r1),\Phi^{-1}(r2)$, the Fréchet-Hoeffding lower bound: $W(u,v)=\max(u+v-1,0)$, the Fréchet-Hoeffding upper bound: $M(u,v)=\min(u,v)$ and the product Copula: $\Pi(u,v)=uv$.

THE DUAL SUB-SWARM INTERACTION QPSO ALGORITHM BASED ON DIFFERENT CORRELATION COEFFICIENTS

A. Learning Stratigies

The analysis of BC-QPSO done by reference [5] shows that the diversity of population decreases along with the increase of correlation coefficient $\rho(-1\leqslant\rho\leqslant1)$ of the binary correlation factors (i.e. $r1$ and $r2$). In the BC-QPSO model given that ρ=-1, particles' unbalanced uses of *p*best and *g*best in determining the position of the potential well center deviate the potential well from the *g*best, resulting in a trend of expansion of the population, which helps extending the search space as well as the population diversity. In the BC-QPSO model given that ρ=1, particles make the most of *p*best and *g*best in a highly balanced manner, making the potential well rapidly move towards extremum, resulting in a quick contraction of population, which helps speeding up the convergence. Based on the above analysis, a conclusion can be drawn that the higher the degree of the correlation between binary correlation factor $r1$ and $r2$, the faster the convergence of the algorithm; the lower the degree of the correlation, the greater diversity of the population and the higher precision of convergence. In order to guarantee the desired convergence precision and efficiency, by the combination of the fully positively correlated BC-QPSO (i.e. ρ=1) and the fully negatively correlated BC-QPSO (i.e. ρ=-1), we derive a new QPSO algorithm – DCC-QPSO.

Suppose that the entire population is S and the size of the population is N; the master group is denoted as S_1, its population size is N_{S1}, and its global best is $gbest_1$; the slave group is S_2, its population size is N_{S2}, and its global best is $gbest_2$; thus we have $S_1 \cup S_2=S$ and $N_{s1} + N_{S2}=N$.

The basic ideas of DCC-QPSO are: in a randomized and quantum-behaved particle swarm, we divide the whole population into two fully independent sub-groups. One group adopts the fully negative correlation strategy against *pbest* and *gbest* when determining its potential well, that is, we carry out the iterated searches in the solution space given ρ= -1 (i.e. the correlation coefficient between $r1$ and $r2$). This group is

referred to as master group S_1. The other group adopts the fully positive correlation strategy against the *pbest* and *gbest* when determining its potential well, that is, we carry out the iterated searches in the solution space given ρ= 1. This group is referred to as slave group S_2.

The evolution equation of each particle in the DCC-QPSO based on the δ potential well is as follows:

$$\begin{cases} X_{i,j}(t+1) = p_{i,j}(t) \pm \alpha_1 \left| C_j(t) - X_{i,j}(t) \right| \left| \ln\left[1/u_{i,j}(t)\right] \right|, u_{i,j}(t) \sim U(0,1) \\ P_{i,j}(t) = \dfrac{c_1 r1_{i,j}(t)}{c_1 r1_{i,j}(t) + c_2 r2_{i,j}(t)} P_{i,j}(t) + \dfrac{c_2 r2_{i,j} s(t)}{c_1 r1_{i,j}(t) + c_2 r2_{i,j}(t)} G_j(t), 1 \le j \le D \\ H(r1, r2) = W(r1, r2), i \in N_{S_1} \\ H(r1, r2) = M(r1, r2), i \in N_{S_2} \end{cases} \quad (1)$$

B. Algorithm Execution Process

Based on the designs and definitions discussed above, the execution process of the DCC-QPSO algorithm is as follows:

Step 1: Setting Parameters.

Step 2: Initialization of population;

Step 3: Calculating the fitness values for all particles in S_1 and S_2;

Step 4: Calculating the average best positions of the entire population $C(t)$and evaluate the parameters $L_{i,j}(t)$ of the master and slave sub-groups respectively.

Step 5: Updating the position for particle $i(1 \leqslant i \leqslant N)$;

Step 6: Recalculating the particle i's current position $X_i(t)$ according to the objective function. (t for iterations)

Step 7: Updating the personal best positions;

Step 6：If $i \in S_1$ and the fitness value of $P_i(t)$ is better than the fitness value of the global best position of the whole swarm $P_g(t\text{-}1)$, i.e., $f(P_i(t)) < f(P_g(t\text{-}1))$, then $P_i(t)$is saved as the global best position of master group S_1 , which is denoted as $P_{gs1}(t)$; otherwise, $P_{gs1}(t)=P_g(t\text{-}1)$. If $i \in S_2$ and the fitness value of $P_i(t)$ is better than the fitness value of $P_g(t\text{-}1)$, i.e., $f(P_i(t)) <$ $f(P_g(t\text{-}1))$, then $P_i(t)$is saved as the global best position of the slave group S_2 , which is denoted as $P_{gs2}(t)$; otherwise, $P_{gs2}(t)=P_g(t\text{-}1)$.

Step 7: Comparing the fitness value of $P_{gs1}(t)$ and $P_{gs2}(t)$, if $f(P_{gs2}(t)) < f(P_{gs1}(t))$ then assign $P_{gs2}(t)$ to be the global best position $P_g(t)$; Otherwise , $P_{gs1}(t)$ is assigned to $P_g(t)$.

Step 8: Termination determination. If the maximum times of iterations *iter*Max or the error precision of fitness value is achieved, then stop the searching process and putout the results. Otherwise, let $t = t + 1$ and repeat the step 3 to step 8.

II. EXPERIMENT DESIGNS AND RESULTS ANALYSIS

A. Experiment Designs

In order to obtain reasonable experiment results, the benchmark functions, including Sphere function, Rosenbrock function, Rastrigin function, Griewank function, Ackley function and Schaffer function, are adopted to test QPSO, BC-QPSO, DIR-QPSO and DCC-QPSO for performance comparison.

The parameter decreases linearly from 1 to 0.5 in every algorithm to be tested. Dimension of the benchmark functions is set to be 20 and the maximum iterations are set to be 1000; The size of the population is set as 50, the size of the master group is 25 and that of the slave group is 25. Every benchmark function is tested times independently, and the average value of each test function is evaluated at the end of 30 iterations.

B. Experiments Results

The mean fitness value and the CPU time of each iteration using QPSO, BC-QPSO, DIR-QPSO[6] solving benchmark functions is as shown in Table 1.

TABLE I. THE RESULTS OF OPTIMIZATION ALGORITHMS (D=20, N=50)

Algorithm		f_1	f_2	f_3	f_4	f_5	f_6
QPSO	FV	2.0941E-289	**1.4351E001**	9.6164	**0.3631**	2.0428E-014	2.8404E-007
BC-QPSO	FV	2.34E-231	1.5205 E001	16.9143	0.756309	2.94E-005	9.77E-009
DIR-QPSO	FV	2.02E-278	1.49123 E001	**0**	0.672136	1.02E-013	3.49E-009
DCC-QPSO	FV	**1.53E-289**	1.7494 E001	**0**	0.858269	**2.66E-015**	**1.12E-009**

We can learn from Table that the dural-group QPSO algorithms exhibit better convergence performance when dealing with different multimodal functions. Compared with the other QPSO algorithms, the DCC-QPSO constructs two potential wells for master and slave group through using different correlation coefficients strategy. This policy offers the novel algorithm outstanding global searching ability due to the sharing of best positions between S1 and S2 through mutual complementation and evolution. The search pattern that two sub-groups carry out simultaneous searching and mutual learning enhances the probability of finding the optimal solution. It also offers higher convergence precision and speed within limited iterations.

ACKNOWLEDGEMENT

This research is jointly supported by the Science and Technology Support Program of Sichuan Province (Grant No. 2016RZ0065 and 2016RZ0053) and the Projects of the Education Department of Sichuan Province (Grant No. 15ZA0396 and 16ZB0212).

REFERENCES

[1] J. Kennedy and R. Eberhart. Particle swarm optimization. IEEE International Conference on Neural Networks,. 1995.

[2] J. Sun, Quantum-behaved particle swarm optimization algorithm, Jiangsu: Jiangsu University,2009.

[3] L. Zhang, Y. Tang, C. Hua, and X. Guan, A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques, Applied Soft Computing, vol 28, March 2015, Pages 138–149

[4] M. Clerc and J. Kennedy. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation, vol 6(1), 2002, pp.58-73.

[5] T. Wu, X. Chen, Y. Yan, Study of the Binary Correlation Quantum-behaved PSO Algorithm, Journal of Sichuan University(Engineering Science Edition), vol 46(4), 2014, pp. 103-110.

[6] T. Wu, Y. Yan, and X. Chen, Improved dual-group interaction QPSO algorithm based on random evaluation, Control and Decision,vol 3, 2015, pp.526-530.

Legalized Path Planning of Mobile Real Time Location System in Emergency Department

Shachi Singh [1], **Andrew R. Winton**[2]
[1]Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada
[2]Department of Computer Science, University of Manitoba, Winnipeg, MB, Canada

Abstract - *While developing our initial mobile real-time location system, several shortcomings were identified, most notably, lack of legal paths on which our mobile RFID readers could travel. This issue leads to difficulty in the practical application of the system. In order to resolve this shortcoming, a grid system and an A* search algorithm was used to identify obstacles, and develop legal movement paths for the readers to follow. With these paths identified, installation of a track system on which the readers will travel can be accomplished.*

Keywords: Path planning; obstacle avoidance; mobile RFID.

1 Introduction

Medical data collection can benefit from a real time system that can record vital patient and medical asset information automatically. An optimized path planning system for RFID based system (mRTLS), proposed in our earlier work, can effectively track with minimal errors but does not consider obstacles in the path [1].

2 Global Path Finding

A 3-D Agent Based Model designed to efficiently track patients with fewer readers optimized the paths travelled using a genetic algorithm to identify areas where patients tended to frequent and a simulated annealing algorithm to find the best order in which to connect these areas shown in Fig 2.

In order to practically implement the improved global path, obstacles such as walls and inaccessible areas in EDs need to be taken into account.

3 Local Path Finding

A uniform grid is created for the ED simulation layout that consists of both blocked white cells and unblocked cells shown in Fig 2. Each time A* algorithm [2] is implemented on the grid between first two points of the global route to get the local path. Combining these paths results in a legalized global path on which the mobile readers can move.

4 High Level View and Simulation

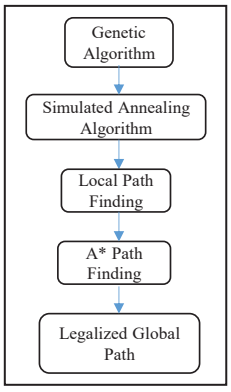


Figure 1. Optimized and legalized path finding process

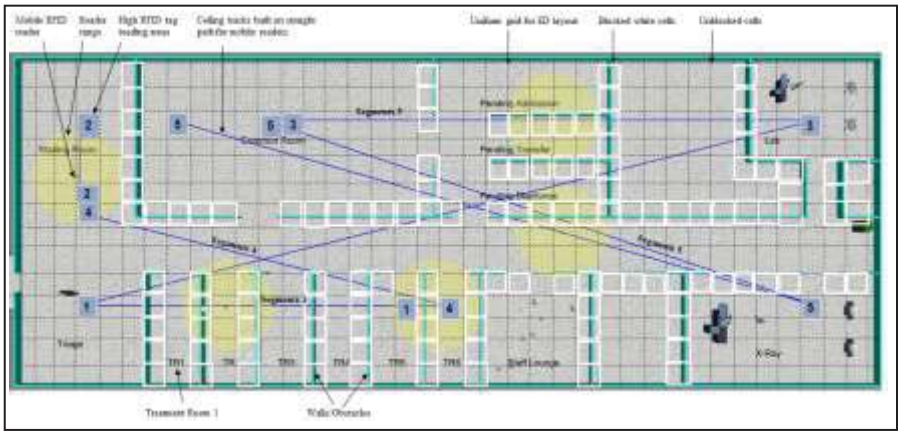


Figure 2. Straight line paths prior to obstacle avoidance.

5 Acknowledgements

We are sincerely grateful to Dr. McLeod and Dr. Friesen for their immense support throughout the research.

6 References

[1] S. Singh, M. Friesen, and R. McLeod, "Optimized Path Planning for a Mobile Real Time Location System (mRTLS) in an Emergency DepartmentMulti-objective optimization in path planning of mobile RFID surveillance system," *worldcomp-proceedings.com*.

[2] Amit, "Introduction to A*." [Online]. Available: http://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html. [Accessed: 12-Apr-2016].

SESSION

INTELLIGENT LINGUISTIC TECHNOLOGIES, ILINTEC'16

Chair(s)

Dr. Elena B. Kozerenko

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Plausible Expectations-Based Inference for Semantic Analysis

Igor Boguslavsky[1,2], Vyacheslav Dikonov[1], Tatiana Frolova[1], Leonid Iomdin[1], Alexander Lazurski[1], Ivan Rygaev[1], Svetlana Timoshenko[1]

[1]Laboratory of Computational Linguistics, A.A.Kharkevich Institute for Information Transmission Problems, Russian Academy of Sciences, Moscow, Russia

[2] Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid, Madrid, Spain

Abstract - *We present a project aiming at developing a semantic analyzer based on linguistic and world knowledge. The major sources of knowledge are a semantic dictionary, an ontology, a fact base, and a set of common sense axioms. We show the types of information stored in these resources, and demonstrate how they interact. As a case study, we take a simple but typical dialogue type, in which one of the interlocutors makes a proposal and the other one gives an indirect answer. The task of the analyzer is to interpret the answer as either the acceptance of the proposal or its rejection and, most importantly, to substantiate this interpretation. We show which knowledge is used and what reasoning should be performed in order to understand an indirect answer.*

Keywords: semantic analysis, inference, plausible expectation, indirect speech act

The authors express their gratitude to the Russian Science Foundation, who supported this work (grant No. 16-18-10422).

1. Introduction

The semantic analyzer SemETAP, under development in the Computational Linguistics lab of the Kharkevich Institute for Information Transmission Problems of the Russian Academy of Sciences, is aiming at performing semantic analysis based on both linguistic and extra-linguistic knowledge. This analyzer includes a powerful wide-coverage linguistic processor capable of building coherent semantic structures, a knowledge-extensive lexicon, which contains a variety of types of lexical information, an ontology, which describes general and domain-specific objects and their properties, a repository of ground-level facts, a set of common-sense axioms, and an inference engine [1]-[5].

This paper will demonstrate how this analyzer can help interpret indirect speech acts. Scholars have repeatedly addressed indirect speech acts in the context of pragmatics. The focus of attention has mainly been on the so-called conventional indirect speech acts which center on conventions of language. A typical example of these are questions used as requests (as in *Could you give me a ride?*). As for non-conventional indirect speech acts, which rely heavily on the context and common-sense knowledge, researchers mostly restrict themselves to citing examples, as in [6]:

(1) *Necesito los apuntes de la clase y tú eres el único estudiante que conozco*
'I need the class notes and you are the only student I know'

Here the addressee is expected to understand that the speaker is asking the interlocutor to lend him the interlocutor's class notes, although this is not what was uttered literally. "In indirect speech acts the speaker communicates to the hearer more than he actually says by way of relying on their mutually shared background information, both linguistic and nonlinguistic, together with the general powers of rationality and inference on the part of the hearer" [7, p.61].

It is difficult to model understanding of such speech acts, since they tend to be "open ended, both in terms of propositional content and linguistic form as well as pragmatic force" [8, p.42]. The most direct, if not the only one, way toward the interpretation of such speech acts seems the construction of a model of knowledge shared by the speaker and the hearer supplemented by a mechanism of common sense inferences. Exactly such an endeavor is undertaken in this paper. We will analyze an example of a dialogue in which one participant proposes to the other participant a joint activity and the latter gives an indirect answer (Section 4). Prior to that, we will discuss the inferences that we will be making (Section 2) and consider the knowledge resources we have at our disposal (Section 3).

2. Natural Language Understanding and Implicit Meaning

It is well known that not all information that we extract from text is explicitly conveyed by linguistic means. We will distinguish between (a) the literal content of a text, i.e. the content that can be extracted on the basis of language knowledge (such as the knowledge of morphology, syntax, semantics contained in lexicographic definitions of words, etc.) and (b) its expanded content, which includes the data that can be obtained through interpreting the text in the respective

context and using our knowledge of the world and the communicative situation.

On the other hand, the extent of reliability of information extracted from both the text and the context may vary. The logical science teaches us to make absolutely reliable conclusions. If an assertion that we believe to be true can be refuted by just one counter-example, this assertion should be regarded as false. As far as natural language is concerned, ontologies offer us a vast source of trustworthy inferences. A typical ontology-based inference may look as follows: If Mary is a little girl then she is human and belongs to the subclass of female beings and to the subclass of non-adult beings. Knowledge of lexical meanings of words is another source of logically correct inferences. If it is true for example that John *persuaded* Mary to marry him then it could be concluded that Mary agreed to marry John. If it is known that Bill *pretends* to be ill then it may be concluded that Bill is in good health. These conclusions may be automated if we have analytical definitions of words written in a formal metalanguage to which inference rules may be applied.

It is known however that, in their everyday verbal and thinking activities, people often resort to inferences that are logically far from being perfect. Yet these inferences help us understand coherent texts, including dialogues. Any coherent text contains much implicit information needed for its proper understanding. If this information is not extracted, a coherent text may look as a collection of disconnected sentences gathered together for some enigmatic reason. In most cases, the hearers easily restore implicit information from texts, using plausible expectations rather than logical inferences. Consider the following example.

(2) *The child had been prescribed an antipyretic but the pharmacy was closed.*

Based exclusively on linguistic knowledge contained in the first clause, we find out that the child is sick (medications are prescribed to sick people). From the second clause, we learn that some commercial enterprise (pharmacy) is closed. The conjunction *but* standing between the first and the second clause points to the fact that there is a discrepancy between the two clauses but nothing tells us what it consists of. As long as the hearer fails to understand why the second clause violates the expectations emerging from the first clause, he will consider the whole sentence as anomalous, in just the same way as the sentence

(2a) *The child had been prescribed antipyretics but the library was closed.*

Everything fits together perfectly if we assume that there was no antipyretic medication prescribed to the child at home so it had to be bought at the pharmacy. Anyone who happens to read (2) will understand that there was no relevant medication – but this conclusion does not follow from anything! It is only made on the basis of the fact that it allows one to complete the picture, so that, due to this assumption, the text becomes coherent and sensible.

In example (2) the missing conclusion did not follow from lexical meanings. It was made due to the fact that the hearer was aware of curing routines. Meanwhile, lexical meanings themselves often prompt the hearer for the situation to be expected. So, it does not follow from sentence

(3) *John decided to quit his job*

that John implemented his decision. Yet, (3) activates this expectation, which could either be confirmed by subsequent text (say, *John decided to quit his job and went travelling*) or be disproved and dismissed. Expectations of this kind reflect the hearer's readiness for a certain turn of events. In Section 4, we will elaborate on several such inferences.

3. Knowledge Resources

The semantic analyzer, SemETAP, is constructed as a component of the multifunctional linguistic processor ETAP, which was created by the team of researchers that includes the authors of this paper (see e.g. [9]). It has a variety of options, including a rule-based machine translation system operating between Russian and English, and is supported by a number of advanced linguistic resources – dictionaries, parsers, and a Russian treebank SynTagRus, fully annotated with dependency trees [10].

The semantic analyzer has several sources of knowledge. Linguistic data is contained in ETAP's databases, the most important of which is the Combinatorial dictionary supplied with ample lexicographic information, including lexical functions [11], [12]. World knowledge is presented by several resources – the Ontology, the Repository of individual entitites, and a set of axioms for inferences. The first resource contains the data on classes of things and situations (such as Human, Artifact, CommunicativeEvent, Buying, etc.), whilst the second one contains the data on individual, singular objects or events (like Moscow, France, World War II).

The most important resources for the purposes of this paper are the Combinatorial dictionary, the Ontology and the set of axioms. We will focus on the dictionary and the ontology in this section putting an emphasis on their use in making inferences based on plausible expectations. Axioms will be illustrated in Section 4.

3.1. The Combinatorial Dictionary

Every lexical entry in the combinatorial dictionary is supplied with information of various types: syntactic and semantic features, subcategorization frames, references to the ontology, lexical functions, as introduced in the Meaning ↔ Text Theory of Igor Mel'čuk [13], translational equivalents for a number of languages,

including the UNL interlingua [14], [15], [16]. Importantly, the dictionary entry contains a semantic decomposition of the word produced using the concepts of the ontology. This part of the entry lists inferences that can be made based on the sense of this word, stating the conditions for these inferences to be valid. Logically strict inferences are listed in the section called Implications and plausible expectations are listed in the Expectations section.

The following example that presents four sections of the dictionary entry for the Russian verb *pomogat'* 'to help' in its main lexical sense ('help someone do something') illustrates this material in more detail.

To facilitate understanding, the semantic decomposition zone, as well as the implications and the expectations zones, will be presented in a (quasi)natural language rather than in the formal language designed for this purpose.

Examples: *Kolja pomogaet Mashe izuchat' kitajskij jazyk* 'Nick helps Mary to learn Chinese'. *Kolja pomog Mashe reshit' zadachu* 'Nick has helped Mary to solve the problem.' *On pomog ej sovetom (tem, chto pogovoril s dekanom)* 'He helped her by giving a piece of advice (by talking to the dean)'.

Decomposition: *HELPER helps HELPEE to reach GOAL by doing AID*[1] = 'Human HELPEE has goal GOAL; Human HELPER does action AID; as a result, it is easy for HELPEE to reach GOAL; the fact that HELPER does AID is good for HELPEE'

Implication: If *pomogat'* stands in the past tense and the perfect aspect, then it is implied: HELPEE has reached GOAL. Namely, if *Kolja pomog Mashe reshit' zadachu* 'Nick helped Mary to solve the problem' then *Masha reshila zadachu* 'Mary solved the problem')

Expectation: If *pomogat'* stands (i) in the nonpast tense **or** (ii) in the past tense and the imperfect aspect, then it is expected: HELPEE will reach GOAL. Specifically, if *Kolja pomogaet (pomogal, pomozhet) Mashe reshit' zadachu* 'Nick helps, was helping, will help Mary to solve the problem' then it could be expected that *Masha reshit zadachu* 'Mary will solve the problem').

Obviously, the same is true, *mutatis mutandis,* for the English verb *help,* maybe with the exception of certain subtleties associated with differences between the Russian and the English system of verbal tenses and aspects.

3.2. The Ontology

Let us now move on to the ontology. By way of illustration, we will present here ontological descriptions of three classes connected with each other: we will need them in Section 4.

Class 1 is Organization. This class belongs to the class Agent and, simultaneously, to the class Group. Every Organization has a Chief from the class Human, Staff, also from the class Human, and a Function – an activity that Organization is supposed to carry out.

Class 2 is ClientServingOrganization, which is a subclass of Organization. Class 2 inherits all properties of its superclass Organization (which we do not repeat here, see above) and has certain properties of its own. This class comprises such organizations as movie theater, bakery, hospital, library, public bath, school etc. but does not include such organizations as ministry or city council. The Function of ClientServingOrganizations is to do certain Activity in the interests of the Client, or to ensure that the Client does certain Activity in the Client's own interests. So, the clients eat at restaurants, buy bread in bakeries and receive medical treatment in hospitals. This information is stored in the ontological description of ClientServingOrganizations under hasClientAction slot.

Finally, class 3 is SeafoodRestaurant, belonging to the class ClientServingOrganization. Its function is to prepare and sell to Clients food made from sea species, while the Clients' Activity is to eat this food, being inside the restaurant.

Formally, Ontology entries look as follows:

```
Organization(X)
    isA(X,Group)
    isA(X,Agent)
    hasChief(X,Human)
    hasInStaff(X,Human)
    hasFunction(X,Action)

ClientServingOrganization(X)
    isA(X,Organization)
    hasUser(X,OR (Human, Organization))
    hasUserAction(X,Action)

SeafoodRestaurant
    isA(X,ClientServingOrganization)
    isA(X,Place)
    hasUser(X,Human-1)
    hasFunction(X,Preparing&Selling)
    hasAgent(Preparing&Selling,X)
    hasObject(Preparing&Selling,Seafood-1)
    hasAddressee(Preparing&Selling,Human-1)
    hasUserAction(X,Eating-1)
    has Agent(Eating-1,Human-1)
    hasObject(Eating-1,Seafood-1)
    hasLocation(Eating-1,X)
```

[1] In FrameNet, the respective semantic role in the frame of ASSISTANCE is called Focal_Entity, which we find a bit too vague.

4. Case Study: an Indirect Answer to a Proposal

We will now consider in detail a specific class of dialogues in which one of the interlocutors suggests that the other interlocutor do something and the latter responds to this suggestion. In the simplest case, the response may be a direct one:

(4a) Speaker A: Will you go to the movies with me?

(4b) Speaker B: Yes, thank you / No, thank you.

Often, however, especially in the case of a negative response, people answer in an indirect way. For instance, the following answers could be expected in response to (4a):

(4c) I have no time to spare.

(4d) I have a lot of things to do.

(4e) I need to prepare for an exam.

(4f) I have no money.

(4g) Today is my father's birthday.

(4e) I don't like movies.

(4f) I have a headache.

(4g) My parents do not permit me to go to the movies with strangers.

We easily interpret all such answers as refusals, but why? How do we know that the interlocutor has not accepted the proposal? The literal meaning of sentences (4c-f) may have nothing to do with A's question, but, by force of Grice's Maxim of Relevance we must proceed from the assumption that each of these sentences indirectly contains the "yes" or the "no" answer. Our goal, then, is to extract this implicit information from any question-answer pair. An important thing is the fact that inferences made in the course of reasoning, are no logical conclusions in the strict sense of the term but are plausible expectations, which we described above. We will demonstrate the way to achieve the goal of extracting the relevant information using the following dialogue (5a-5b) as an example:

(5a) Speaker A: *Let's go to a seafood restaurant!*

(5b) Speaker B: *My doctor has forbidden me to eat fish.*

We will begin with (5a). Let us assume that the proposal contained therein is accepted and try to draw inferences from this assumption. We will show that these inferences are incompatible with the inferences made from (5b). Accepting the proposal implies that (6) will be true.

(6) A and B go to a seafood restaurant.

If it is to be taken literally, (6) reports on a movement which has a seafood restaurant as its destination point. The proposition

(7) 'X goes from P1 to P2'

allows for a number of conclusions. Three events are important here: the action of motion, the end of being located in P1 and the start of being located in P2. These events have different epistemic statuses. While we can derive from (7) three following inferences with absolute certainty:

(7a) at moment t0 X is moving,

(7b) at moment t1<t0 X was located in P1,

(7c) at moment t0 X is not located in P1,

we cannot reliably assert anything about the future location of X in P2, simply because we cannot be sure about any future events. After the start of motion, circumstances may emerge which could prevent X from reaching the destination point. For instance, the object may feel sick and return back home. Yet, in absence of information pointing to unforeseen circumstances it is natural to expect that the destination point will be reached. This is what the circumscription principle by McCarthy [17] suggests: things are as expected unless otherwise specified. This assumption can be formulated as Axiom 1[2].

Axiom1: if Object moves to Place at t0, then it CanBeExpected that Object will be located at Place at t1>t0.

The next step is as follows. What conclusion could be derived from the fact that the goal of motion, the destination point, is reached, and the object is in the restaurant? We remember that, as stated above, the restaurant belongs to the class of Organizations that has users (clients). A relevant property of ClientServingOrganizations is that they are associated with a particular expectation, namely that if an Object is located in such an Organization, it is natural to expect that this Object will be using this Organization according to its intended purpose, i.e. fulfill the role of a client.

The expectation consisting in the fact that a Human who is in a ClientServingOrganization will use it according to its intended purpose belongs to the same class of expectations as the expectation concerning the use of an Instrument. If a text activates the fact that the Subject has an Instrument at his disposal, we should expect that he will be using it according to its intended purpose. If we

[2] For the reader's convenience, the axioms are written in a (quasi)natural language. In the semantic analyzer they are naturally presented in the formal language.

come across a sentence saying that *Father took an axe and went to the woods* than we feel prepared to the fact that father is going to chop wood. Moreover, often the very fact that there is a tool ready presents itself as a nomination of manipulation with this tool. If someone says that he has not taken a violin in his hands for a long time it is to be implied that he hasn't played it for a long time.

The conclusion that a human present in a restaurant is its client is not the only possible one. Another natural assumption could be that this human belongs to the restaurant's staff and is there on duty. Since we are only concerned with plausible expectations, we need not list all feasible situations and can confine ourselves with the expectations most naturally activated in a given situation. Hence, Axiom 2 can be considered to be legitimate:

Axiom 2. If a Human is located in ClientServingOrganization, then it CanBeExpected that Human either is a Client or belongs to Staff.

Another example of using Axiom 2 could be seen in the dialogue (8a-8b):

(8a) Speaker A: *Where is your wife?*

(8b) Speaker B: *She is on the beach.*

A beach is not just a plot of land where a human can be as on any other plot. It is a plot of land intended to be used by humans who swim in the sea bordering on this beach or bathe in the sun. In a way, the beach is, technically, similar to a ClientServingOrganization in that it also has a function and clients.

The next logical step consists in the assumption that if a Human is in a ClientServingOrganization as a client then it is not difficult to predict what he is going to do there if we know the specialization of this Organization. Hence, Axiom 3 can be proposed:

Axiom 3. If a Human is Client at ClientServingOrganization for which its ClientAction is Action, then it CanBeExpected that Human performs Action.

Axiom 3 says that if someone is a Client of a ClientServingOrganization, then he does what clients of such organizations normally do. We saw in Section 3 above that the ontological description of a restaurant includes the information that the actions of its clients consist in consuming the food prepared in this restaurant – in much the same way as the actions of clients of a library consist in reading the books borrowed there.

Axiom 3 can be used even when the speech act to be analyzed is not indirect. For example, dialogue (9a-9b)

(9a) Speaker A: *Where is John?*

(9b) Speaker B: *He is in the hospital*

gives a direct and relevant answer to the posited question. The first sentence asks about John's location and the second sentence specifies a concrete place – hospital. However, the content of the answer is not exhausted by stating the place. The hearer has all reasons to assume that John is receiving medical treatment, and this is an important part of information he has received, which may trigger further units of a conversation (such as *What happened to him?*).

In a similar way, the reply in dialogue (10a-10b)

(10a) *What did you do yesterday?*

(10b) *I went to the theater.*

gives a direct answer: the interlocutor moved to the theater. However, the real pragmatically relevant reply is not the action of movement, but the action determined by Axiom 3 and the ontological description of a theatre: I watched a performance.

Thus, we see that both direct and indirect speech acts use the same inference mechanisms.

There is one more gap between (5a) and (5b) that has to be bridged: (5a) is related to seafood, while (5b) mentions fish, which is not exactly the same. Seafood denotes a larger class that includes two subclasses - fish and shellfish. This is what can be inferred directly from the ontology: if A belongs to the class of Seafood, it belongs either to the class of Fish or Shellfish. What we need now is a general axiom covering disjunction:

Axiom 4. If P is true of (A or B), then CanBeExpected that P is true of A.

For example, if we know that Peter or Bill will come, we can expect that Peter will.

Now, we are prepared to build the following chain of inferences from (6):

(11) Speaker B goes to a seafood restaurant ➔ Speaker B is in a seafood restaurant (by Axiom 1) ➔ Speaker B is Client of a seafood restaurant (by Axiom 2) ➔ Speaker B eats seafood there (by Axiom 3) ➔ Speaker B eats fish there (by Axiom 4).

Let us now return to the reply (5b) and see what inferences it invites.

(5b) Speaker B: *My doctor has forbidden me to eat fish*

Here, we enter the area of modalities, which provide ample space for versatile inferences, many of which are not one hundred percent reliable – especially if deontic modality is concerned.

Let us first of all turn our attention to the fact that expectations generated by modal predicates may vary in their degree. If an order was issued to do some P, then the probability of P taking place is likely higher than if it was a request or a piece of advice. However, at the present stage of our study we abstract away from such differences and only distinguish between two types of inferences: rigid (= 100-percent true conclusions) and soft ones (plausible expectations).

In the area of deontic necessity it is a valid deliberation that if X needs to do P then it can be expected that he will do P. Naturally, if P is lack of action Q (such as expressed by the sentence like *He must not go*) then it is to be expected that Q will not be done. Hence, Axiom 5 is appropriate:

Axiom 5: if Human must perform Action, then CanBeExpected that Human performs Action.

A complication may arise that if the idea of necessity is within the scope of another operator, such as a predicate of communication or opinion, then the imperativeness is suppressed or, in any case, drastically reduced. If A believes or says that B must do P, this does not imply that B really must do P. If however the predicate denotes prohibition or order then the imperativeness is likely not canceled or relaxed, since, in accordance with the semantics of these predicates the Agent has the right to postulate the necessity.

Let us address again the unit (5b). The point of departure here is the dictionary definition for *prohibit*:

(12) *X prohibits Y to do Z* = 'X says to Y that Y must not do Z, X having right to do so'.

In agreement with what has been said above, the following Axiom 6 may be proposed:

Axiom 6: if Human-1 prohibits that Human-2 performs Action, then it CanBeExpected that Human-2 does not perform Action.

When applied to (5a), this axiom brings us to the proposition

(13) 'Speaker B does not eat fish'.

Thus, given the inference chain (11) and (13), we see that propositions (6) and (5b) give rise to contradictory inferences. Hence the hypothesis that the proposal (5a) was accepted should be rejected.

5. Conclusion

Semantic analysis of a coherent text, in particular, a dialogue, requires extraction of implicit information. This task becomes especially relevant when we are confronted with indirect speech acts, so that information conveyed by the text is essentially different from what is literally said. We endeavor to show how understanding indirect speech acts can be automated, using an example of dialogue containing an indirect answer to a question. We have shown that inferences made on the basis of plausible expectations play a key role in this process. Such a model can be implemented within the framework of a linguistic processor that has access to linguistic and word knowledge and is able to make inferences.

References

[1] I.M. Boguslavsky, L.L. Iomdin, V.G. Sizov, S.P. Timoshenko. "Interfacing the Lexicon and the Ontology in a Semantic Analyzer"; COLING 2010. Proceedings of the 6th Workshop on Ontologies and Lexical Resources (Ontolex 2010), Beijing, pages 67–76, 2010.

[2] I.M.Boguslavsky. "Semantic Analysis Based on Linguistic and Ontological Resources"; Proceedings of the 5th International Conference on the Meaning - Text Theory. Barcelona. Igor Boguslavsky and Leo Wanner (Eds.). P. 25-36, 2011.

[3] I.M.Boguslavsky, V.G.Dikonov, S.P.Timoshenko. "An ontology for the support of tasks of extracting meaning from natural language text"; Information technogies and systems. Moscow, 2012. (In Russian).

[4] I.M.Boguslavsky, V.G.Dikonov, L.L.Iomdin, S.P. Timoshenko. "Semantic representation for NL understanding"; Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue". Issue 12(19), Moscow, RGGU Publishers. P. 132-144, 2013.

[5] I.M.Boguslavsky, V.G.Dikonov, L.L.Iomdin, A.V.Lazursky, V.G.Sizov, S.P.Timoshenko. "Semantic Analysis and Question Answering: a System Under Development"; Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue" Issue 14(21). Moscow, RGGU Publishers. P.62-79, 2015.

[6] J. Cesar Félix-Brasdefer. "Indirectness and politeness in Mexican Requests"; D. Eddington (ed.), Selected Proceedings of the 7th Hispanic Linguistic Symposium . Somerville, MA: Cascadilla Press. P. 66-78, 2005. (Paper available on-line at http://www.lingref.com/cpp/hls/7/)

[7] John Searle. "Indirect speech acts. Syntax and semantics", ed. by Peter Cole and Jerry L. Morgan. New York: NY: Academic Press. 59-82; 1975.

[8] Shoshana Blum-Kulka, Juliane House, and Gabriele Kasper (eds.). "Cross-cultural pragmatics: Requests and apologies". Norwood: Ablex Publishing. 1989.

[9] Ju. D.Apresjan, I.M. Boguslavsky, L. L.Iomdin, A.V. Lazursky, V. Z.Sannikov, V.G. Sizov,

L. L.Tsinman. "ETAP-3 Linguistic Processor: a Full-Fledged NLP Implementation of the MTT"; First International Conference on Meaning-Text Theory (MTT'2003). Paris: Ecole Normale Superieur. P. 279-288, 2003.

[10] P.V. Dyachenko, L.L. Iomdin, A.V. Lazursky, L.G. Mityushin, O.Ju.Podlesskaya, V.G. Sizov, T.I. Frolova, L.L. Tsinmann. "The current state of deeply annotated corpus of Russian texts (SynTagRus)"; Russian National Corpus. 10 years of the project. Proceedings of the V.V. Vingoradov Institute of Russian Language. Moscow, Vol. 6. P. 272-299, 2015. (In Russian)

[11] I.M. Boguslavsky, L.L. Iomdin, V.G. Sizov. "Multilinguality in ETAP-3. Reuse of Linguistic Resources"; Proceedings of the Workshop "Multilingual Linguistic Resources". 20th International Conference on Computational Linguistics. Geneva P. 7-14, 2004.

[12] Ju.D.Apresjan, I.M.Boguslavsky, L.L.Iomdin, L.L.Tsinman. Lexical Functions in Actual NLP-Applications. "Selected Lexical and Grammatical Issues in the Meaning–Text Theory"; In honour of Igor Mel'čuk / Leo Wanner (ed.). Studies in Language Companion. Series 84. Amsterdam: Benjamins Academic Publishers. P. 199-230, 2007.

[13] I.A. Mel'čuk. "The Theory of Lingiustic Models of the type "Meaning ↔ Text"". Moscow, 1974 (2nd ed., 1999). (In Russian).

[14] H. Uchida, M. Zhu, and T. Della Senta. "The Universal Networking Language", 2nd ed. UNDL Foundation, 2005.

[15] I.Boguslavsky, J. Cardeñosa, C.Gallardo, L.Iraola. "The UNL Initiative: An Overview"; CICLing 2005. Lecture notes in computer science. A. Gelbukh (ed.). Berlin-Heidelberg: Springer-Verlag, Vol. 3406. P. 377 – 387, 2005.

[16] V.G. Dikonov, I.M. Boguslavsky I.M. "Universal Dictionary of Concepts"; MONDILEX First Open Workshop "Lexicographic Tools and Techniques". Moscow. P. 31-41, 2008.

[17] John McCarthy. "Circumscription – a form of non-monotonic reasoning"; Artifical Intelligence 13:27-39, 1980.

ISBN: 1-60132-438-3, CSREA Press ©

SESSION

LATE BREAKING PAPERS - OPTIMIZATION METHODS AND MODELS, LEARNING TECHNIQUES, FUZZY LOGIC, AND APPLICATIONS

Chair(s)

TBA

ISBN: 1-60132-438-3, CSREA Press ©

ISBN: 1-60132-438-3, CSREA Press ©

Solving 0-1 Multi-Dimensional Knapsack Problem using a Discrete Binary Version of Grey Wolf Optimizer Algorithm

Srivathsan Lakshminaryanan [1], **Devinder Kaur** [2]
[1] EECS, University of Toledo, Toledo, OH,USA
[2] EECS, University of Toledo, Toledo, OH,USA

Abstract - *Grey Wolf Optimizer Algorithm (GWO) is a recently proposed meta-heuristic algorithm based on the social hierarchy and hunting behavior of grey wolves. The GWO algorithm is used to solve the continuous optimization problems. In this paper, a discrete binary version of the GWO algorithm called Discrete Binary Grey Wolf Optimizer (DBGWO) is proposed to solve the NP-Hard 0-1 Multi-Dimensional Knapsack Problem (MKP). The proposed DBGWO method is tested for a set of seven MKP benchmark problems and the results are compared with three recent methods available in the literature for solving the MKP namely Multi Hybrid Particle Swarm Optimization (MHPSO) algorithm, Egyptian Vulture Optimization(EVO) Algorithm and binary Artificial Fish Swarm Algorithm (b-AFSA). The experimental results show that the proposed DBGWO method outperforms the MHPSO, EVO and b-AFSA algorithms.*

Keywords: Discrete Binary Grey Wolf Optimizer; 0-1 Multi-Dimensional Knap Sack problem, MKP, swarm intelligence, Grey Wolf Optimizer

1 Introduction

The Multi-Dimensional Knapsack Problem (MKP) is a combinatorial NP-Hard problem that can be applied to many fields, such as capital budgeting, cutting stock problem and allocating processors in distributed systems [1]. In the past, exact methods, such as dynamic programming and branch and bound algorithms were proposed to solve MKP [2] [3]. The solution space of MKP grows exponentially with the problem size. For this reason, the exact methods are not practical for large size MKP. Therefore several meta-heuristic algorithms were proposed to address the Multi-Dimensional Knapsack Problem [4-9].

In reference [1], Hong Li, Yong-Chang Jiao, Li Zhang, and Ze-Wei Gu used genetic algorithm based on the orthogonal design to solve the MKP. Moreover, a check and repair operator was used to modify the infeasible chromosomes generated in each generation to feasible chromosome. Md. Abul Kalam Azad, Ana Maria A.C. Rocha1 and Edite M.G.P. Fernandes proposed binary-Artificial Fish Swarm Algorithm (b-AFSA) with a decoding algorithm to repair the infeasible solutions for solving MKP [10].

Said Labed, Amira Gherboudj and Salim Chikhi used a Modified Hybrid Particle Swarm Optimization Algorithm for solving the Multidimensional Knapsack Problem. In their method they have combined the cross over operation used in genetic algorithm with the Binary Particle Swarm Optimization Algorithm (PSO) [11]. In reference [12], Chiranjib Sur, Sanjeev Sharma, and Anupam Shukla proposed Egyptian Vulture Optimization (EVO) Algorithm based on the food acquiring behavior of Egyptian Vultures to solve the MKP.

In the year 2014, Seyedali Mirjalili, Seyed Mohammad Mirjalili and Andrew Lewis proposed swarm-intelligence based algorithm called Grey Wolf Optimizer(GWO) mimicking the social hierarchy and hunting behavior of grey wolves to solve optimization problems in continuous domain [13]. Since then, the GWO algorithm has been used to address optimization problems in various engineering fields [14]-[19].

In this paper, we propose a discrete binary version of the GWO algorithm called Discrete Binary Grey Wolf Optimizer (DBGWO) to solve the 0-1 Multi-Dimensional Knap-Sack problem. The proposed method was tested on seven benchmark Multi-Dimensional Knapsack problems.

This paper is organized as follows: Section II presents the mathematical formulation of Multi-Dimensional Knapsack problem. Section-III presents an overview of GWO algorithm. Section V describes the mathematical dynamics of the proposed DBGWO algorithm and encoding of DBGWO to solve the MKP problem. Section VI presents the results obtained using the DBGWO on the seven benchmark MKP datasets. The performance of the DBGWO algorithm is then compared with previous research work which addressed the same MKP datasets using other meta-heuristic algorithms viz., EVO, MHPSO and b-AFSA. Section VI presents the conclusion.

2 Multi-Dimensional KnapSack Problem Formulation

In 0-1MKP, it is assumed that we have a knapsack of dimension m with each dimension having a different capacity Cj (j=1,…, m) and a set of n objects. Each object has a profit Pi (i=1, …, n) and different weight Wij in each of the dimension j of the knapsack.

The objective is to find a subset of objects that maximize the profit without exceeding the capacity in any of the dimensions of the knapsack. This is explained by following equations:

$$\text{Maximize:} \quad \sum_{i=1}^{n} P_i * X_i \tag{1}$$

$$\text{Subject to:} \quad \sum_{i=1}^{n} W_{ij} X_i \le C_j \tag{2}$$

where j=1, ..., m and

$$\text{Xi} = \begin{cases} 1, \text{ if the object i is selected} \\ 0, \text{ otherwise} \end{cases} \tag{3}$$

where i=1, ...,n.

Since, Xi can be either 1 or 0 at a time, there are 2^n possible solutions for a set of n objects. For example, there are 2^{50} possible solutions for a Multi-Dimensional Knapsack Problem with 50 objects.

3 Overview of GWO Algorithm

3.1 Social Hierarchy and Group Hunting Behavior of Grey Wolves

GWO algorithm is a recent meta-heuristic algorithm based on the social hierarchy and group hunting behavior of grey wolves. Grey wolves maintain a strict social hierarchy in the pack to maintain stability and to mutually assist each other in hunting. Fig. 1 shows the social hierarchy of grey wolf pack.

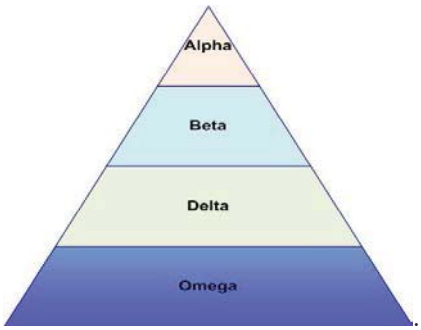


Fig. 1. Social Hierarchy of grey wolves [13]

Alpha (α) is the leader of the pack. The alpha can be either male or female. Alpha wolf is chosen based on the strength and the ability to fight. The entire pack obeys the decision dictated by the alpha. Beta (β) occupies 2nd level in the hierarchy. Beta serves as the advisor to the alpha in decision making. It also commands the lower-level wolves and maintains the discipline over the pack. Delta stands 3rd in the grey wolf social hierarchy. They obey the orders of alpha and beta wolves. Deltas are the aged wolves that watch the pack's territory boundary and warn the pack in case of any danger. They also play the role of caretakers for the ill and wounded wolves. Omega wolves are at the bottom of the social hierarchy. Omegas submit to the commands of all the other dominant wolves and are allowed to eat after everyone else has eaten. Grey wolves hunt in a group, coordinating with each other, with the alpha leading the way.

Hunting of prey takes place in multiple steps. First the pack locates the herd of prey and then surrounds them. Wolves prefer for the prey that is weak/sick/old or injured [20]. Alpha, beta and delta locate the weakened prey through its body stance, uncoordinated movements or the smell of wounds and start to chase it. Omegas follow the dominant wolves.

After the pack nears the prey, it encircles the prey and harasses it until the prey stops moving. The pack then jumps and attacks the prey.

3.2 Mathematical Modeling of Social Hierarchy and Group Hunting:

This section describes the mathematical model of group hunting behavior of a pack of grey wolves. In order to mathematically model the social hierarchy, the fittest solution is assumed to be the alpha and it is represented as (Xα). The next fittest wolf becomes beta (Xβ). Similarly, the 3rd fittest wolf becomes delta (Xδ). The rest of the wolves are assumed to be omega.

Omegas update their locations in the search space based on their relative positions from alpha, beta and delta. Omegas may encircle the prey whose location it estimated based on the positions of alpha, beta and delta or it may diverge from the estimated location of the prey to determine a better prey. If omega finds a prey fitter than the prey surrounded by alpha, beta and delta, it becomes the alpha. The wolves at 2nd and 3rd position from the prey become beta and delta respectively. The remaining wolves become omega.

First, the distance of omegas from alpha, beta and delta are calculated using the following equations:

$$D_{\omega\alpha} = \left| C_{\omega\alpha} \cdot X_\alpha - X_\omega \right| \tag{4}$$

$$D_{\omega\beta} = \left| C_{\omega\beta} \cdot X_\beta - X_\omega \right| \tag{5}$$

$$D_{\omega\delta} = \left| C_{\omega\delta} \cdot X_\delta - X_\omega \right| \tag{6}$$

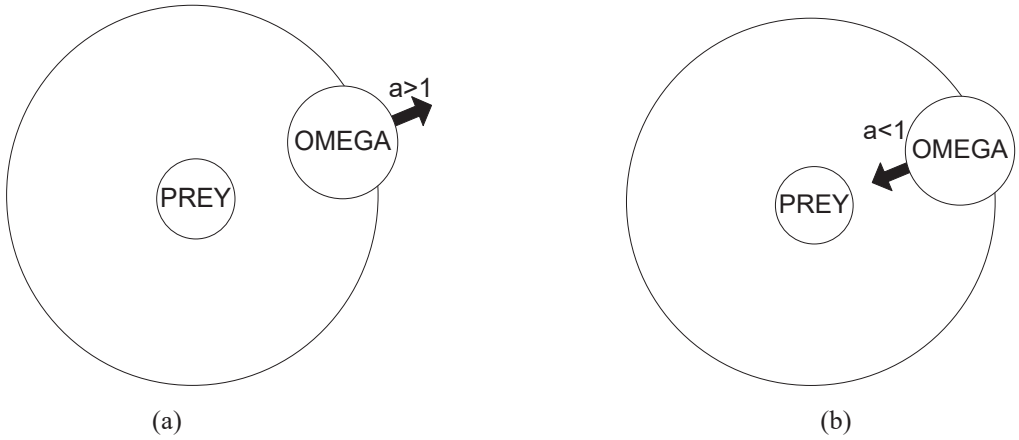
where

$D_{\omega\alpha}$ is the distance between omega (Xω) and alpha (Xα)

$D_{\omega\beta}$ is the distance between omega(Xω) and beta (Xβ)

$D_{\omega\delta}$ is the distance between omega(Xω) and delta (Xδ).

Due to the natural obstacles, the omega cannot find the exact locations of alpha beta and delta. There is always some



(a) (b)

Fig. 2. (a) Omega diverging away from a prey to find a fitter prey (b) Omega finding a suitable location to encircle the prey

error in omegas estimation of dominant wolf's location. This is modeled by obstacle-vector Cωα, Cωβ and Cωδ. Elements of Cωα, Cωβ and Cωδ are random numbers in the interval [0 2].

($C_{\omega\alpha} \cdot X_{\alpha}$) is the approximate location of alpha.

($C_{\omega\beta} \cdot X_{\beta}$) is approximate location of beta.

($C_{\omega\alpha} \cdot X_{\delta}$) is approximate location of delta.

After calculating the distance from dominant wolves, omega updates to or diverges away from the prey locations it calculates based on the positions of alpha, beta and delta using the following equations:

$$X_{\omega\beta} = X_{\beta} - A_{\omega\beta} \cdot (D_{\omega\beta}) \quad (7)$$

$$X_{\omega\alpha} = X_{\alpha} - A_{\omega\alpha} \cdot (D_{\omega\alpha}) \quad (8)$$

$$X_{\omega\delta} = X_{\delta} - A_{\omega\delta} \cdot (D_{\omega\delta}) \quad (9)$$

where

$X_{\omega\alpha}$ is the new location of omega based on alpha (Xα).

$X_{\omega\beta}$ is the new location of omega based on beta (Xβ).

$X_{\omega\delta}$ is the new location of omega based on delta (Xδ).

Aωα, Aωβ and Aωδ are exploration-encircle coefficient vectors. Random movement of omega around prey is modeled using Aωα by assuming that prey is close to alpha. Aωβ models the random movement of omega around the prey by assuming the prey is close to beta. Similarly, Aωδ models the random movement of omega around the prey by assuming the prey is close to delta.

Aωα, Aωβ and Aωδ are calculated using equations:

$$A_{\omega\delta} = 2 * a * rand[0,1] - a \quad (10)$$

$$A_{\omega\beta} = 2 * a * rand[0,1] - a \quad (11)$$

$$A_{\omega\alpha} = 2 * a * rand[0,1] - a \quad (12)$$

where rand[0,1] is a random number in the interval [0,1] and 'a' is called decision-variable.

'a' uniformly decreased from 2 to 0 during the course of iterations. In each iteration, each of $A_{\omega\alpha}$, $A_{\omega\beta}$ and $A_{\omega\beta}$ take a random value in the interval [-a, a]. For example, when a=2, each of $A_{\omega\alpha}$, $A_{\omega\beta}$ and $A_{\omega\beta}$ take a random value in the interval [-2,2] and when a=1, they take random values in the interval [-1, 1].

Decision variable 'a' is used to model the exploration and encircling behavior of wolves. 'a' decides whether the omega should encircle the prey or diverge away from the prey to find the fitter prey.

For first half of iterations, 'a' is greater than 1. During these iterations, omegas diverge from the prey location they have estimated based on the locations of dominant wolves to find a better prey. Fig. 2 (a) shows how omega diverges from a prey to find a better prey.

For the second half of iterations, 'a' is lesser than 1. The omegas encircle the prey in smart location they have estimated based on the positions of dominant wolves in to trap and attack it. Fig. 2(b) shows how an omega positions around a prey for a powerful attack.

During the last iteration 'a' becomes 0 and all elements of the randomization-coefficient vectors Awα, Awβ and Awδ become zero. The entire pack jumps and attacks the prey. The coordinates of all the wolves become the same as the coordinate of the fittest prey.

The value of 'a' in each iteration is calculated using the following equation:

$$a = 2 - \left(\frac{2 * t}{t_{total}}\right) \quad (13)$$

where

t is current iteration and t $_{total}$ is the total number of iterations

After calculating the new positions based on alpha, beta and delta separately, omegas update to the mean of the three new locations it estimated using the following equation:

$$X(t+1) = \frac{X_{\omega\alpha} + X_{\omega\beta} + X_{\omega\delta}}{3} \quad (14)$$

where $X_{\omega\alpha}$, $X_{\omega\beta}$ and $X_{\omega\delta}$ are new positions of omega alpha, beta and delta calculated using the equations (7), (8) and (9).

4 The proposed DBGWO Algorithm for 0-1 Multi-Dimensional Knapsack Problem

In this section, first the mathematical dynamics of the proposed Discrete Binary Grey Wolf Optimizer (DBGWO) algorithm is presented. Then, the encoding of DBGWO algorithm to MKP problem is described.

4.1 Mathematical Dynamics of the Proposed Discrete Binary Grey Wolf Optimizer (DBGWO) Algorithm

The solutions to continuous domain optimization problems are represented by real numbers. But, the solutions to discrete combinatorial problems are represented by discrete integer numbers. In case of MKP, the solutions must be represented as combination of 0's and 1's [21].

The original GWO algorithm was proposed to solve optimization problems in continuous search space. But MKP is a discrete binary combinatorial optimization problem. Hence, the solutions must be converted from real values to binary value. To do this, first each element x_i of the solution x is constrained in the interval [0, 1] using the sigmoid function as follows:

$$S(x_i) = \frac{1}{\left(1 + e^{-x_i}\right)} \quad (15)$$

To obtain the binary representation (x_i') of each element x_i, we generate a random number from the interval [0, 1] for each dimension i of the solution x and compare it with $S(x_i)$. If the random number is lower than $s(x_i)$, binary equivalent of x_i (x_i') takes the value 1. Otherwise, x_i' takes the value 0. This is explained by the following equation:

$$x_i' = \begin{cases} 1, \; rand[0,1] < s(x_i), \\ 0, \; \text{otherwise} \end{cases} \quad (16)$$

4.2 Encoding DBGWO to solve MKP problem

The MKP problem is modeled as a pack of wolves searching together to find a best prey i.e. to find a subset of objects from the set of objects to maximize the profit for the knapsack problem. Each wolf x in the pack is encoded as an n-dimensional vector, where n represents the total number of objects available in the object set. Each element x_i $(i=1, ..., n)$ of the x vector represents whether or not the object is chosen to be in the knapsack. To do this, each element is randomly initialized in between two random real numbers. Each element x_i of x is initialized in the interval [-500,500] using the following equation:

$$x_i = -500 + rand[0,1] \times (500 - (-500)) \quad (17)$$

Then the x vector is converted to binary using (15) and (16)

If $x_i = 1$, i^{th} object is present in all the n-dimensions of the knapsack. Else, $x_i=0$ and the i^{th} object is not chosen to be in any of the n-dimensions of the knapsack. For example, Fig. 3 shows a potential solution to the MKP problem with 8 objects. Each bit represents whether the object is chosen to be in knapsacks or not. In the example shown in Fig. 3, 2nd, 4th, 5th and 7th objects are selected to be in all the dimensions of the knapsack whereas 1st, 3rd, 6th and 8th are not chosen to be in any of the dimensions of the knapsack.

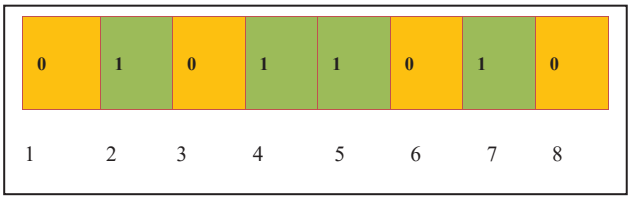


Fig. 3. Sample DBGWO representation of a potential solution to MKP with 8 objects

If a solution x exceed capacity in any dimension of the knapsack, it becomes infeasible. Hence, the infeasible solutions are converted to feasible ones using the check and repair operator method proposed in [1]. This method can be divided into three steps: Preprocessing Phase (Algorithm-1), Removing Phase (Algorithm-2) and Improvement Phase (Algorithm-3).

4.3 Preprocessing phase (Algorithm-1)

This algorithm is implemented before using the DBGWO. It involves the following steps:

- Calculate the profit density (δ_{ij}) of each object for every dimension of knapsack using the following equation:

$$\delta_{ij} = C_j * \left(\frac{P_i}{W_{ij}}\right) \quad (18)$$

where C_j is the capacity of the knapsack in j^{th} dimension, P_i is the profit of the i^{th} object and W_{ij} is the weight of the i^{th} object in j^{th} dimension of the knapsack.

- Calculate the lowest profit density (LPD) for each object

$$LPD = \min\left\{C_j * \left(\frac{P_i}{W_{ij}}\right)\right\} \quad (19)$$

- Sort the objects in ascending order according to the values of LPD.

4.4 Removing Phase (Algorithm-2)

In each iteration, infeasible solutions are converted to feasible ones by removing the objects one by one in the ascending order of LPD (Lowest Profit Density) until the knap-sack capacity constraints gets satisfied. The Pseudo-Code for Removing Phase is shown below:

Algorithm-2: Removing Phase

input: solution vector x
output: repaired solution vector x
for *i=[1, ..., n]* **do**
  */ b_j is the accumulated weight in j^{th} dimension */
  calculate $b_j = \sum_{i=1}^{n} W_{ij} X_i$ for j=1,.....,m.
  if (x_i =1) && ($b_j > C_j$ for any j=1,....,m)
    $x_i = 0$
    for *j=[1, ..., m]*
      $b_j = b_j - w_{ij}$
    end for
  end if
end for

4.5 Improvement Phase (Algorithm-3)

The infeasible solutions that are converted to feasible ones using algorithm-2 are improved by adding the objects one by one in descending order of LPD without exceeding the knapsack capacity in any-dimension. The Pseudo-Code for Improvement Phase is shown below:

Algorithm-3: Improvement Phase

input: repaired solution vector x

output: improved solution vector x

*/ b_j is the Accumulated weight in j^{th} dimension */

calculate $b_j = \sum_{i=1}^{n} W_{ij} X_i$ for j=1,.....,m
for *i=[n, ..., 1]* **do**
  t=0;
  if (x_i = 0)
    for j=[1,...,m] **do**
      if ($b_j + wij < C_j$)
        t=t+1
      end if
    end for
    if (t =m)
      $x_i = 1$
    end if
  end if
end for

Fig. 5 shows the flow chart for implementing the DBGWO algorithm for 0-1 MKP problem.

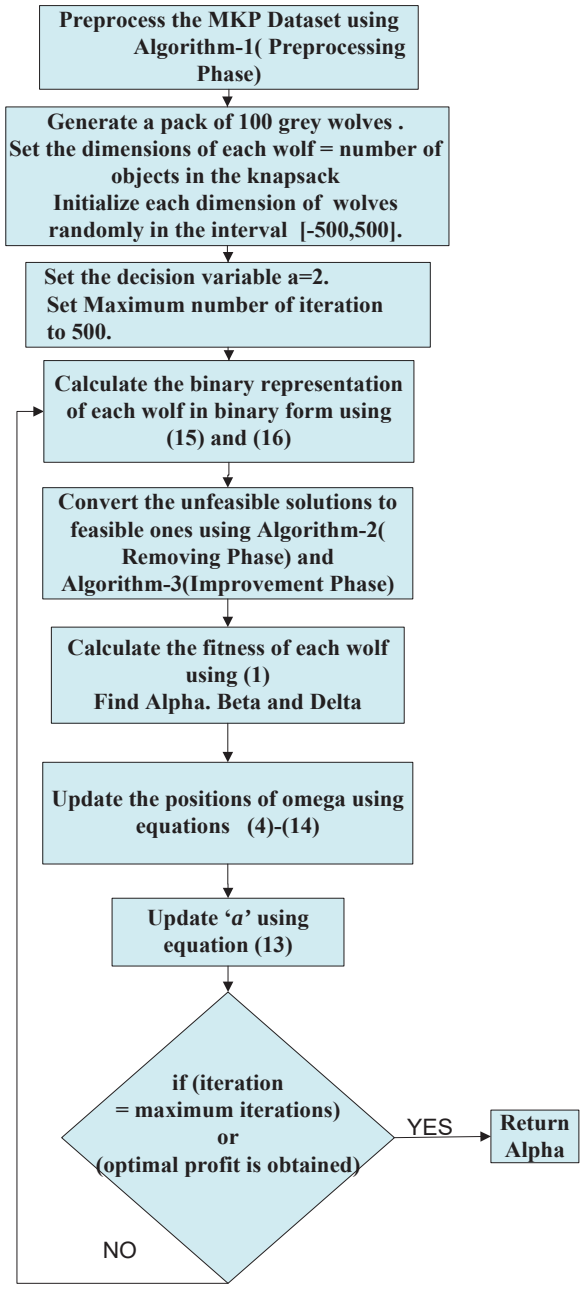


Fig. 5. DBGWO flow chart for solving MKP

5 Results and Comparison

The proposed DBGWO algorithm was applied on seven benchmark Multi-Dimensional Knapsack Problem instances called Petersen (PT) datasets taken from OR-library [27]. The following settings were used for the proposed DBGWO algorithm:

- Number of Wolves=100.
- Maximum Iterations=500.

Table I shows the number of objects, knapsack dimensions and optimal profit of each MKP dataset taken from OR library [22].

TABLE I. MKP DATASETS USED FOR TESTING THE PROPOSED METHOD

Knap Sack Problem Instances from OR-library	No. of Objects	Knapsack Dimensions	Optimal Profit obtained from the OR-library
PT1	6	10	3800
PT2	10	10	8706.1
PT3	15	10	4015
PT4	20	10	6120
PT5	28	10	12400
PT6	39	5	10618
PT7	50	5	16537

The proposed method was run independently for 30 times on each datasets. The performance of the DBGWO algorithm was compared with previous research works which addressed the same PT datasets using three other algorithms namely MHPSO, b-AFSA and EVO[10]-[12].

Table II shows the best profit obtained out of the 30 independent runs and the average profit obtained from the 30 independent runs using the proposed DBGWO along with the results from other algorithms. The Information about the number of independent runs for the EVO algorithm was not provided in the literature [12].

Column 1 and 2 show the best and average profit obtained using the proposed DBGWO algorithm. Column 3 and 4 indicates the best and average profit obtained using b-AFSA algorithm. Similarly, columns 5, 6, 7 and 8 show the best and average profit obtained using the MHPSO and EVO algorithms.

The performance of each algorithm is evaluated based on the average error of the best profits (AE_{best}) and average error of mean profits (AE_{mean}) calculated using the following two equations:

$$AE_{best} = \frac{1}{N}\sum_{i=1}^{N}\frac{O_i - B_i}{O_i} * 100 \quad \%$$

TABLE II. COMPARATIVE RESULTS OF DBGWO, *b* -AFSA, MHPSO AND EVO

MKP Instance	DBGWO		b-AFSA		MHPSO		EVO	
	BEST PROFIT	AVERAGE PROFIT	BEST PROFIT	AVERAGE PROFIT	BEST PROFIT	AVERAGE PROFIT	BEST PROFIT	BEST PROFIT
PT-1	3800	3800	3800	3800	3800	3800	*	*
PT-2	8706.1	8706.1	8706.1	8706.1	8706.1	8706.1	8706.1	8706.1
PT-3	4015	4015	4015	4015	4015	4015	4015	4015
PT-4	6120	6120	6120	6120	6120	6120	6120	6120
PT-5	12400	12399	12400	12382.3	12400	12386	12350	12318.89
PT-6	10618	10597	10618	10559.2	10618	10566	10570	10538.1
PT-7	16537	16482	16537	16422.5	16537	16460	16256	16204

where N is the total number of bench-mark datasets used for testing. ($N=7$ for DBGWO, b-AFSA and MHPSO; $N=6$ for EVO)

O_i is the optimal profit of i^{th} benchmark obtained from OR-library.

B_i is the best profit obtained for i^{th} benchmark out of k independent runs ($k=30$ for DBGWO, b-AFSA and MHPSO; '*k' value* is *unknown* for EVO)

M_i is the average/mean of the profits of k independent runs ($k=30$ for DBGWO, b-AFSA and MHPSO; '*k' value* is *unknown* for EVO)

Table III summarizes the result of four algorithms in terms of average error of best and mean profits (AE_{mean} and AE_{best}).

TABLE III. AVERAGE ERROR OF MEAN AND BEST PROFITS OBTAINED USING DIFFERENT ALGORITHMS:

Algorithm	Average Error of Mean Profits (%) (AE_{mean})	Average Error of Best Profits (%) (AE_{best})
DBGWO	0.0769	0
b-AFSA	0.1526	0
MHPSO	0.1984	0
EVO	0.57	0.42

It can be seen that for DBGWO, b-AFSA and EVO the average error of Best Profit is 0 % as these three algorithms

have resulted in best profits equal to the optimal profits for all the benchmark MKP. But, for the EVO, the average error of best profits (AE_{best}) is 0.42 %. This is because, EVO resulted in best profits much lesser than optimal profits for PT-5, PT-6 and PT-7 datasets.

Average Error of Mean Profits (AE_{mean}) is lowest for the DBGWO with value of 0.0769 %. EVO has the highest AE_{mean} of 0.57 %. AE_{mean} is lesser for b-AFSA and MHPSO when compared to EVO but higher than the proposed DBGWO. This is because, average profit obtained using b-AFSA and MHPSO are lesser than DBGWO for PT-5 and PT-6 and PT-7 datasets.

Hence, we can conclude that DBGWO outperformed the other algorithms with lowest average error of mean profits (AE_{mean}) and 0 % average error of best profit (AE_{best}).

6 Conclusions

Discrete binary version of the GWO algorithm called DBGWO was proposed to solve NP-hard 0-1 Multi-Dimensional Knapsack problem. The mathematical dynamics and encoding of the DBGWO to solve the Multi-Dimensional Knapsack Problem were discussed in detail.

To verify the performance, DBGWO was tested on seven standard MKP benchmarks taken from OR-Library and the results were compared with the previous research which addressed the same MKP datasets using three other algorithms viz., EVO, MHPSO and b-AFSA. DBGWO outperformed the three other algorithms by giving lowest average error of mean profit and zero average error of best profit. This proves the ability of the DBGWO to give optimal solutions to MKP and similar problems.

The High performance of the proposed DBGWO algorithm suggests that, DBGWO can be applied to many real world complex combinatorial optimization problems in future.

7 References

[1] Li, Hong, et al. "Genetic algorithm based on the orthogonal design for multidimensional knapsack problems." Advances in Natural Computation. Springer Berlin Heidelberg, 2006. 696-705.

[2] Balev, Stefan, et al. "A dynamic programming based reduction procedure for the multidimensional 0–1 knapsack problem." *European Journal of Operational Research* 186.1 (2008): 63-76.

[3] Shih, Wei. "A branch and bound method for the multiconstraint zero-one knapsack problem." *Journal of the Operational Research Society* (1979): 369-378.

[4] Hanafi, Saïd, and Arnaud Freville. "An efficient tabu search approach for the 0–1 multidimensional knapsack problem." *European Journal of Operational Research* 106.2 (1998): 659-675.

[5] Drexl, A. "A simulated annealing approach to the multiconstraint zero-one knapsack problem." *Computing* 40.1 (1988): 1-8.

[6] Chu, Paul C., and John E. Beasley. "A genetic algorithm for the multidimensional knapsack problem." *Journal of heuristics* 4.1 (1998): 63-86.

[7] Djannaty, Farhad, and Saber Doostdar. "A hybrid genetic algorithm for the multidimensional knapsack problem." *International Journal of Contemporary Mathematical Sciences* 3.9 (2008): 443-456.

[8] Khuri, Sami, Thomas Bäck, and Jörg Heitkötter. "The zero/one multiple knapsack problem and genetic algorithms." *Proceedings of the 1994 ACM symposium on Applied computing*. ACM, 1994.

[9] Kong, Min, Peng Tian, and Yucheng Kao. "A new ant colony optimization algorithm for the multidimensional Knapsack problem." *Computers & Operations Research* 35.8 (2008): 2672-2683.

[10] Azad, Md Abul Kalam, Ana Maria AC Rocha, and Edite MGP Fernandes. "Solving multidimensional 0–1 knapsack problem with an artificial fish swarm algorithm." Computational Science and Its Applications–ICCSA 2012. Springer Berlin Heidelberg, 2012. 72-86.

[11] Labed, Said, Amira Gherboudj, and Salim Chikhi. "A modified hybrid particle swarm optimization algorithm for multidimensional knapsack problem." Int. J. Comput. Appl 34.2 (2011): 11-16.

[12] Sur, Chiranjib, Sanjeev Sharma, and Anupam Shukla. "Egyptian vulture optimization algorithm–a new nature inspired meta-heuristics for knapsack problem." The 9th International Conference on Computing and InformationTechnology (IC2IT2013). Springer Berlin Heidelberg, 2013.

[13] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. Advances in Engineering Software, 69, 46-61.

[14] Mirjalili, S. (2015). How effective is the Grey Wolf optimizer in training multi-layer perceptrons. Applied Intelligence, 1-12.

[15] Wong, L. I., Sulaiman, M. H., Mohamed, M. R., & Hong, M. S. (2014, December). Grey Wolf Optimizer for solving economic dispatch problems. InPower and Energy (PECon), 2014 IEEE International Conference on (pp. 150-154). IEEE.

[16] Emary, E., Zawbaa, H. M., Grosan, C., & Hassenian, A. E. (2015, January). Feature Subset Selection Approach by Gray-Wolf Optimization. In Afro-European Conference for Industrial Advancement (pp. 1-13). Springer International Publishing.

[17] Madadi, A., & Motlagh, M. M. (2014). Optimal Control of DC motor using Grey Wolf Optimizer Algorithm.

[18] Mahdad, B., & Srairi, K. (2015). Blackout risk prevention in a smart grid based flexible optimal strategy using Grey Wolf-pattern search algorithms. Energy Conversion and Management, 98, 411-429.

[19] Srivathsan Lakshminarayanan and Devinder Kaur, "Efficient Energy Management in Green Smart Home with Grey Wolf Optimizer," unpublished.

[20] Wolfcountry.net, 'Wolf Country, wolves as hunters, survival', 2015. [Online]. Available: http://www.wolfcountry.net/information/WolfHunting.html. [Accessed: 10- Jul- 2015].

[21] Gherboudj, Amira, Abdesslem Layeb, and Salim Chikhi. "Solving 0-1 knapsack problems by a discrete binary version of cuckoo search algorithm."International Journal of Bio-Inspired Computation 4.4 (2012): 229-236.

[22] People.brunel.ac.uk, 'Multiple knapsack', 2015. [Online]. Available: http://people.brunel.ac.uk/~mastjjb/jeb/orlib/mknapinfo.html. [Accessed: 24- Aug- 2015].

The design of a Bayesian Network vehicle traffic flow prediction model for Johannesburg

Ziphozethu Nkosi
University of Johannesburg
Auckland Park 2006
Johannesburg, South Africa

zipho.sokhela@gmail.com

Barnabas Ndlovu Gatsheni
University of Johannesburg
Auckland Park 2006
Johannesburg, South Africa

0027115591506
bgatsheni@uj.ac.za

ABSTRACT

Vehicle traffic congestion in Johannesburg has a negative impact on the economy of South Africa in that services and products are not being rendered on time. In this paper prediction models were constructed using historical data and machine learning algorithms to help commuters mitigate traffic congestion. The results show that the Bayesian model provides a reliable alternative for traffic flow prediction as it outperformed the Naive Bayes, K-NN and the Decision tree models. Cross-validation and the rmse were used to evaluate the models. These results will benefit commuters and employers and save costs for companies, improve the South African economy as well as assist Johannesburg in aligning future road traffic strategies.

Keywords – d-separation, confusion matrix, cost matrix, Bayesian Networks

1.0 INTRODUCTION

Traffic congestion impacts negatively on the economy of Johannesburg, for example, as a result services are not being provided on time [5]. The city of Johannesburg contributes just under a quarter of the gross domestic product (GDP) of South Africa and thus anything that goes wrong in this metropolitan area tends to be felt throughout the whole country. It is estimated that every month companies lose approximately US $95 million in the form of paying salaries [8] for hours not worked as a result of workers who report late for work. It has been established that 8% of employees' meetings in Johannesburg are delayed because of traffic congestion [8].

In addition, traffic congestion can prevent emergency services from reaching on time individuals who are in a critical condition.

Among the known causes of traffic congestion are incidents such as car accidents, roadblocks, bad weather and lane closure due to road works.

Johannesburg's population of 5,065,100 as of in 2015 [12] is growing at 3.18 percent per annum due to rural to urban migration and the influx of migrants all seeking fortune in this city [2]. This growth rate has resulted in a 0.37% increase in the number of vehicles per month [3] and hence traffic congestion. Vehicle traffic flow prediction is an intervention that can improve the punctuality and the quality of life of commuters by reducing the long commuting time than when traffic is flowing freely [1].

2.0 LITERATURE REVIEW

Ma *et al.* [10] used vehicle speed profiles and lane changing behaviour as the attributes to determine the occurrence and characteristics of an incident which was categorised into normal conditions, passing a possible incident scene, and stopped in a queue [10]. They presented a real-time traffic condition assessment framework conducted in a simulation environment in Spartanburg. The results showed 95% accurate vehicle traffic incident detection for support vector machines (SVM) and 92% for the artificial neural networks (ANN). The two attributes used are insufficient. The time of day when the incident occurred would be useful. The research was conducted under simulation environment, which may produce different results from those that can be obtained from a real-world environment.

This research differs from [10] in that this study predicts vehicle traffic flow using more data in the form of the attributes that include vehicle volume, speed, time of day and day of the week and thus it potentially gives better coverage of the vehicle traffic congestion problem. The feature selection results in section 3 will show that all these attributes do contribute to the prediction.

Sun *et al.* [11] and Hoong *et al.* [7] proposed the Bayesian network (BN) model for forecasting vehicle traffic flow evaluated using the minimum mean square error (MMSE).

Although they use data from adjacent road links they claim their BN model can predict traffic flow when there is missing data. The RMSE obtained was 0.13. The weakness of the study is that the pre-processing of the data was not mentioned. In addition when large data is used the model may not be perform optimally.

Hoong *et al.* [7] proposed a BN framework for road condition prediction. For the evaluation process, they created the Naive Bayesian Network (M2) and parameter-learning Bayesian Network (M3) as benchmarks to measure the predictive accuracy of the proposed BN (M1). M2 depicted the average accuracy (58.57%), M1 (74.37%), while M3 scored 76.01%. Their results are lower than the results from this study. Their work relied on tweets and user reports as a main source of traffic status data which if not reported on time could result in commuters getting trapped in traffic due to this blackout of information.

Ji *et al.* [9] created a genetic neural network (GNN) vehicle traffic flow forecasting model using data collected in Jinan. The ANN weights and biases were obtained using the genetic algorithm and the backpropagation algorithm was used for training the ANN. The results show biases of -0.1150 and weight of -0.1677 which mean that the GNN is a good and valid method for traffic flow forecasting. The weakness of the study is that they did not discuss the results.

The research in [10] [9], did not address attributes that include day of the week, public holidays, and an increase in the population. This paper uses the Bayesian networks, the k-NN and the decision tree to predict vehicle traffic flow on the Johannesburg N1 freeway. In this way, the focus is on the study of traffic patterns. The vehicle traffic congestion attributes were built into the model.

Related work shows that BN [11] [7] [9] [6] followed by the ANN [9] gives good results. The ANN in not used in this paper because it requires a large amount of data to produce better prediction results. Thus the most suitable approach for related work is the Bayesian Network model. In addition, some of the models used on related work were carried out under simulation environments, which may differ when used in the real world situations. In addition, some of the reported work does not cover a whole year of data.

3.0 METHODS

Bayesian networks (BN) also called Bayesian belief networks (BBN) describe the probability distribution over a set of variables (nodes) by specifying a set of conditional independence assumptions along with a set of conditional probabilities [18]. In BN, learning constitutes parameter estimation and structure optimisation. BN network provides knowledge of the structural relationship (links) between nodes and these nodes can be events, situations etc. These links integrate the situations and events to form a holistic view of their meaning.

Each random variable can take on the set of possible values $V(Y_i)$. Thus the *joint space* of the set of variables Y is the cross product$V(Y_1) \times V(Y_2) \times \ldots . V(Y_n)$. Each item in the *joint space* corresponds to one of the possible assignments of values to the instance of variables. The probability distribution over this joint is called the joint probability distribution (jpd) [18] which is similar to the knowledge base in expert systems. Thus any query on this domain has an answer provided by the jpd. A BN describes the joint probability distribution for a set of variables [18].

BN unlike in other knowledge-based systems, handles uncertainty using mathematics in a simple efficient way. In addition, BN uses network representation of problems and statistics or maths to solve problems.

- BN becomes a tool for projecting future events or situations. In BN the network captures the qualitative nature of dependence relations, and the quantitative is captured by the conditional probability table which gives contextual knowledge as a compact and single model.

The other reason for choosing BN is that it is able to deal with incomplete data using sound mathematical theory.

3.1 D-Separation

Conditional independence in BN was explained through d-separation. Nodes X and Y are d-separated if on any undirected path between X and Y there is a variable Z such that:

- Z is in a serial or diverging connection and Z is known or observed.
- Z is in a converging connection and neither Z nor any of Z's descendants are known or observed.

If X and Y are d-separated by Z, then X and Y are conditionally independent.

3.1.1 Steps To Test D-Separation

In this paper d-separation was established through constructing the ancestral graph consisting of triples *A, B* and *C* together with all nodes from which there is a directed path to *A, B* or *C*.
If all paths connecting *A* and *B* intersect *C* in the moral graph, then *A* and *B* are d-separated given *C*. All the triples in the moral graph were examined in this fashion.
Once we were satisfied that the network met the conditional independence requirements, a Bayesian prediction model was constructed as detailed in the next section.

3.2 Steps for Constructing a BN Prediction Model

1. Define from the instance space *X*, the set of relevant target value X_i .
2. Provide the training set of the target function
3. Add the instance, X_i to the network.
4. Add relations to the node X_i from a set of selected attributes to form a network

$$P(X_i \mid X_1, X_2 ..... X_m) = P(X_i \mid Parents(X_i))$$

Where $X_1, X_2 ..... X_m$ are all the variables preceding X_i that are not in $Parents(X_i)$

5. Determine the conditional probability table given by, $P(X_i \mid Parents(X_i))$ for each attribute

Prediction is the foretelling of the occurrence of an event based on historical data. The model must be able to predict future unseen data that come from the same distribution as data used to construct the model. For the model to be able to predict traffic congestion, one needs to first train the model and then deploy a trained model to predict a new instance (unseen data) as shown in Figure 2.1.

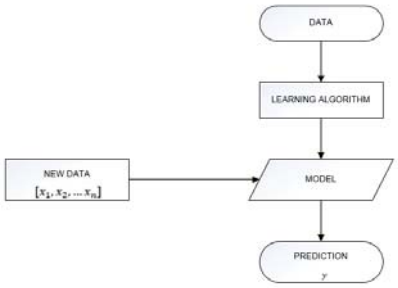


Figure 2.1- Shows steps for the construction of the traffic flow prediction Model

The steps for training the model are as follows:

Step 1 – Load the training examples on WEKA

Step 2 – select the Bayesian algorithm, then start training

Step 3 – Save the model

The saved model in step 3 is the brains of the system as it has captured the underlying function that describes the data. This model is used for prediction in steps 4 and 5.

Step 4 – upload the unseen test data

Step 5 – Reload the saved model from step 3 by right clicking the result list then select re-evaluate model. The output is the prediction of new (unlabelled) data.

4.0 EXPERIMENTATION

The tools used are machine learning algorithms from WEKA software package.

A set of 347 training examples, where each instance is described by the attributes Day, Time of day, Volume, Speed and Class (Target) were used. The model needs to predict the target value (freeflow, trafficJam, flowingCongestion) which is the state of vehicle traffic flow on the freeway. The total number of instances is 520.

The measures that were used to assess the model's performance are the accuracy for prediction which is a ratio of the number of correct predictions and the total number of test examples; the RMSE defined by equation 3.1; and the cost of prediction defined by equation 3.5. Confusion matrix is used together with the loss matrix to calculate the cost of prediction for each model. The confusion matrix shows a report on the number of instances that the model predicted correctly or incorrectly.

$$\text{RMSE} = \sqrt{\frac{\sum (p_i - y_i)^2}{M}} \qquad \text{(Equation 3.1)}$$

In equation 3.1:

p is the model's predicted output

y_i is the true output for example ***i*** and

M is the total number of test examples.

A high RMSE means that the result is poor, while a low RMSE means that the result is good. Two-thirds of the data was set aside as training data and one-third for test data as shown in Table 3.1.

Table 3.1– A summary of the data sets

	Classes (target values)				
Instances		**TraffiJam**	**FreeFlow**	**Flowing Congestion**	**Total**
	Training set	101	160	86	347
	Testing set	46	76	51	173
	Total instances	147	236	137	520

4.1 The Attribute Selection

The attribute selection was done manually and it was used to identify the attributes that contribute most to the vehicle traffic flow prediction. These attributes included day, time of day, volume and speed. Time of day includes peak time (06h00 - 09h00, 15h00 - 18h00), off-peak time (10h00 - 13h00, 19h00 - 22h00), vehicle Speed is segmented into (<= 40 km/h low, >=41 km/h and <=100 km/h average and >100 km/h high), day which is the day of week, volume of vehicles was segmented into (<=3000 low, <=60000 average <=90000 high).

Table 3.2– Attribute selection comparing two attributes for the BN model dataset

	Learning Algorithm	Attributes	RMSE	Correct Prediction
1	BN	Time of day and Speed	0.14%	96%
2	BN	Day and Speed	0.14%	96%
3	BN	Volume and Speed	0.14%	96%
4	BN	Time of day and Volume	0.31%	80%
5	BN	Day and Volume	0.35%	63%
6	BN	Time of day and Day	0.24%	58%

Results in Table 3.2 mean that speed and time of day; day and speed; volume and speed realised a better prediction of 96% and RMSE of 0.14%. Results in Table 3.1 mean that speed and time of day are the two attributes that influence the vehicle traffic flow status the most.

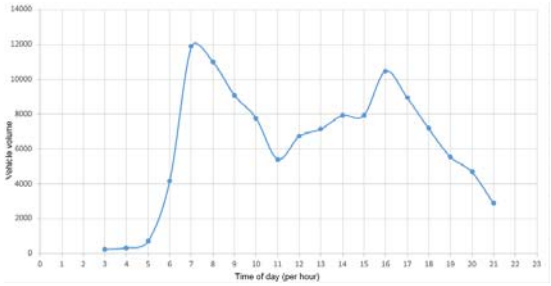


Figure 3.1– Shows the variation of vehicle traffic volume with time of day

Figure 3.1 shows that the largest vehicle traffic volume is at 07.15hrs and at this time, the volume is slightly more than 12000 vehicles. The next highest is at 16.00hrs with about 10600 vehicles. The evening peak is lower as knock off-time is spread from 15.30hrs to 17.00hrs whereas in the morning peak companies start work between 07.30hrs and 08.00hrs.

4.2 The Bayesian Networks

The results in Table 3.3 mean 74 Freeflow instances were predicted correctly and 2 instances were predicted incorrectly as Trafficjam. A total of 46 training instances were predicted correctly as TrafficJam and 51 instances were predicted correctly as FlowingCongestion.

Table 3.3– Bayesian Net Confusion matrix

Actual	Predicted			
		Freeflow	**Traffic Jam**	**FlowingCongestion**
	Freeflow	74	2	0
	TrafficJam	0	46	0
	Flowing Congestion	0	0	51

The BN model using the K2-P1 S search was able to predict the test instances at 98.8% correctly with an RMSE of 0.06. The lower the RMSE the better is the prediction model.

4.3 The Naïve Bayes Model

The Naïve Bayes Model was constructed as follows:

Step 1: Instantiate Equation 3.2 to fit the current task. The target value $\mathcal{V}_{N\mathcal{B}}$ output is given by the Naïve Bayes predictor:

$$\mathcal{V}_{N\mathcal{B}} = \underset{v_j \in \{freeflow, trafficjam, flowingCongestion\}}{\operatorname{argmax}} P(v_j) \prod_i P(a_i | v_j)$$

$$= \underset{v_j \in \{freeflow, trafficjam, flowingCongestion\}}{\operatorname{argmax}} P(v_j)$$

(Equation 3.2)

Now given a new instance:

P(Day=weekday|v_j) P(Time of day=off-peak|v_j)
P(Volume=Low|v_j) P(Speed=Average|v_j)

The state of vehicle traffic flow on the freeway was predicted as follows:

Step 2: Determine the probabilities of the target values based on the occurrences over the training data.

P(*State = freeflow*) = 160/ 347 =.46

P(*State = trafficJam*) = 101/ 347=.29

P(*State = flowingCongestion*) = 86/ 347 =.24

Step 3: Estimate the conditional probabilities

Those of Volume = Low are

P(*Volume= Low*|*State=freeflow*) = 139/ 139=1

P(*Volume = Low*|*State=trafficjam*) = 0/ 0=0

P(*Volume = Low*|*State=flowingCongestion)* = 0/0 =0

The results from Table 3.4 mean that 74 Freeflow instances were predicted correctly and 2 instances were predicted incorrectly as Trafficjam. The 46 actual instances for Trafficjam were all

correctly predicted as Trafficjam, and 51 instances were predicted correctly as FlowingCongestion.

Table 3.4– Naive Bayes Confusion matrix

Actual	Predicted			
		Freeflow	**Traffic Jam**	**FlowingCongestion**
	Freeflow	74	2	0
	TrafficJam	0	46	0
	FlowingCongestion	0	0	51

The NB was able predict the test instances at 98% correctly with an RMSE of 0.09.

4.5 Recipe For the *k*-Nearest Neighbour (k-NN)

The k-NN stores training examples and classification. What makes it lazy is that *k*-NN defers the decision on prediction until a new query instance x_q is encountered. Thus learning consists of storing presented training data. When a new query instance x_q is encountered, similar instances are retrieved from memory and used to classify x_q. Thus k-NN does not form an explicit general hypothesis regarding the target function. Table 3.5 shows that the k-NN model performance is 98% prediction with an RMSE of 0.07.

Table 3.5– *k*-Nearest Neighbor Confusion matrix

Actual	Predicted			
		Freeflow	**Traffic Jam**	**FlowingCongestion**
	Freeflow	74	2	0
	TrafficJam	0	46	0
	FlowingCongestion	0	0	51

4.6 The Decision Tree Model

Recipe for the decision tree

Step 1: Calculate the entropy of each attribute in the training example to determine which attribute should be tested first in the tree. Entropy is calculated using Equation 3.3

$$Entropy\ (B) \equiv -p_{\oplus} log_2 p_{\oplus} - p_{\ominus} log_2 p_{\ominus} \quad \text{(Equation 3.3)}$$

Step 2: Calculate information gain of each attribute in the training data using Equation 3.4.

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad \text{(Equation 3.4)}$$

Step 3: From Step 2, choose the attribute with the most information gain.

Step 4: Calculate the entropy of each attribute value of the root node and partition them against all the remaining attributes.

Step 5: Calculate the information gain of the attribute values against the remaining attributes. The attribute with the highest information gain is the next attribute to branch from.

Step 6: Repeat the process until all attributes are included in the tree.

The Decision tree model is able to predict 95% test instances correctly with an RMSE of 0.14.

Table 3.6 shows the Confusion matrix for Decision trees which was used in section 4 to calculate the cost of vehicle traffic prediction.

Table 3.6– Decision Tree Confusion matrix

Actual	Predicted			
		Freeflow	**Traffic Jam**	**FlowingCongestion**
	Freeflow	69	2	5
	TrafficJam	0	46	0
	FlowingCongestion	0	0	51

4.7 Post-Processing of Results

The Confusion matrix was used to identify the prediction rate for each algorithm in section 3, where the false positive and the false negative was used to evaluate the model. The Loss matrix is used to measure the overall cost incurred in taking any of the variable decisions or actions [4]. In Table 3.7 a penalty of 4 was assigned when a model predicted FreeFlow when the actual traffic state was TrafficJam. A 1 was assigned when it predicted TrafficJam when the traffic status was FlowingCongestion. A 2 was assigned when it predicted FlowingCongestion when the actual traffic status was TrafficJam. A 3 was assigned when it predicted FreeFlow when the actual traffic status was FlowingCongestion. A 3 was assigned when it predicted TrafficJam when the actual traffic status was FreeFlow. A 1 was assigned when it predicted FlowingCongestion when the actual traffic status was FreeFlow. The cost of the model is given by Equation 3.5.

$$C_{COST} = \sum C\_M_{kj} * L\_M_{kj} \quad \text{(Equation 3.5)}$$

$$Cost = \sum confusion\ matrix\ X\ Loss\ Matrix$$

Where C_{COST} is the cost incurred, C_M_{kj} is the Confusion matrix from Table 3.3 and L_M_{kj} is the Loss matrix from Table 3.7. The elements (L_M_{kj}) of the loss matrix in Table 3.7 specify the penalty associated with the prediction. These elements were chosen by hand.

Table 3.7: The Loss Matrix for computing the cost of vehicle traffic prediction

		Predicted		
		FreeFlow	**TrafficJam**	**FlowingCongestion**
Actual	FreeFlow	0	3	1
	TrafficJam	4	0	2
	Flowing Congestion	3	1	0

4.7.1 Total cost for Naive Bayes

The cost for Naïve Bayes is obtained by multiplying cell values of the Loss matrix in Table 3.7 with the corresponding cell values of the Naïve Bayes model's Confusion matrix in Table 3.4; then add the sums to get the cost for vehicle traffic flow prediction given in equation 3.5:

$$C_{COST\_MLP} = \sum C\_M_{kj} * L\_M_{kj} \qquad \text{(Equation 3.5)}$$

Total cost = 74x0 + 2x3 + 0x1 + 0x4 + 46x0 + 0x2 + 0x3 + 0x1 + 51x0 = 0.6

Total cost for Bayesian Network

The Cost computed as in Naïve Bayes (NB) but using the Confusion matrix in Table 3.3. The cost of vehicle traffic flow prediction is 6.

Total Cost for *k*-Nearest Neighbour

The Cost computed as in NB but using the Confusion matrix in Table 3.5. The cost of vehicle traffic flow prediction is 6

Total cost for Decision tree

The total Cost computed as in NB but using the Confusion matrix in Table 3.6. The cost of vehicle traffic flow prediction is 11.

4.7.2 Vehicle Traffic Flow Prediction Results

Results in Table 3.8 mean that BN outperforms the other models. Although the other models performed well as the BN model but with a higher RMSE value.

Table 3.8– Prediction models results

Model	Test Prediction Accuracy	RMSE	Cost
Bayesian Networks	98.8%	0.06	6
Naive Bayes	98%	0.09	6
K-Nearest Neighbour	98%	0.07	6
Decision Tree	95%	0.14	11

Figure 3.2 shows RMSE values were obtained from WEKA. The BN has the lowest RMSE of 0.06 which makes it the best model, with the Decision tree with the highest RMSE of 0.14.

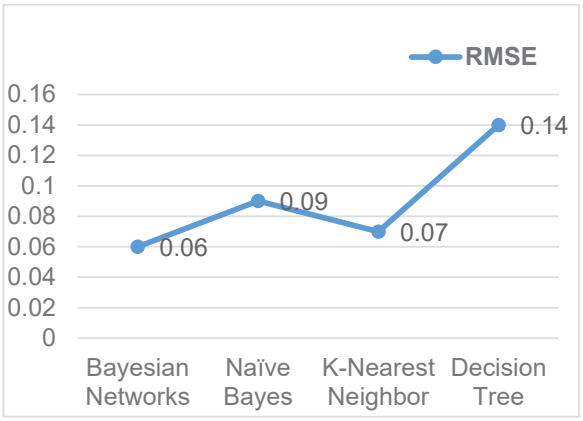


Figure 3.2–shows the RMSE of each model

5.0 DISCUSSION

The results in Table 3.8 mean that the Bayesian network (BN) model is a reliable tool for predicting the state of traffic flow. It achieved a test prediction accuracy of 98.8% and a cost of 6 and the lowest RMSE of 0.06. The results are what they are because the BN has the ability to deal with uncertain situations when there is missing data. Thus using this model commuters are less likely to make costly travel decisions.

The results also mean that whilst the *k*-NN model performs poorly than BN with an RMSE of 0.07, it is a better model to use in predicting traffic flow congestion compared to the naïve Bayes (NB) model. The results for the *k*-NN are what they are because *k*-NN makes no assumption about the data distribution (cf. NB model assumes attributes are conditionally independent). However, the *k*-NN model breaks down quickly and its performance degrades when there are few data points for comparison.

The Decision tree model results in Table 3.8 mean that it is a poor model to use in predicting vehicle traffic flow. The decision tree had been expected to do well as it has been found to be able to handle continuous variables well such as temperature which is almost similar in behaviour as the vehicle traffic flow. The difference between the performances of these 4 models is small and thus any of the other 3 models can be equally used for predicting traffic congestion.

Our results are better than that of Sun *et al.* [11] and Hoong *et al.* [7] who achieved a poor RMSE of 0.13. Their approach differs

from this study in that whilst it uses incomplete data, their data was from adjacent road links.

Our model performs better than Hoong *et al.* [7] whose BN framework for road condition predictions achieved 76.01%.

The implications of BN traffic prediction model is that commuters will know the status of vehicle traffic flow on the freeway in advance and thus make informed decisions in mitigating traffic congestion. This will save businesses millions of US dollars per month of lost production due to workers reporting late for work.

The prediction model is only valid for 18 months. Having a model that predicts traffic for 5 years is not realistic as 5 years is a long time and new variables could come and disrupt the pattern and hence result in the degradation in performance by the model. As an example, within the past 5 years there has been an introduction of cycle tracks and corridors of freedom in Johannesburg which are meant to encourage more public transport use. Further work can include collecting weather and road construction data.

6.0 CONCLUSION

A vehicle traffic flow prediction model based on the Bayesian Networks (BN) was built using vehicle traffic flow historical data captured from Johannesburg freeways. From the results, the BN model outperformed the Naive Bayes, k-Nearest Neighbor and Decision tree in predicting vehicle traffic flow. The results of this study will benefit commuters and employers by potentially reducing stress levels and saving on production costs and potentially improve the efficiency of South African organisations. The study recommends that Johannesburg must intervene in the area of vehicle traffic speed and also target specific times of the day to promote free flowing traffic. It also recommends that Johannesburg must invest in machine learning tools to enhance vehicle traffic flow on Johannesburg freeways.

7.0 REFERENCES

[1] Acentric. Acentric survey reveals traffic congestion's effect on SA. 2011.

[2] Brinkhoff, T. Major agglomerations of the world. 2013. Available at http://www.citypopulation.de/world/Agglomerations.html.

[3] City of Johannesburg Metropolitan Municipality. The Local Government Handbook: A complete guide to municipalities in South Africa. 2012. Available at http://www.localgovernment.co.za/metropolitans/view/2/city-of-johannesburg-metropolitan-municipality.

[4] DistanceFrom. Travel Time Calculator. 2013. Available at http://www.distancesfrom.com/Travel-Time.aspx.

[5] ENaTiS. Live vehicle population. 2013. Available at http://carinsurance.arrivealive.co.za/category/car-statistics.

[6] Ghazy, A., and Ozkul, T. Design and simulation of an artificially intelligent VANET for solving traffic congestion. In *Proceedings of IEEE 6th International Symposium on Mechatronics and its Applications*., (Mar. 2009), 1 – 6.

[7] Hoong, P., Tan, I., Chien, O., and Ting, C. Road traffic prediction using Bayesian Networks. In *Proceedings of IET International Conference on Wireless Communications and Applications (ICWCA '12)* (Kuala Lumpur, October 8-10, 2012). IET, 2012, 1-5.

[8] InterNations. Johannesburg at the glance: working in Johannesburg. 2013. Available at http://www.internations.org/johannesburg-expats/guide/working-in-johannesburg-15840.

[9] Ji, T., Pang, Q., and Liu, X. Study of Traffic Flow Forecasting Based on Genetic Neural Network. In *Proceedings of IEEE 6th International Conference on Intelligent Systems Design and Applications* (Jinan, October 16-18, 2006). IEEE, 2006, 960 – 965.

[10] Ma, Y., Chowdhury, M., Sadek, A., and Jeihani, M. Real-time highway traffic condition assessment framework using vehicle-infrastructure integration (VII) with Artificial Intelligence (AI*)*. In *Proceedings of IEEE Transactions on Intelligent Transportation Systems*., 10, 4 (Dec. 2009), 615-627.

[11] Sun, S., Zhang C., and Yu, G. A Bayesian network approach to traffic flow forecasting. In *Proceedings of IEEE Transactions on Intelligent Transportation Systems*., 7,1 (Mar. 2006) , 216 – 221.

[12] Vorster, G. Joburg vs Cape Town: Best city challenge. 2015. Available at http://businesstech.co.za/news/general/82201/joburg-vs-cape-town-best-city-challenge.

A novel extreme learning machine-refined binary coding based feature extraction algorithm

Maziar Moradi Fard
Department of computer science and electrical engineering
Islamic Azad University of Science and Research
Tehran, Iran
maziar.moradifard@srbiau.ac.ir

Hossein Ghomeshi
Department of Engineering and Technology
University of Tehran, Kish International Campus
Kish, Iran
h.ghomeshi@ut.ac.ir

***Abstract*—In this paper a novel feature extraction method based on extreme learning machine (ELM) and binary coded based feature extraction (BCFE) is proposed. In the first step, the dataset is classified by ELM to determine the number of extracted features. The second step is extracting important features based on refined BCFE (RBCFE) algorithm. In RBCFE, features should be segmented and the weighted mean of features in each segment is considered as a new feature vector. To achieve the final weights, between class information and within class information will be used. Finally, to classify the dataset, support vector machines are employed. The results demonstrate considerable improvement in BCFE algorithm.**

Keywords—Pattern recognition, Feature extraction, classification, Binary coding.

I. INTRODUCTION

Pattern recognition searches for possible existing patterns in a dataset. In fact, pattern recognition looks for similarities and dissimilarities in a dataset. As the use of pattern recognition is growing widely in various fields of studies [1-3], it has motivated a lot of researchers to develop new algorithms. Since the goal of pattern recognition is to identify existing patterns in a given dataset, the structure of the dataset is very important. One of the major problems that we often face is the high dimensionality of dataset. This problem occurs when the goal is to classify the dataset but there are some redundancies in features of the dataset. In order to find these redundancies, the feature extraction step is required. In this step, the dimensionality of the dataset is reduced so that the classification could be done more efficiently. In fact, feature extraction projects the data samples in a new space with lower dimensionality and complexity. Since feature extraction has a direct influence on the classification accuracy, many algorithms have been proposed to develop new feature extractors. One of the most well-known methods that has been used by many researchers is principal component analysis (PCA) [4]. In PCA the information in the scatter total class of a dataset is utilized to obtain optimum projection vectors. This method ignores the class labels information. Another popular method is linear discriminant analysis (LDA) [5]. Unlike PCA, LDA uses class labels information in order to extract final projection vectors. Although these methods are very useful, more powerful algorithms have been proposed recently. Some of these methods are based on reconstruction of each data point according to its k nearest neighbors like locally linear embedding (LLE) [6] and locally preserving projections (LPP) [7]. So far, many derivatives of LLE and LPP algorithms have been proposed [8-13]. Although these methods are very useful, they are computationally expensive. To address this problem a new method called binary coding based feature extraction (BCFE) is proposed recently [14]. In this method the feature vectors have to be partitioned into different segments. Then each final feature vector is obtained according to the weighted mean of features in related segment. To obtain the final weight vector, the binary values of class means have to be calculated according to a specific approach. This approach will be discussed in details later in this paper. BCFE method uses binary coded class means differences and also edges of binary coded class means to calculate the final weight vector. In some aspects, like speed and accuracy, it has been proved that this method is more efficient than other traditional methods for feature extraction in remote sensing high dimensional data [14]. Despite all of its advantages, this method has some drawbacks. First of all, the number of features that have to be extracted is remained unspecified. Second, this method ignores the within class information. To address these problems, we have proposed an extreme learning machine-refined binary coding based feature extraction (ELM-RBCFE) method. In the proposed algorithm, we first use an ELM to classify the dataset, then the number of neurons corresponding to maximum value of accuracy is considered as the number of features that should be extracted. Afterwards, by using BCFE algorithm we can obtain between class information to compute the first weight vector. Then the within class information have been employed to compute the second weight vector. Eventually the final weight vector will be calculated according to the first and second weight vectors. By calculating weighted feature vectors in each segment, the final features can be obtained. In the end, support vector machine (SVM) is used to classify the dataset. We will describe the proposed algorithm in details later in this paper. The results of implementing of our algorithm shows considerable improvements in BCFE algorithm. This paper is organized as follows. In section 2 the review of ELM and BCFE algorithms are provided. In section 3 The proposed algorithm is described. In section 4 the results of implementing the proposed algorithm and other feature extraction algorithms on variety of standard datasets are

demonstrated, and finally section 5 contains discussion of the conclusion.

II. REVIEW OF ELM AND BCFE

Since the proposed algorithm is based on ELM and BCFE, in this section we will review the ELM algorithm and demonstrate how it classifies the dataset. Next the BCFE algorithm will be described.

A. Review of ELM

ELM algorithm is a single layer neural network algorithm that was proposed in [15] to deal with problems that exist in feed forward neural networks. One of the major drawbacks of feed forward neural networks is that biases and weights should be tuned during each iteration. In fact, learning algorithms are needed in order to find the optimum values for biases and weights. Moreover, the process of finding optimum values of weights and biases is time consuming. For instance, gradient descent is a well-known algorithm that is used to train neural networks. Alongside the fact that this method is time consuming, the value of learning rate can affect the algorithm drastically. For example, if the learning rate is too small then the algorithm fails to achieve optimum values for weights and biases. Also, if the learning rate is too large then the algorithm will diverge. To avoid this problem, ELM was proposed based on Bartlett's theory [16]. This theory states that smaller norm of weights yields better generalization performance. ELM removes the iterative learning part of the feed forward neural networks. In this algorithm the weights and biases of the hidden layer are chosen randomly because as long as the activation function is infinitely differentiable, the values of weights and biases are not important [15]. In this algorithm the hidden layer matrix (denoted by H) is obtained in formula (1).

$$H(w_1 \dots w_{\acute{N}}, b_1 \dots b_{\acute{N}}, x_1 \dots x_N) = \begin{bmatrix} g(w_1 . x_1 + b_1) & \dots & g(w_{\acute{N}} . x_1 + b_{\acute{N}}) \\ \vdots & \dots & \vdots \\ g(w_1 . x_N + b_1) & \dots & g(w_{\acute{N}} . x_N + b_{\acute{N}}) \end{bmatrix} \quad (1)$$

Where, $g(x)$ is the activation function which is infinitely differentiable, N is the number of training samples and $\acute{N}$ is the number of neurons. Weights and biases are denoted by w and b. After computing H, the next step is to calculate the weight vector that connects hidden nodes and output nodes which is denoted by Θ. This weight vector can be obtained according to formula (2).

$$\Theta = (HH^t)^{-1}.H^t.T \quad (2)$$

Where T is the target vector containing labels of training samples and H^t is the transpose of matrix H. In the following, we have summarized the algorithm.

- Suppose $\mathcal{E}=\{(x_i, t_i) | \ x_i \in R^n, \ t_i \in R^m, \ i=1,...,n\}$, $g(x)$ is the activation function.
- Weights and biases have been set to random numbers in the range of 0 to 1.
- The hidden layer matrix (H) should be calculated according to formula (1).
- The weights of output (Θ) should be calculated according to formula (2).

Then the accuracy of classification can be obtained simply. Next we will represent BCFE algorithm.

B. Review of BCFE

BCFE uses binary coded class means to extract between class information. The algorithm contains several steps. In the first step the feature vectors should be partitioned into some segments. Indeed, if the number of features is d and the number of extracted feature vectors is m, then the number of segments is calculated as formula (3).

$$k = \frac{d}{m} \quad (3)$$

The BCFE calculates the final feature vectors as a weighted mean of feature vectors in each segment. If the original feature vectors are considered as $X = \{x_1, x_2, \dots, x_d\}$ and the extracted feature vectors are considered as $= \{y_1, y_2, \dots, y_m\}$ and w_i denotes the ith value of the final weight vector, then Y is calculated as formula (4).

$$y_j = \sum_{i=(j-1)k+1}^{jk} w_i x_i \qquad 1 \le j \le m-1 \quad (4)$$

$$y_m = \begin{cases} \sum_{i=(m-1)k+1}^{d} w_i x_i & mk < d \\ \sum_{i=(m-1)k+1}^{mk} w_i x_i & mk = d \end{cases} \quad (5)$$

In BCFE the final weight vector which is used in formula (4), is calculated as a linear combination of two different weight vectors which are defined as w_1 and w_2. To calculate w_1, first the class means and the average of features in each class mean should be computed. Then the binary coded class mean for each class should be calculated as formula (6).

$$\begin{cases} b_{ij} = 0 & \mu_{ij} < \mu_j' \\ b_{ij} = 1 & \mu_{ij} \ge \mu_j' \end{cases} \quad 1 < i < d, \ 1 < j < c \quad (6)$$

Where, μ_{ij} is the ith feature in jth class mean and μ_j' is the average of features in jth class mean. b_{ij} is the binary coded of the ith feature in jth class mean. c is the number of classes in the dataset.

By calculating the binary coded class means for all of the training samples in each class, we can obtain a $c \times d$ matrix. Each of the columns of the binary coded class means matrix is the corresponding value of each class mean in related feature. Then the difference between number of zeros and ones is calculated for each column of binary coded class means matrix. The column corresponding to lowest value of the difference between zeroes and ones, should gain higher value of weight. That is because, the lower difference between number of zeros and ones implies more discriminant information. Thus the features with more discriminant information should gain more weights. The weight corresponding to feature i is calculated as formula (7).

$$w_{1i} = \frac{1}{|numberof(0) - numberof(1)| + 1} \quad i < 1 < d \tag{7}$$

After calculating the first weight vector, the second one should be computed. To obtain the second weight vector, the information in the edges of binary coded class means matrix is used. These edges are categorized into 3 parts: 1-positive edge 2-negative edge and 3-no edge. Suppose $b_k(i)$ is the binary coded value in class mean k feature i. Since the type of edges in each class mean should be found, we define positive edges and negative edges as below formulas.

Case(1): $2 < i < d - 1$

$$\text{Negative edge} = \begin{cases} b_k(i) - b_k(i-1) = -1 \\ b_k(i) - b_k(i+1) = +1 \end{cases} \tag{8}$$

$$\text{Positive edge} = \begin{cases} b_k(i) - b_k(i-1) = +1 \\ b_k(i) - b_k(i+1) = -1 \end{cases} \tag{9}$$

Case(2): $i = 1$

$$\begin{cases} b_k(i) - b_k(i+1) = -1 & Positive\ edge \\ b_k(i) - b_k(i+1) = +1 & Negative\ edge \end{cases} \tag{10}$$

Case(3): $i = d$

$$\begin{cases} b_k(i) - b_k(i-1) = +1 & Positive\ edge \\ b_k(i) - b_k(i-1) = -1 & Negative\ edge \end{cases} \tag{11}$$

In all cases if the value of subtraction is equal to zero, then it is the no edge situation. The number of positive edges ($npe(i)$) and the number of negative edges ($nne(i)$) in feature i of the binary coded class means matrix should be calculated. Then, the second weight vector will be obtained according to formula (12).

$$w_{2i} = \frac{1}{std[npe(i), nne(i)] + 1} \quad i < 1 < d \tag{12}$$

Where std denotes the standard deviation. Then the final weight vector will be calculated as below:

$$w_i = \alpha w_1(i) + (1 - \alpha) w_2(i) \tag{13}$$

α is a number in the range of 0 and 1 which will be tuned during the training process. Finally, we use formula (4) to calculate feature vectors in the new reduced space. Although BCFE is a useful feature extraction algorithm especially in small sample size problem (SSS), there are some problems with BCFE that may reduce the classification accuracy. Later in this paper we will discuss drawbacks of BCFE and propose our ELM-RBCFE algorithm.

III. THE PROPOSED ELM-RBCFE ALGORITHM

In the proposed algorithm we have tried to maintain the efficient aspects of BCFE and simultaneously address the problems that it has engaged with. ELM-RBCFE can be classified into 3 parts. We will discuss each part sequentially and comprehensively.

A. Determining the number of features

In section 2 we have introduced the ELM algorithm completely. Now we will discuss the usage of this algorithm in our proposed algorithm. Since the number of features affects the classification accuracy intensely, determining the optimum number of features is of a great issue. In [17], the ELM algorithm has been used to determine the number of features. Indeed, before feature extraction step, the whole dataset is classified by ELM, then the ELM will be tested on data with different number of neurons. Finally, we can extract two kinds of information out of this classification. First, the number of neurons corresponding to maximum accuracy of classification is considered as the number of features that should be extracted. Second, the number of misclassified data in each class should be computed. In this paper it has been considered that the class with higher number of misclassified data needs more attention in feature extraction step. Indeed, the number of misclassified data in each class is considered as a coefficient that determines the level of importance of each class. Then these coefficients will be normalized. Coefficients are denoted as $q(c)$ (c is the number of classes in dataset). We will get back to this later in part *C* of current section.

B. Using between class information

After calculating the number of features and coefficients of each class, we have to calculate the weights. In our proposed algorithm the weights are calculated based on between class and within class information. To obtain weights from between class information, we have used the BCFE algorithm with only one minor change which is the way we have calculated μ'. In our proposed method μ' is calculated as formula (14).

$$\mu' = \frac{1}{c} \sum_{i=1}^{c} \mu_i \tag{14}$$

Where c is the number of classes and μ_i is The mean of class(i). Then the first weight vector is calculated just like BCFE algorithm. The next step is to calculate the weights based on within class information.

C. Using within class information

To use the within class information, first the class means are used to obtain binary coded data points. Let's consider $t_{jc}(i)$ as the value of data j in class c and feature i and $P_{jc}(i)$ is the corresponding binary coded value. Then $P_{jc}(i)$ is calculated by formula (15).

$$P_{jc}(i) = \begin{cases} 1 & t_{jc}(i) \geq \mu_c(i) \\ 0 & t_{jc}(i) < \mu_c(i) \end{cases} \quad 1 \leq i \leq d \tag{15}$$

$\mu_c(i)$ is the ith feature of cth class mean. After calculating the binary coded values of each data point, the difference between the number of zeros and ones for each feature in each binary coded class should be calculated. Features should be weighted in the way that features with higher number of differences will gain greater weights. Because higher values of differences in each feature in an arbitrary class means strong similarity between data points of a same class. Indeed, by using between class information the aim is to increase differences between data points belonging to different classes and by using within class information the goal is to increase the similarity

between data points belonging to the same class. The weights are calculated as below.

$$w_c(i) = |\#zeros(i,c) - \#ones(i,c)|, \quad 1 \le i \le d \qquad (16)$$

Where $\#zeros(i,c)$ is the number of zeros in class c feature i and $\#ones(i,c)$ is the number of ones in class c feature i. $w_c(i)$ denotes the weight value for feature i in class c. Note that w_c should be normalized. Now the weight vector obtaining from within class information is calculated as formula (17).

$$w_{within}(i) = \sum_c q(c).w_c(i) \quad ,1 \le i \le d \qquad (17)$$

Then the final weights can be calculated as a linear combination of w_{within} and $w_{between}$ where w_{within} is the weight vector obtained from within class information and $w_{between}$ is the weight vector obtained from between class information. The final weight vector is calculated as formula (18).

$$w(i) = \Delta.w_{within}(i) + (1-\Delta).w_{between}(i) \qquad (18)$$

In formula (18) Δ is in the range of 0 and 1. Finally, the weights obtained from formula (18) will be used in formula (4) to find new feature vectors in a space with less dimensions and complexity. The proposed algorithm is illustrated in Table 1.

TABLE I.
THE PROPOSED ALGORITHM

1- Using ELM to determine the number of features that should be extracted and each class coefficients.
2- Calculate $w_{between}$ according to BCFE.
3- Calculate w_{within} according to within class information.
4- Obtain the final weight vector according to formula (18).
5- Calculate the extracted features according to formula (4)
6- Use SVM to classify dataset.

IV. RESULTS

To testify the algorithm, we have used multiple of standard datasets from UCI datasets [18]. To evaluate the results of implementation of ELM-RBCFE, we have used the results of implementation of other feature extraction methods like BCFE, PCA and LDA on the same datasets. After implementing each of the mentioned feature extraction methods, we have used SVM to classify the dataset. The results of the proposed algorithm and other mentioned methods in this paper are demonstrated in Table 2. Also the comparison of experimenting those algorithms on different datasets are illustrated in Fig. 1. The bolded numbers indicate the maximum value of accuracy. The numbers in the parentheses indicate the numbers of extracted features.

TABLE II.
RESULTS OF IMPLEMENTING ELM-RBCFE AND OTHER MENTIONED METHODS

DATASETS	LDA	PCA	BCFE	ELM-EBCFE
AUSTRALIAN	80.07(5)	79.71(5)	79.53(5)	**81.77(8)**
INDIAN DIABETES	79.01(6)	75.70(5)	77.06(5)	**79.84(5)**
SPECT	83.40(7)	63.75(7)	60.72(7)	**88.50(8)**
SPECTF	75.01(6)	66.25(6)	75.22(6)	**86.53(7)**
HEART	**81.90(7)**	77.78(7)	79.27(7)	80.49(9)

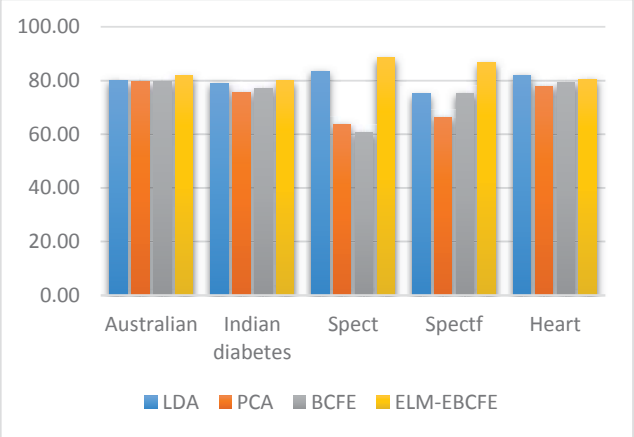


Fig. 1. Comparison of different feature extraction methods on various datasets.

V. CONCLUSION

Although BCFE algorithm has many advantages over other feature extraction algorithms, but the algorithm contains two major drawbacks which are: (1) the number of features that should be extracted remains unspecified and (2) the within class information has not been used to obtain the final weight vector. To address the first problem, we have used an ELM algorithm to classify the dataset and to determine the number of features that should be extracted. To deal with the second problem, we have used the within class information by obtaining the binary coded values of training samples in each class of dataset. The results obtained from proposed algorithm demonstrates considerable improvements in BCFE algorithm.

REFRENCES

[1] Griffin, Julian L. "Metabonomics: NMR spectroscopy and pattern recognition analysis of body fluids and tissues for characterisation of xenobiotic toxicity and disease diagnosis." *Current opinion in chemical biology* 7.5 (2003): 648-654.

[2] Polat, Kemal, and Salih Güneş. "Breast cancer diagnosis using least square support vector machine." *Digital Signal Processing* 17.4 (2007): 694-701.

[3] Kubat, Miroslav, Robert C. Holte, and Stan Matwin. "Machine learning for the detection of oil spills in satellite radar images." *Machine learning* 30.2-3 (1998): 195-215.

[4] Jolliffe, Ian. *Principal component analysis*. John Wiley & Sons, Ltd, 2002.

[5] Scholkopft, Bernhard, and Klaus-Robert Mullert. "Fisher discriminant analysis with kernels." *Neural networks for signal processing IX* 1.1 (1999): 1.

[6] Roweis, Sam T., and Lawrence K. Saul. "Nonlinear dimensionality reduction by locally linear embedding." *Science* 290.5500 (2000): 2323-2326.

[7] Niyogi, X. "Locality preserving projections." *Neural information processing systems*. Vol. 16. MIT, 2004.

[8] de Ridder, Dick, et al. "Supervised locally linear embedding." *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003*. Springer Berlin Heidelberg, 2003. 333-341.

[9] Polito, Marzia, and Pietro Perona. "Grouping and dimensionality reduction by locally linear embedding." (2002): 1255-1262.

[10] Kouropteva, Olga, Oleg Okun, and Matti Pietikäinen. "Incremental locally linear embedding." *Pattern recognition* 38.10 (2005): 1764-1767.

[11] Pan, Yaozhang, Shuzhi Sam Ge, and Abdullah Al Mamun. "Weighted locally linear embedding for dimension reduction." *Pattern Recognition* 42.5 (2009): 798-811.

[12] Yu, Weiwei, Xiaolong Teng, and Chongqing Liu. "Face recognition using discriminant locality preserving projections." *Image and Vision computing*24.3 (2006): 239-248.

[13] Zhu, Lei, and Shanan Zhu. "Face recognition based on orthogonal discriminant locality preserving projections." *Neurocomputing* 70.7 (2007): 1543-1546.

[14] Imani, Maryam, and Hassan Ghassemian. "Binary coding based feature extraction in remote sensing high dimensional data." *Information Sciences*(2016).

[15] Huang, Guang-Bin, Qin-Yu Zhu, and Chee-Kheong Siew. "Extreme learning machine: theory and applications." *Neurocomputing* 70.1 (2006): 489-501.

[16] Bartlett, Peter L. "The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network." *Information Theory, IEEE Transactions on* 44.2 (1998): 525-536.

[17] Duan, Lijuan, et al. "Feature Extraction of Motor Imagery EEG Based on Extreme Learning Machine Auto-encoder." *Proceedings of ELM-2015 Volume 1*. Springer International Publishing, 2016. 361-370.

[18] Lichman, Moshe. "UCI machine learning repository." University of California, Irvine, School of Information and Computer Sciences (2013).

Fuzzy-logic Controller for Brushless DC PM Motor

Martin Moreno Guzmán[1], Antonio Hernández Zavala[1], Iván Domínguez López[2] and Rodolfo Orosco Guerrero[3]
[1]Mechatronics Department, Instituto Politécnico Nacional - CICATA, Querétaro, México
[2]Materials Department, Instituto Politécnico Nacional - CICATA, Querétaro, México.
[3]Electronics Department, Instituto Tecnológico de Celaya, Celaya, Guanajuato, México.

Abstract - *This paper describes a speed controller for brushless motor. The driver was electronically designed and constructed by the authors to control the speed of a commercial brushless motor. The authors considered the error and delta error as inputs, and only one output named as delta output, these inputs and outputs were communicated between a 32-bit and an 8-bit microcontroller by serial peripheral protocol. Each input and output has five-membership functions. Inference mechanism has 25 rules and as defuzzification the authors used the Center of Gravity method. The brushless motor behavior was studied for two different set points. A comparison between the real plant versus simulations using fuzzy-logic control is presented.*

Keywords: *Brushless DC motor, fuzzy-logic control, real time systems, speed controller.*

1 Introduction

The demand of direct current machines, which are more efficient, faster and with high performance, provoke the proliferation of using electrical machines like brushless motor. Brushless motors have some advantages over other DC machines [1]. The most significant advantage is that they require lower maintenance due to the mechanical commutator that was deleted.

It is important to emphasize that the disadvantage of this motors is the complexity for controlling it. Given that, brushless motor requires two feedbacks sources. The fist source is for synchronizing the commutation (Hall Effect Sensors); and the second source is required for speed controlling (Encoder). Theoretically the speed is proportional to the primary DC supply voltage and the commutation can be controlled by three half bridges in parallel working as an inverter.

Several papers [3]-[10] have described how to model and control a brushless DC motor. However, they only present simulation results. In these sense, building a driver for a brushless motor involves a complex work.

The advantages of simulating a fuzzy-logic controller with another technique, such as adaptive-fuzzy-PID or adaptive-fuzzy control was made by R. Kandiban [3] and S. Yuan [9], respectively. They can control higher speed than with a fuzzy-logic controller [4]-[8], [10]. Even though, it generates a bigger overshoot.

The authors in [4]-[8], [10] did not present a maximum overshoot. J. Sriram [7], showed the biggest settling time, the reason is that his controller was based on the Back-EFM of the BLDC motor, which takes more computational time and resources.

It is known that motor control at low speeds is the most difficult work, but C. Kwon [10] presented a control which had a one millisecond settling time. It could be possible only in simulation. These papers fail to consider all possible variables that involve the brushless response, because simulation is not enough for knowing the performance of a brushless motor in real.

The paper is organized as follows: Section 2 has an explanation of the generalities of brushless motors, and presents the driver circuit. Section 3, describes the proposed fuzzy-logic controller and its principal characteristics. Section 4, explains how does the system was embedded in a 32-bit microcontroller, and how communicate with our driver. The response of the BLDC motor for different experiments is shown in section 5, where a comparison of our results with those from similar works is made. Finally, conclusion and future work are given.

2 Brushless motor

The basic structure of a brushless motor is that it has a permanent magnet (PM) rotor and wound stator. It is supplied with direct current using an inverter [11]. Another characteristic of this motor is that it also has three Hall effect sensors. They are separated 120 degrees and are used as feedback signal, indicating the rotor position with respect to the stator.

2.1 Brushless DC Motor system

A brushless DC motor is modeled as the equivalent circuit shown in Figure 1. Each phase has an inductance, a resistance and a back-EMF [2]. It is connected to an inverter. The inverter consists of six power switches (M1, M2, M3, M4, M5 and M6), which make three half-bridge structures in

parallel. Each phase of the brushless motor is connected in the middle of each half-bridge structure.

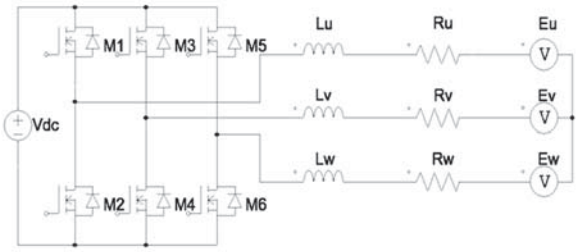


Figure 1. The electrical equivalent circuit

For modeling this system, we suppose that all inductances, resistances and back-EMFs are constant. The voltage and current flows through two phases while the other is floating. Equation 1 is the model of the brushless motor and gives the voltage of each pair of phases.

Where V_{uv}, V_{vw} and V_{wu} are of couple-phase voltages; L_u, L_v and L_w are the phase inductances; R_u, R_v and R_w are phase resistances; E_u, E_v and E_w are the phase back-EFMs. i_u, i_v and i_w are the phase currents.

$$\begin{bmatrix} V_{uv} \\ V_{vw} \\ V_{wu} \end{bmatrix} = \begin{bmatrix} R_U & -R_V & 0 \\ 0 & R_V & -R_W \\ -R_U & 0 & R_W \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix}$$

$$+\frac{d}{dt}\begin{bmatrix} L_u - L_v & L_v - L_u & 0 \\ 0 & L_v - L_w & L_w - L_v \\ L_u - L_w & 0 & L_w - L_u \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} \quad (1)$$

$$+\begin{bmatrix} E_u - E_v \\ E_v - E_w \\ E_w - E_u \end{bmatrix}$$

2.2 Brushless commutation

The six-step commutation scheme for the motor motion is presented in Table 1. It indicates the rotor position by Hall U, Hall V and Hall W. Hall-Effect sensors provide a logic value to determine which gate of the MOSFET transistor will be turned on. U High and U low corresponds to M1 and M2 MOSFET, as in Figure 1. The high side of the inverter corresponds to M1, M3 and M5. The low side corresponds to M2, M4 and M6 MOSFET.

In the first case when Hall U, Hall V and Hall W gives a 1, 0 and 0, respectively; the MOSFET transistors M1 and M6 are ON. The current flow through the MOSFET transistors provokes the first step in the motor. By continuing systematically the sequence of Table 1, it will generate the constantly motion of the rotor. The commutation of a brushless motor requires a driver that has this table embedded by software or hardware.

Table 1. Six-Step BLDC commutation

HALL U	HALL V	HALL W	U HIGH	U LOW	V HIGH	V LOW	W HIGH	W LOW
1	0	0	1	0	0	0	0	1
1	1	0	0	0	1	0	0	1
0	1	0	0	1	1	0	0	0
0	1	1	0	1	0	0	1	0
0	0	1	0	0	0	1	1	0
0	1	1	1	0	0	1	0	0

2.3 Drive system

The complete block diagram of the BLDC motor controller is shown in Figure 2. The system has two control loops. The inner loop refers to the sequence of the inverter gate signals according to Table 1, using Hall-Effect signals as a feedback. The outer loop controls the motor speed by a PWM signal and uses an incremental encoder as feedback.

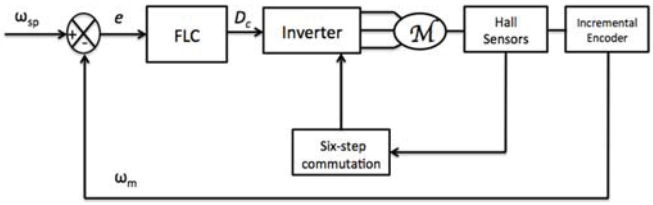


Figure 2. Block diagram of the system

3 The Proposed Fuzzy-Logic Controller

The fuzzy-logic controller uses error and delta error as inputs. The error is obtained by the subtraction of ω_{sp} (Set-point) and ω_m (The encoder measure); delta error is the gradient of error, and is calculated by: *delta error= error(n)-error(n-1)*. The FLC consists of three stages: fuzzification, inference mechanism and defuzzification, which will be described in the following.

3.1 Fuzzification

This stage performs the conversion from crisp inputs to fuzzy inputs as consequence of evaluating a crisp value into a membership function.

For our case, each input and output was defined by means of five membership functions. Each function has an

associated linguistic value as: Negative Large (NL), Negative Small (NS), Zero (ZE), Positive Small (PS) and Positive Large (PL). Error input is graphically presented in Figure 3; Delta error input is presented in Figure 4; and the output is presented in Figure 5.

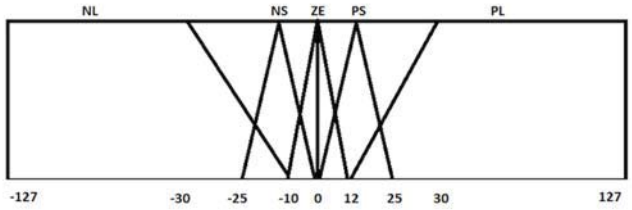


Figure 3. Membership function for error

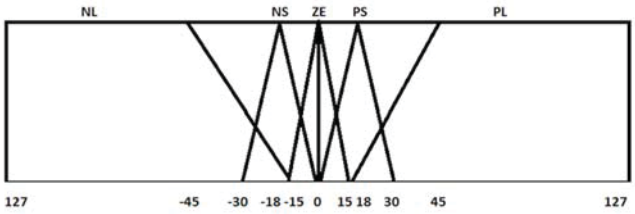


Figure 4. Membership function for delta error

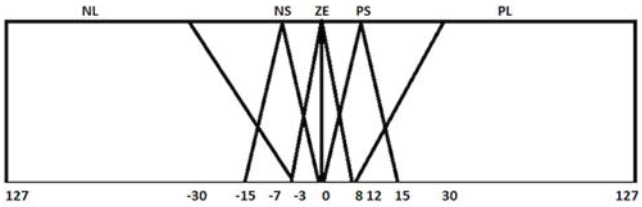


Figure 5. Membership function for output

3.2 Inference mechanism

The inferred fuzzy value is determined by interacting with the rule base. The rule base is defined by a set of rules in the form:

IF (condition 1) AND (condition 2) THEN (conclusion) (3)

The number of rules used for this controller is 25. The inference method chosen is the Max-Min method, which was proposed by Mamdani [12].

Equation 4 determines when an error value (condition 1) and a delta error value (condition 2) are in a specific membership function. As a result, the equation gives the minimum value of both inputs. The output value (consequent) will represent a cut in a membership function.

In the case is when error and delta error are the linguistic value NL, the inference uses the minimum of both linguistic values. The minimum value will cut the output membership function in NL. Table 2 shows the rule base used. It contains the input and output variables relation.

$$\begin{aligned} DOF &= \mu_{NL}(error) \wedge \mu_{NL}(\Delta error) = \mu_{NL}\left(\hat{e}\right) \\ DOF &= \mu_{NL}(error) \wedge \mu_{NS}(\Delta error) = \mu_{NL}\left(\hat{e}\right) \\ &\vdots \\ DOF &= \mu_{PL}(error) \wedge \mu_{PL}(\Delta error) = \mu_{PL}\left(\hat{e}\right) \end{aligned} \quad (4)$$

Table 2. Fuzzy Rule-Base

e / Δe	**NL**	**NS**	**ZE**	**PS**	**PL**
NL	NL	NL	NL	NS	ZE
NS	NL	NL	NS	ZE	PS
ZE	NL	NS	ZE	PS	PL
PS	NS	ZE	PS	PL	PL
PL	ZE	PS	PL	PL	PL

3.3 Defuzzification

The last stage in a fuzzy system consists in converting the resulting output fuzzy value into a crisp one. There are several methods for realizing the defuzzification. The most common and widely use is the Center of Gravity, which is defined by equation (5) for the case of discrete time systems.

$$y_{COG} = \frac{\sum_{i=1}^{k} x_i \mu(x_i)}{\sum_{i=1}^{k} \mu(x_i)} \quad (5)$$

4 Implementation of the Embedded System

The proposed fuzzy controller was implemented in an Atmel SAM3X8E ARM Cortex-M3 CPU, which is a 32-bit microcontroller. The whole fuzzy controller was implemented in the SAM3X8E. It has as disadvantage that only manages 3.3v input-output voltages. For these reason, an 8-bit microcontroller was necessary to handle 5v input-output voltages. The communication is made by serial peripheral protocol SPI, in which the SAM3X8E is the master and the

ATMega32 is the slave. SPI protocol uses four logic signals SCKL, MOSI, MISO and SS.

The algorithm for realizing the controller is as follows:

1. SAM3X8E reads the frequency of the incremental encoder and calculates the speed of the BLDC motor.
2. The SAM3X8E uses the previous information as input for the fuzzy-logic controller.
3. A PWM-output signal is obtained from the FLC, which is related to the required speed.
4. The SAM3X8E sends this PWM information to the ATMega 32.
5. The PWM-output ATMega 32 connects to input of the inverter. The driver realizes the Six-Step commutation for causing a speed variation in the BLDC motor.
6. Repeat from 1

The brushless motor was a KOLLMORGEN model AKM21E whose main characteristics are as follows:

Table 3. Characteristics of BLDC motor

Max voltage	**Vdc**	**160**
Continuous current	Arms	3.11
Peak Current	Arms	12.4
Peak Torque	Nm	0.48
Max mechanical speed	Rpm	6000

5 Results

To evaluate the performance of the proposed controller, we considered as metrics the rise time t_r, maximum overshoot M_p and the setting time t_s.

In comparison to other proposals presented in literature, they only present simulation results. Consequently, they exhibit ideal metrics as for rise time than for setting time. Unfortunately, in simulation, they do not consider external factors as static friction, viscous damping, thermal resistance, brake inertia, or field electromagnetics.

Figure 6 shows the controller response for an 850-rpm set point. Figure 7 shows the response for the case of 500-rpm. Both cases uses the AKM21E BLDC motor, and the constructed driver.

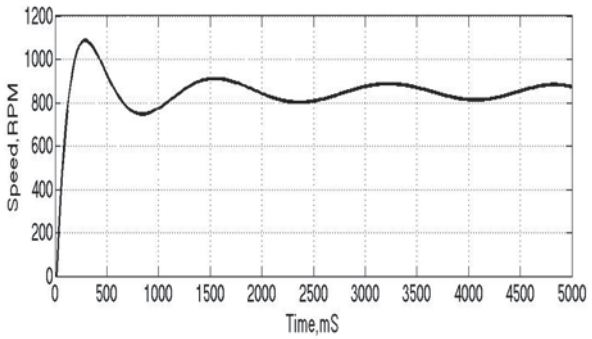


Figure 6. Motor response in a Set point of 850 RPM

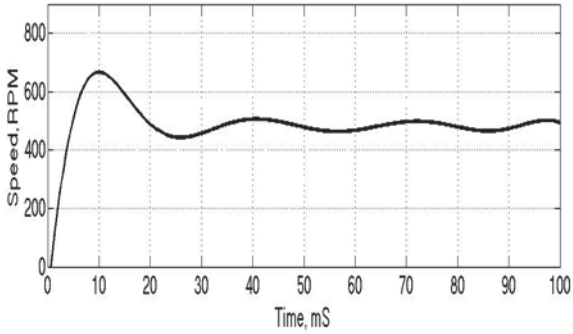


Figure 7. Motor response in a Set point of 500 RPM

Table 4 resumes the performance metrics of our system, and compares them to those obtained by other authors. It is noteworthy to mention that the only case that uses a real motor is ours. The rest of the compared works presented were realized only in simulation.

Table 4. Comparison of different FLC speed controllers.

Authors	**Control**	**RPM**	**t_r (s)**	**M_p(%)**	**t_s(s)**	**Test**
M.Moreno	FLC	850	0.3	34.1	3	Real
M.Moreno	FLC	500	0.007	34	0.4	Real
R.Kandiban[3]	Adaptive Fuzzy-PID	1500	0.020	16.5	0.35	Sim
M.N.Uddin[4]	FLC	1910	-	0	0.1	Sim
H.S.EL-Sayed [5]	FLC	600	-	0	0.25	Sim
Sreekala.P [6]	FLC	500	-	0	0.2	Sim
J.Sriram [7]	FLC	1400	-	0	2.5	Sim
Shang-zun Yuan [8]	Genetic with FLC	500	.	0	0.3	Sim
Changliang Xia [9]	Adaptive FLC	2100	0.005	4.76	0.015	Sim
Chung-Jin Kwon[10]	FLC	100	-	0	0.001	Sim

6 Conclusion

There exists a big difference between the real data obtained with our proposal when compared to the existing ones. It is mainly due to the inherent variations induced by the environmental variables, which were not considered in simulations. As this is a nonlinear case, our performance metrics (t_r, M_p and t_s), were higher than those from simulations.

One of the presented phenomena is that a simulation does not have to break the inertia for the BLDC to begin rotating. Another issue is that big speed increases, provokes big overshooting and consequently a longer setting time and bigger oscillations.

As a future work the authors proposed two goals: fist goal is to tune the fuzzy-logic controller for obtaining better metrics (t_r, M_p and t_s). Second goal is applying different loads for knowing how the fuzzy-logic controller responds.

7 Acknowledgment

Authors would like to thank Instituto Politécnico Nacional and CONACYT, Mexico, for their financial support.

8 References

[1] P. Pillay and R. Krishnan, "*Modeling, simulation, and analysis of permanent-magnet motor drives, part ii: The brushless dc motor drive*", *IEEE Trans. Ind. Appl.*, Vol. 25, No. 2, 274–279, Mar./Apr. 1989.

[2] R. Shanmugasundram, K. M. Zakariah, and N. Yadaiah, *"Low-cost high performance brushless dc motor drive for speed control applications"* , IEEE Int. Conf. Adv. Recent Technol. Commun. Comput., Kottayam, India, Oct. 27–28, 2009, pp. 456–460.

[3] R.Kandiban and R. Arulmozhiyal, *"Speed control of BLDC motor using Adaptive Fuzzy-PID controller"*, International Conference on Modeling, Optimization Comput.,2012, pp.306-313.

[4] M.N. Uddin and M.A. Rahman, *"Fuzzy logic based speed control of an IPM synchronous motor drive"*, IEEE Canadian Conference on Electrical and Computer Engineering, May.1999. pp.1259-1264.

[5] H.S. El-Sayed, F.M. El-Khouly, M.N. Khater and A.M. Osheiba, *"Fuzzy logic based speed control of a permanent magnet brushless DC motor drive"*, International Conference on Electrical Engineering, April, 2007. pp.1-6.

[6] P.Sreekala and A. Sivasubramanian, "*Speed control of brushless DC motor with PI and fuzzy logic controller using resonantpole inverter*", Innovative Smart Grid Technologies - India (ISGT India), Dec. 2011.pp. 334-339.

[7] J. Sriram and K. Sureshkumar, *"Speed Control of BLDC Motor Using Fuzzy Logic Controller Based on Sensorless Technique"*, International Conference Green Computing Communication and Electrical Engineering (ICGCCEE), March 2014.pp. 1-6.

[8] S.Yuan, L.Song, P. Du and Y. Wang, *"Brushless DC Motor Speed Fuzzy Adaptive Control System"*, Fuzzy Info. and Eng., Volume 2, AISC 62, pp. 1229-1236.

[9] C. Xia, P.Guo, T. Shi and M. Wang, "Sp-eedControl of Brushless DC Motor Using Genetic Algorithim Based Fuzzy Controller", International Conference on Intelligent Mechatronics and Automation,. Proceedings. Aug. 2004. pp. 460-464.

[10] C. Kwon, W. Han, S. Kim and C. Lee, *"Speed Controller with Adaptive Fuzzy Tuning For BLDC Motor Drive Under Load Variations",* SICE 2003 Annual Conference (Volume:3), Aug. 2003.pp. 3118 – 3121.

[11] A. Rubai, A. Ofoli, and M. Castro, *"dSPACE DSP-Based Rapid Prototyping of Fuzzy PID Controls for High Performance Brushless Servo Drives"*, IEEE Industry Applications Conference, 41st IAS Annual Meeting, page(s):1360–1364, 2006

[12] E. H. Mamdani, *"Application of fuzzy algorithms for the control of a dynamic plant"*, *Proc. IEEE*, vol. 121, no. 12, Dec. 1974, pp. 1585– 1588.

A new Fuzzy MultiGrouped Particle Swarm Optimization Algorithm

Maziar Moradi Fard
Department of computer science and electrical engineering
Islamic Azad University of Science and Research
Tehran, Iran
maziar.moradifard@srbiau.ac.ir

Hossein Ghomeshi
Department of Engineering and Technology
University of Tehran, Kish International Campus
Kish, Iran
h.ghomeshi@ut.ac.ir.com

Abstract—Nowadays, the role of optimization in every aspect of our daily life is undeniable. The problem of finding global optimum point has been one of the most challenging problems in the context of optimization over the past decades. This problem grows intentionally in optimizing multimodal functions. Several approaches have been developed to overcome multimodal function optimization problems especially premature convergence problem. In this paper a new fuzzy multigrouped particle swarm optimization (FMGPSO) algorithm has been proposed. The aim of the proposed algorithm is to overcome multimodal optimization problems. Proposed algorithm has been examined on several benchmark functions. Experiments show that FMGPSO can perform better in finding global maxima or minima in comparison with multigrouped particle swarm optimization (MGPSO).

key words: Particle swarm optimization, Fuzzy, Optimization, Parameter adaptation.

I. INTRODUCTION

Since Fuzzy logic has been proposed in 1965 by A. Zadeh [1], it has challenged the classic definition of mathematics by introducing the fuzzy inference systems. In the classic definition of mathematics, the membership of an object to a specific set is considered as 0 or 1. As opposed to this definition, fuzzy logic considers the membership of an object to a set as a continuous value in the range of 0 and 1. Fuzzy logic has been used in several fields, from decision and control systems [2] to spatial load forecasting [3] and neural networks [4]. As it will be stated later in this paper, fuzzy plays an important role in the proposed algorithm.

Particle Swarm Optimization (PSO) was proposed in 1995 [5]. In this algorithm a set of random particles will be generated. These particles will be updated according to their personal best positions and global best positions. PSO is one of the most popular algorithms in optimization. PSO has lots of advantages over other optimization algorithms which makes it one of the most popular approaches to use in different fields [6-8]. Although PSO is one of the most attracting methods, in some conditions this algorithm fails to find global optimum point. One of the major problems of PSO is getting trapped in local minima or maxima when the task is to find global optimum point of a multimodal function. This problem is addressed as premature convergence. Several extensions of PSO have been proposed in order to overcome this drawback [9-11]. One of the most important algorithms that was proposed to overcome premature convergence of PSO is Multigrouped PSO (MGPSO) [12]. MGPSO was proposed to overcome multimodal function optimization. In this algorithm several groups of particles will be generated randomly. In MGPSO, besides personal best position and global best position, some extra criteria have been defined to update each particle's velocity. In MGPSO, just like other optimization algorithms, the strategy is first exploring then exploiting but in a different way. This strategy involves fast convergence for particles in the same group and simultaneously making particles belonging to the same group to get away from the global best positions of the other groups. This strategy prevents particles from merging and converging to a single point. There are two drawbacks that MGPSO algorithm is engaged with. First, the parameters that have been defined for updating particles, remain constant for the entire process. The second problem that associates with the strategy of the algorithm, makes all of the particles that belong to the same group to get away from global best positions of other groups during the entire process of algorithm. To address these problems, in this paper we introduced a new Fuzzy MGPSO (FMGPSO) algorithm. In FMGPSO, all of the parameters that are used for updating the velocity are adjusted by using fuzzy inference systems in the way that in each iteration the values of the parameters are calculated according to two independent fuzzy inference systems. Moreover, a new strategy for updating the particles have been presented. Similar to MGPSO, our proposed algorithm looks for exploring and then exploiting, but particles of a same group do not get away from global best positions of other groups statically. Instead, in the infancy of the algorithm particles in the same group will make to get away from global best position of the nearest group but as the algorithm proceeds, they will be able to get closer to the global best position of the nearest group. The idea behind this strategy is that there's a great chance to search the entire space. This algorithm will be described in details below. The rest of the paper is organized as follows. In section 2 we will review PSO and MGPSO algorithms. In section 3 we will present our proposed FMGPSO algorithm. Finally, in section 4 the results of implementation of the proposed algorithm and MGPSO will be demonstrated. The results of FMGPSO show an intense improvement in comparison with MGPSO algorithm in both unimodal and multimodal cases.

II. REVIEW OF PSO AND MGPSO

Since PSO is a basis for either MGPSO or FMGPSO, in this section, we first describe PSO in details, then we will review MGPSO formula and strategy.

A. PSO algorithm

As mentioned in section 1, PSO is one of the most important algorithms in the context of optimization. The nature of the algorithm is based on movements of birds such, some random particles will be generated and each particle will look forward the global optimum according to two criteria:

- Each particle personal best position.
- Global best position of all of the particles.

The value of coefficient corresponding to personal best position (cognitive parameter) controls the importance of personal behavior and the value corresponding to global best position (social parameter) controls the level of importance of social behavior. It is obvious that as the value of cognitive parameter increases, the algorithm tends to behave like a stochastic search, and vice versa, as the value of social parameter increases, the algorithm will face fast convergence or even premature convergence. In PSO, every particle will be updated according to the formula presented below.

$$V_{ij}^{k+1} = V_{ij}^{k} + c_1 r_1 (x_{pbest(ij)}^{k} - x_{ij}^{k}) + c_2 r_2 (x_{gbest(j)}^{k} - x_{ij}^{k}) \quad (1)$$

$$x_{ij}^{k+1} = x_{ij}^{k} + V_{ij}^{k+1} \quad (2)$$

x_{ij}^{k}: Position of particle i in group j, round k.

V_{ij}^{k+1}: Velocity of particle i in group j, round $k+1$.

$x_{pbest(ij)}^{k}$: Personal best position of particle i in group j, round k.

$x_{gbest(j)}^{k}$: Global best position of group j, round k.

One of the major drawbacks of this algorithm is that the optimal values of social parameters and cognitive parameters are hard to find. Consequently, one may find these optimal values by testing multiple of values for these parameters. The other problem associated with this algorithm is the problem of premature convergence in multimodal function optimization. To address this problem, MGPSO was proposed. Below we will describe this method in details.

B. MGPSO algorithm

As mentioned previously, MGPSO is one of the possible solutions to the problem of premature convergence that PSO is engaged with. One of the most important problems that PSO faces is the premature convergence problem. Many researchers developed new algorithms for conquering this problem. Some of them used some technics of genetic algorithms like mutation to overcome the premature convergence [13-14]. The major drawbacks of these algorithms include first they are time consuming and second, they change the nature of the PSO drastically. But through MGPSO the nature of PSO is preserved.

To make sure that the algorithm will not get trapped in a local optimum point, some groups of particles will be generated randomly and each group looks for global optimum point independently. It is clear that if groups merge together, again the problem of getting trapped in a local optimum point will appear. To avoid such problem, the current position of particles of a same group will be subtracted from the position of global best positions of the other groups. This strategy ensures us that global best position of each group is not able to attract particles belonging to other groups. The MGPSO formula is presented below.

$$V_{ij}^{k+1} = V_{ij}^{k} + c_1 r_1 (x_{pbest(ij)}^{k} - x_{ij}^{k}) + c_2 r_2 (x_{gbest(j)}^{k} - x_{ij}^{k}) + c_3 r_3 (x_{ij}^{k} - x_{gbestOthers}^{k}) \quad (3)$$

$$x_{ij}^{k+1} = x_{ij}^{k} + V_{ij}^{k+1} \quad (4)$$

$x_{gbestOthers}^{k}$: Global best positions of other groups.

There are two major problems associated with MGPSO:

- All of the parameters are constant.
- Making particles of a same group to get away from global best positions of other groups is not the best strategy.

To avoid the problems described above, we have proposed FMGPSO algorithm. We will describe this algorithm below.

III. PROPOSED ALGORTIHHM

In this section, our goal is to present a new algorithm which is able to avoid the problems that MGPSO faces. Below we will describe the algorithm in three phases.

A. Initializing the algorithm (phase one)

In the beginning of the algorithm, some of the parameters should be defined or calculated. Number of groups and number of particles in each group should be initialized. The initial value of each particle and velocities should be randomly defined then the personal best positions should be determined. The global best position of each group should be calculated with respect to the maximum value of personal best positions of particles pertaining to the same group.

B. FMGPSO proposed formula (phase 2)

FMGPSO formula is pretty much the same as MGPSO. Some minor changes have been made to MGPSO formula.

$$V_{ij}^{k+1} = V_{ij}^{k} + c_1 r_1 (x_{pbest(ij)}^{k} - x_{ij}^{k}) + c_2 r_2 (x_{gbest(j)}^{k} - x_{ij}^{k}) + c_3 r_3 (x_{ij}^{k} - x_{gbestNearest}^{k}) \quad (5)$$

$$x_{ij}^{k+1} = x_{ij}^{k} + V_{ij}^{k+1} \quad (6)$$

$x_{gbestNearest}^{k}$: Global best position of the nearest group to the group j.

The only change that have been considered toward formula (3) is that instead of making particles of a same group to get away from global best positions of other groups, they will get away from global best position of the nearest group. The reason is, the strategy that has been considered for MGPSO may cause groups to move around corners of the search space. Besides, in all of the optimization algorithms, the aim is to search the whole of the search space. But in formula (3) by considering such strategy, we may lose the chance for searching the entire search space.

As mentioned before, all the parameters in formula (5) will be adjusted dynamically through the fuzzy inference systems. This parameter adaptation should be done according to our goal. We define our goal in adjusting parameters below.

In FMGPSO, in the first step of the algorithm we look for exploration then in the final step, we look for exploitation. We can control this policy by adjusting parameters c_1 and c_2. This strategy can be implemented by increasing and decreasing the values of c_1 and c_2. The rules that have been considered for updating these values in each iteration are shown in Table 1.

The parameter c_3 controls the closeness between particles of a same group and global best position of the nearest group. Through our presented strategy, we can avoid merging groups in the infancy of the algorithm. In the final steps of the algorithm we will search the area that might haven't been searched by the other groups. Because in the final steps, by ignoring limitations over merging the groups, algorithm will be capable of searching the majority of the search space.

Parameters c_1, c_2 and c_3 change dynamically through the entire process of optimization. For calculating these parameters, we have defined two fuzzy inference systems for calculating c_1, c_2 and c_3. The first fuzzy inference system determines the value of c_1, c_2 and the second fuzzy inference system calculates the value of c_3. These fuzzy inference systems are stated below.

C. Parameter adaptation(phase 3)

As mentioned previously, in this paper two different fuzzy systems with three outputs have been designed. First fuzzy inference system calculates values of social parameter and cognitive parameter. The second fuzzy inference system calculates the value of c_3.

Exploration and exploitation can be controlled by cognitive and social parameters. The values of these parameters have been considered to be in the range of 0.5 and 2.5 [15].

It has been proved that by utilizing fuzzy inference systems for calculating the parameters of PSO, the algorithm will reach better results [16]. In this paper, diversity of the particles in each group and number of iterations are considered to be the inputs of the first fuzzy inference system. In the second fuzzy inference system, iteration is the only input and the output is c_3.

In this paper the diversity for each group is calculated as formula (7):

$$\text{diversity}(j) = \frac{1}{n_j} \sqrt{\sum_{i=1}^{n} \left(x_{ij} - \tilde{x}_j\right)^2} \qquad (7)$$

In formula (7), n_j is the number of particles in group j, $\tilde{x_j}$ is global best position in group j. The other input for both fuzzy inference systems is iteration. The iteration has been calculated as formula (8).

$$\text{Iteration} = \frac{current\ iteration}{total\ number\ of\ iterations} \qquad (8)$$

It is obvious that the output in formula (7) must be normalized. But the output in formula (8) is in the range of 0 and 1 then normalization is not needed.

Parameters c_1 and c_2 are used to control exploration and exploitation. Higher value of c_1 corresponds to more exploration and higher value of c_2 corresponds to more exploitation. Since in the beginning of the algorithm, exploration is more crucial, the value of c_1 should be high. But as the algorithm proceeds, the value of c_1 should decrease and the value of c_2 should increase. By increasing c_2, the algorithm tends to exploitation rather than exploration. Rules for controlling these parameters are shown in Table 1.

The value of c_3 should be specified such, in the infancy of the algorithm, c_3 should reach the highest possible value (+1) and as the algorithm proceeds, the value of c_3 should decrease. By decreasing the value of c_3 from +1 to -1, the algorithm will be capable of searching the majority of the search space. Especially near the global best position of nearest group. The rules for the second fuzzy inference system is shown in Table 2.

The two fuzzy inference systems work independently. To design fuzzy inference systems, first we have to define membership function of inputs and outputs. All of the membership functions have been considered triangular. The membership functions are illustrated in Fig.1-5.

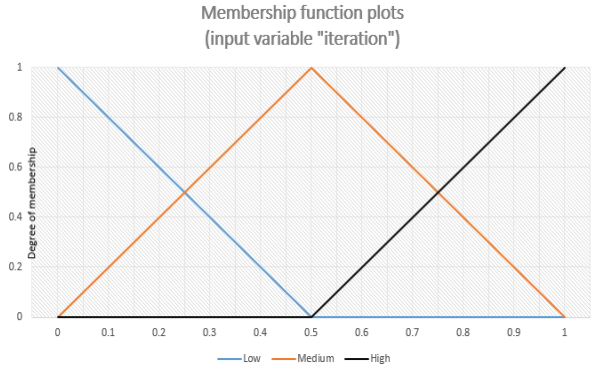


Fig.1. Input 1: iteration

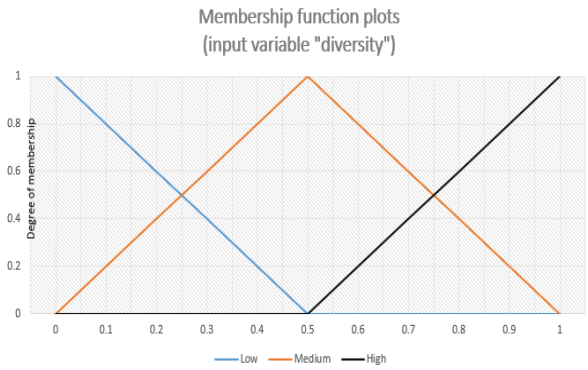


Fig.2. Input 2: diversity

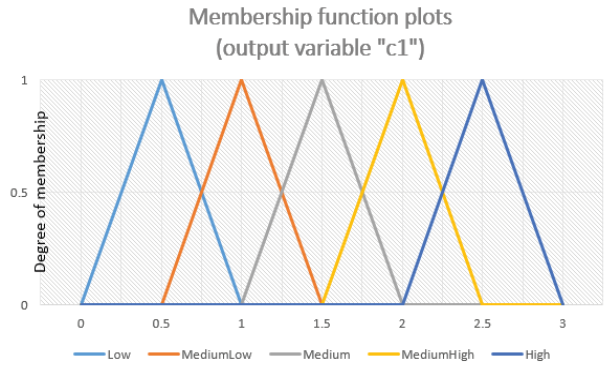


Fig.3. Output 1: C_1

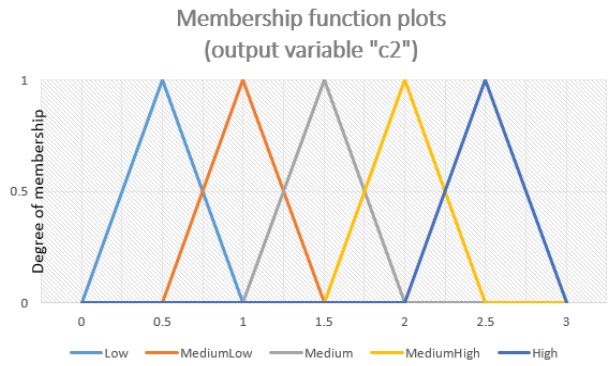


Fig.4. Output 2: C_2

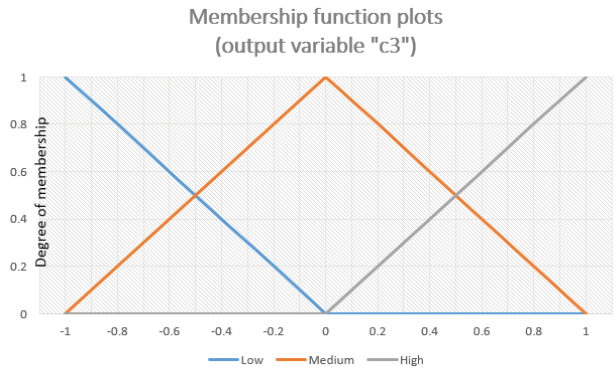


Fig.5. Output 3: C_3

With respect to all the member functions which illustrated before, we defined rules for inference fuzzy system number one and two according to table.1 and table.2.

Table I. Rules of First Fuzzy Inference System

Input	Premise of Fuzzy rule	Output of Fuzzy rule
diversity, iteration	If iteration is low and diversity is low	Then C1 is high, C2 is low
diversity, iteration	If iteration is low and diversity is medium	Then C1 is medium high, C2 is medium
diversity, iteration	If iteration is low and diversity is high	Then C1 is medium high, C2 is medium low
diversity, iteration	If iteration is medium and diversity is low	Then C1 is medium high, C2 is medium low
diversity, iteration	If iteration is medium and diversity is medium	Then C1 is medium, C2 is medium
diversity, iteration	If iteration is medium and diversity is high	Then C1 is medium low, C2 is medium high
diversity, iteration	If iteration is high and diversity is low	Then C1 is medium, C2 is high
diversity, iteration	If iteration is high and diversity is medium	Then C1 is low, C2 is medium high
diversity, iteration	If iteration is high and diversity is high	Then C1 is medium low, C2 is high

Table II. Rules of Second Fuzzy Inference System

Input	Fuzzy rules	output
iteration	If iteration is low	C_3 is high
iteration	If iteration is medium	C_3 is medium
iteration	If iteration is high	C_3 is low

IV. Results

The proposed algorithm was examined on several benchmark functions, from unimodal functions to multimodal functions. Functions that have been used are: Rastrigin, Shwefel 2.26, Ackly, Step function and Rosenbruck. These functions have been used for testing the proposed algorithm. In some of the above functions the problem of finding global minima is of a great issue. The number of dimensions are set to 1000 for all of the functions. The results are the average of 100 independent

executions. The results of testing MGPSO and FMGPSO on the mentioned functions are shown in Table 3.

Table III. Results

Functions	Fuzzy MGPSO	MGPSO
shwefel 2.26	-3.3187*E4	-2.5914*E4
rastrigin	9.804*E3	1.1691*E4
rosenbrock	1.76*E4	9.0483*E4
step function	698	1.7327*E3
ackly	0.995	15.94

V. CONCLUSION

One of the most important and challenging contexts in computer science is optimization. We have shown that by a few changes in MGPSO algorithm we can achieve a more powerful algorithm. First of all, these changes include determining parameters of MGPSO by defined fuzzy inference systems. Second of all, instead of making particles of a same group to get away from global best position of other groups constantly, we need a more dynamic strategy. The proposed strategy is that in the beginning of the algorithm, making particles of a same group to get away from global best position of the nearest group and as the algorithm proceeds, these particles will get closer to the global best position of the nearest group. This strategy makes algorithm to search most of the search space and consequently reach better results than MGPSO algorithm. Defining new fuzzy rules or proposing new formula for updating particles, can be a good subject for later researches.

REFERENCES

[1] L.Zadeh, "Fuzzy logic and its applications". New York, NY, USA, 1965.

[2] Lin, Chin-Teng, and CS George Lee. "Neural-network-based fuzzy logic control and decision system." Computers, IEEE Transactions on 40.12 (1991): 1320-1336.

[3] Chow, Mo-yuen, and Hahn Tram. "Application of fuzzy logic technology for spatial load forecasting." Transmission and Distribution Conference, 1996. Proceedings., 1996 IEEE. IEEE, 1996.

[4] Al Alavala, Chennakesava R. Fuzzy logic and neural networks. New Age International, 2008.

[5] Eberhart, Russ C., and James Kennedy. "A new optimizer using particle swarm theory." Proceedings of the sixth international symposium on micro machine and human science. Vol. 1. 1995.

[6] AlRashidi, Mohammed R., and Mohamed E. El-Hawary. "A survey of particle swarm optimization applications in electric power systems." Evolutionary Computation, IEEE Transactions on 13.4 (2009): 913-918.

[7] Assareh, E., et al. "Application of PSO (particle swarm optimization) and GA (genetic algorithm) techniques on demand estimation of oil in Iran." Energy35.12 (2010): 5223-5229.

[8] Jiang, Ming-Hui, and Xu-Chuan Yuan. "Construction and application of PSO-SVM model for personal credit scoring." Computational Science–ICCS 2007. Springer Berlin Heidelberg, 2007. 158-161.

[9] Shi, Yuhui, and Russell C. Eberhart. "Empirical study of particle swarm optimization." Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on. Vol. 3. IEEE, 1999.

[10] Zhan, Zhi-Hui, et al. "Adaptive particle swarm optimization." Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 39.6 (2009): 1362-1381.

[11] Secrest, Barry R., and Gary B. Lamont. "Visualizing particle swarm optimization-Gaussian particle swarm optimization." Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE. IEEE, 2003.

[12] Seo, Jang-Ho, et al. "Multimodal function optimization based on particle swarm optimization." Magnetics, IEEE Transactions on 42.4 (2006): 1095-1098.

[13] Li, Tao, Chengjian Wei, and Wenjang Pei. "PSO with sharing for multimodal function optimization." Neural Networks and Signal Processing, 2003. Proceedings of the 2003 International Conference on. Vol. 1. IEEE, 2003.

[14] Brits, Riaan, Andries P. Engelbrecht, and Frans van den Bergh. "Scalability of niche PSO." Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE. IEEE, 2003.

[15] Engelbrecht, Andries P. Fundamentals of computational swarm intelligence. John Wiley & Sons, 2006.

[16] Melin, Patricia, et al. "Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic." Expert Systems with Applications 40.8 (2013): 3196-3206.

mpEAd: A Tool For Diagramming Multi-Population Evolutionary Algorithms

Sebastian Lenartowicz[1] **and Mark Wineberg**[2]
University of Guelph
50 Stone Road, Guelph ON N1G 2W1
[1]*slenarto@mail.uoguelph.ca*
[2]*mwineber@uoguelph.ca*

Abstract— *Multi-population evolutionary algorithms are, by nature, highly complex and difficult to describe. Even two populations working in concert (or opposition) present a myriad of potential configurations that are often difficult to relate using text alone. Little effort has been made, however, to depict these kinds of systems, relying solely on the simple structural connections (related using ad hoc diagrams) between populations and often leaving out crucial details. In this paper, we propose a notation and accompanying formalism for consistently and powerfully depicting these structures and the relationships within them in an intuitive and consistent way. Using our notation, we examine simple co-evolutionary systems and discover new configurations by the simple process of "drawing on a whiteboard". Finally, we demonstrate that even complex, highly-interconnected systems with large numbers of populations can be understood with ease using the advanced features of our formalism.*

1. Introduction

From the beginning, it has been obvious that evolutionary algorithms (EAs) could make use of multiple populations in order to facilitate more complex searches and increase the power of the search itself. Though it is doubtless that others exist, there are four main types of multi-population systems that have been investigated. In the island model[1], solutions move between different discrete populations that use the same objective function. The predator/prey model[2] uses multiple populations to perform fitness evaluation – an individual in one population is compared against one or more individuals in another population, where its fitness score increases as those of the others decrease (and vice versa). Co-operative co-evolution[3] is yet another different system, in which members of each different population comprise different elements of the complete solution (and members must be drawn from each population to form and evaluate a full solution). Finally, hierarchical systems, though sparsely investigated (and with conflicting definitions[4][5]), use a variety of different multi-population structures that utilize levels in order to achieve their aims.

Investigation into multi-population EA systems has waxed and waned over the decades, likely because such systems tend to produce complexities that simple single-population systems do not incur. Populations may exchange both genetic and evaluative information[1][6], and, in more esoteric systems, other types of information as well[7]. There are often complex spatial structures connecting the populations formed by this web of relationships[1], and the structure may be distinctly different depending upon the nature of the information[7]. Furthermore, there are recursive effects between multiple populations – as seen in many co-evolutionary systems[3][6][8] – that lead to problems such as Red Queen[9], the loss-of-gradient effect[8], and decoupling[8]. This complexity in the structure of information flow throughout the system is in addition to the actual movement of individuals between populations, as seen in island-model migration[1]. All of this is further exacerbated when considering hierarchical multi-population systems. Confusion about the very idea of what constitutes a hierarchical system can easily be seen in even a cursory review of the literature[5][7][10]; all of these systems are called hierarchical and incorporate elements of hierarchy, but all are different, with very few elements in common!

To combat the confusion arising from all this complexity, we have developed a graphical formalism that encapsulates the different relationships that can exist between multiple populations in an EA system. The multi-population EA diagram (mpEAd, pronounced "emm-pede") employs a concise visual grammar to depict multiple populations and the information flow between them, in a similar way to how the Unified Modelling Language (UML)[11] captures the categorization of and relationships between the component parts of object-oriented software systems.

At this juncture, it is important to note that the mpEAd system is not intended to depict the internal mechanisms and dynamics of single populations; in other words, it does not describe the different selection methods, reproductive operators, etc., of the evolutionary algorithm used by a population. Instead, mpEAd treats each population as a black box, accepting inputs and producing outputs for consumption by other populations or itself. This is approximately analogous to the static "structural diagrams" of UML, which depict the relationship between the system components (such as classes and subsystems) rather than their time-dependent activities, which are modelled in its "behavioural" diagrams. It should also be noted, however, that, while inspired by UML, mpEAd does not incorporate any of the notational conventions found in it, instead using a visual language that is more suited to modelling EA systems. Finally, it is important to stress that mpEAd is more than the topological structure of the populations, as frequently seen when discussing migration. The relationships depicted in mpEAd are much broader in scope, and model all types of information exchange between populations, with migration being only a single subset.

This paper is divided into two parts: the first presenting the basic elements of mpEAd necessary to structure any system and the second extending this notation to model more complex systems. Examples are provided throughout to demonstrate the notation and power of the mpEAd system.

2. Essential Structure of mpEAd

All attempts to depict multi-population EAs base themselves on the graph formalism, with populations represented as nodes (with edges taking variable meanings). However, as with class diagrams in UML, where nodes are classes and edges the relationships between them (messages, roles, etc.), the complexity of an EA system is not fully captured by a simple graph and additional diagrammatic formats are required in order to capture all aspects of the EA's functioning.

With this in mind, a few design principles emerged organically while constructing the mpEAd formalism: intuitiveness, consistency,

ISBN: 1-60132-438-3, CSREA Press ©

distinctiveness, and simplicity. While the roles of intuitiveness and simplicity are simple and intuitive (and yet sometimes difficult to achieve), a brief explanation of the other principles is warranted. While consistency may seem similarly straightforward, the notion of establishing a visual grammar is often lost when attempting to communicate information. Ensuring that similar things appear similar helps bring to mind meaning and allows ease of learning, extensibility, and creativity. In contrast to this, differences should appear different. This promotes readability and ease of interpretation. We also held to two supplementary principles: that the diagram should render naturally in black and white (for publication), and that it should be simple to draw by hand on a whiteboard or piece of paper.

2.1 Basic Elements of mpEAd

The graph at the heart of mpEAd incorporates two types of nodes: population nodes and computation nodes, as well as a number of different edge types. All of these are described in detail below.

2.1.1 Population Nodes

A population node corresponds to a single optimization algorithm (usually an EA) and a set of solutions (the population). It is denoted using a simple hollow rectangle with solid borders and the name of the node written inside. The population node, when drawn, always includes a set of multiple parallel lines, which serve as a visual reminder of the many members of the population inside.

2.1.2 Information as Edges

Edges in the mpEAd graph are used to model the information flow between nodes. As information flow is directional, mpEAd becomes a directed graph, and, per the convention, uses arrows to indicate direction. The types of information available in a multi-population EAs are numerous and varied but can, however, be categorized into two distinct groups: genetic and non-genetic. Genetic information consists of any information used to construct or embody a solution, being referred to as genotypic and phenotypic, respectively. These are often interchangeable, as the genotypic is often immediately evaluated for fitness as if it were a phenotype. Furthermore, even when phenotypes are used, they are produced and consumed during a single decoding/evaluation step, and discarded upon completion. This often stops being true in multi-population systems, where phenotypes may be passed around for various purposes and used by multiple evaluations. The flow of genetic information is represented using a solid edge with a closed arrowhead that is either hollow (genotypic information) or filled (phenotypic information). Examples of these edges are given in Figure 1.

Evaluative information, such as objective and fitness values, is nearly universal in optimization; this information is represented in mpEAd using a dashed edge with an open arrowhead, with the dashed edge serving to indicate that evaluative information is non-genetic in nature. While other types of non-genetic information, such as statistical or control information, may exist in multi-population optimizing systems, a full discussion of these is outside the scope of this paper.

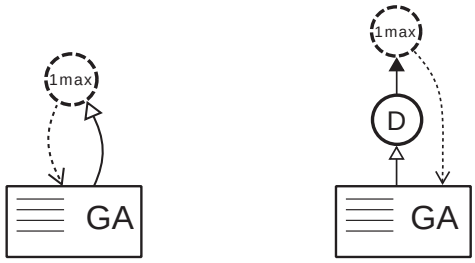
If the same unit of information has multiple simultaneous recipients, the edge is drawn as diverging from a single source. Conversely, if two pieces of information are required, the lines to the node are kept separate and distinct and are not combined in a similar way.

2.1.3 Computation Nodes

The second type of node, the computation node, is less obvious and is one of the elements that makes the mpEAd formalism more than simply a topological model of the connections between populations.

Depiction	Type of Information
	Phenotypic
	Genotypic
	Evaluative

Fig. 1: Arrowheads and edges used in mpEAd.



(a) A basic single-population EA solving the 1max problem.

(b) A variant that decodes genotypes into phenotypes for evaluation.

Fig. 2: Examples of a basic EA.

The role of the computation node is to take in one or more streams of information, perform processing on them, and to provide the result to another node or nodes. Computation nodes perform a variety of information processing operations, including but not limited to decoding genotypic information into phenotypic information, evaluating fitness, and combining information from different sources. The computation node is depicted using a large hollow circle, often labelled with a name, such as the name of the fitness function used for evaluation.

In general, the border of the circle matches the line type of the output edge(s). If the output is of mixed type, the circle's border alternates evenly between solid and dashed. It should be noted that the fundamental difference between a computation node and a population node is that, while both types can perform information processing, the computation node is stateless and does not store information, only taking input and producing output based upon it.

2.1.4 Putting the Basic Elements Together

The simplest multi-population system is one with only a single population; i.e. the simple EA. Two simple examples of this kind of system, solving the universally-known 1max problem, can be seen in Figure 2 to provide context for understanding the basic elements of mpEAd. In Figure 2a, the genetic information is evaluated directly, with the resulting fitness value being passed back to be stored within the population. In Figure 2b, the same evaluation takes place, but must first be decoded into a phenotype before evaluation can take place.

2.2 mpEAd in Action

While efforts have been made in the past to model the interactions between populations, they are often simplistic and rely on *ad hoc*

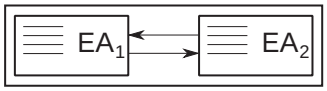
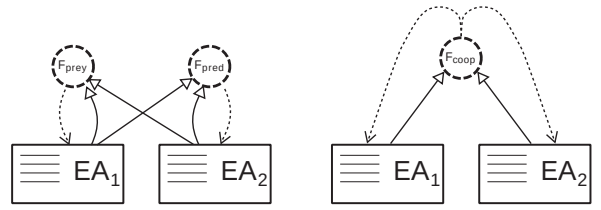
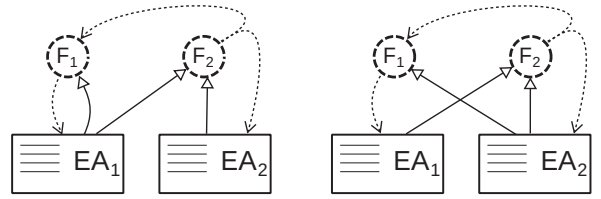


Fig. 3: A typical naïve way to model the systems in Figure 4.



(a) A basic predator/prey co-evolutionary system.

(b) Standard co-operative co-evolution.



(c) A hybrid of co-operative and predator/prey co-evolution with modification.

(d) A hybrid of co-operative and predator/prey co-evolution with cross-modification.

Fig. 4: Examples of various co-evolutionary systems. The two complementary systems in 4c and 4d are both, to the best of our knowledge, novel.

notations, similar to what is seen in Figure 3. The power of mpEAd becomes apparent in comparison to this, as it permits much more accurate and detailed modelling of how the populations interact. All of the diagrams in Figure 4 are different co-evolutionary systems that would be equivalent to the one in Figure 3. Many disparate types of multi-population systems (in this case, a variety of co-operative and competitive co-evolutionary systems) can thus be represented in a way such that their similarities, as well as their differences, become apparent.

Figures 4a and 4b depict a pair of standard co-evolutionary systems that are familiar to most EC researchers. In Figure 4a, the diagram shows both EA populations sending members to the predator and prey evaluation functions, which are used to compute the two different fitnesses. Figure 4b, in comparison, depicts a co-operative system – where the individuals from the two populations are combined to produce a single fitness value that is applied to both.

With the co-evolutionary systems in Figures 4c and 4d, interesting possibilities begin to appear. These systems are unknown in the literature, but by using mpEAd can easily be conceived of, modelled, and constructed. On examination, they appear to be a hybrid between the co-operative and predator/prey systems seen in Figures 4a and 4b. For both of these systems, a single fitness value is produced based on input from the two different populations; however, for one of the populations, the fitness value is modified, either by the individual itself or the individual from the other population. Many different co-evolutionary systems can be easily constructed in an analogous manner, demonstrating the power of the mpEAd formalism for both modelling and discovery.

3. Advanced Features of mpEAd

While mpEAd has many additional, powerful features that make modelling even very complex systems trivial, there are too many to exhaustively discuss here. Instead, we concentrate on the ones necessary to provide the most understanding for the most systems. To this end, we describe four additional features: edge labels, inset arrowheads, macro boxes, and ellipsis notation.

3.1 Edge Labels

The co-evolutionary examples given in Figure 4 hide a great deal of important detail regarding the structure of the information being passed around the system. In particular, when considering evaluation, they do not provide information about how many individuals are required, where they come from, how they are to be selected, where they are to be stored, etc. There are many approaches for matching individuals between co-evolutionary populations, ranging from simple pairing, to pairing with an elite, to exhaustive combination pairing. Because these different constructs would result in an mpEAd that otherwise looks the same, edge labels can be used to disambiguate the selection and matching of individuals between nodes.

The simplest kind of edge label is a letter variable, which is used to indicate sequential iteration through the individuals in a given population, both for selection and storage of incoming values. These variables can be thought of as indices to individuals within the population. Similarly, numbers (either single or in a range) are used to indicate when multiple individuals are drawn from a population in order to perform a computation. The algorithm by which these individuals may be chosen can be specified by a forward slash followed by an algorithm name or symbol (which should be described in accompanying literature) following the number or range. A more thorough discussion of this notation is outside the scope of this paper and will be explored in future work. An asterisk (*) is a special case of numeric value, in which the entire population is used and, in this case, no algorithm specifier may be provided. The asterisk was chosen rather than the more conventional n used in computer science because n could be mistaken for an index variable, and the asterisk as a symbol is commonly used to denote "everything" or "all" (e.g. as used in string matching and the Unix command line).

It should be noted that these categories of edge labels were developed while considering currently existing co-evolutionary and multi-population systems, and are likely to be far from exhaustive. It is almost unquestionable that extensions to the edge label notation will occur in future work as more use cases are considered.

3.1.1 Examples

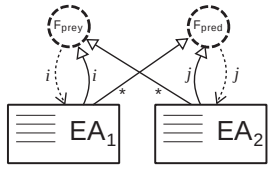
The utility of edge labels becomes apparent when considering the mpEAds in Figure 5, all of which are some variation on two-population co-evolution (as seen in Figures 4a and 4b).

Figures 5a to 5c describe variations on Figure 4a. The first example, Figure 5a, is a common implementation of the predator/prey model, in which each predator is tested against all prey, and each prey is tested against all predators for their respective fitness values. This is computationally expensive (being $O(n^2)$), and subsequently, variants using less than the full population are common.

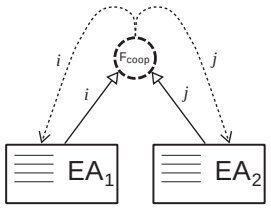
Figure 5b represents such a system, in which each individual from one population is paired with some other individual from the other population for evaluation. The method by which the other individual is selected is left unspecified. This individual could be randomly chosen, be the most fit, or selected by some other algorithm.

In Figure 5c, which represents an asymmetrical approach to predator/prey (used only for demonstration purposes – so far as we are aware, no such system exists in the literature). In this example, each predator is compared against ten prey, using the specified selection mechanism (drawing them at random). The prey, meanwhile, is evaluated in a fixed 1:1 pairing with a given predator.

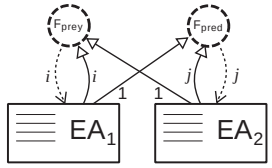
Figure 5d depicts the simplest co-operative system that can be inferred from Figure 4b, using a similar 1:1 pairing of individuals from each of the populations to that seen in the predator/prey exaple in Figure 5c. This structure, though outwardly very different, is actually quite similar to that seen in Figure 5e, which, despite



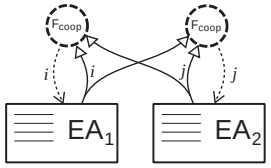
(a) The basic predator/prey co-evolutionary system from Figure **??** with edge labels.



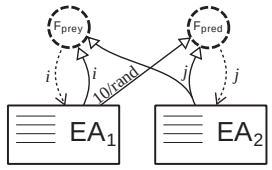
(d) The standard co-operative co-evolution from Figure 4b with edge labels.



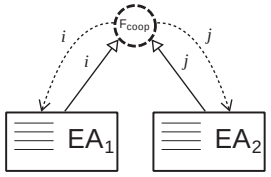
(b) A variant of the basic predator/prey system.



(e) A slightly different co-operative system that appears similar to a predator/prey system.



(c) A more complex variant of the predator/prey system.



(f) The same system as depicted in 5e that is more clearly co-operative (though still subtly different from 5d).

Fig. 5: Examples of edge labels.

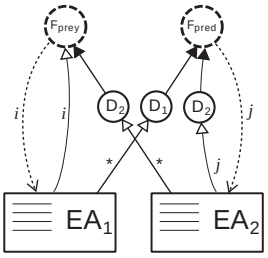
its superficial resemblance to the various predator/prey systems, is actually co-operative due to the 1:1 mapping between i and j for evaluation and the use of the same objective function for the evaluations.

There is, however, a very subtle difference between Figures 5d and 5e: in 5d, the same information is sent to both populations (being stored at locations i and j), whereas, in 5e, the information sent to both populations may not necessarily be the same. Finally, Figure 5f actually depicts the same functional system as 5e, as the different information streams are depicted separately rather than coming from the same source.

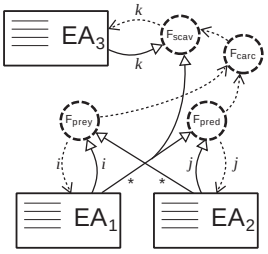
3.1.2 Examples of Greater Complexity

Figure 6a depicts a variation on predator/prey in which the raw genotypes are first decoded into phenotypes before being used for standard predator/prey evaluation. It should be noticed that the borders of the decoding functions are solid, as the output of the computation nodes is phenotypic (and therefore genetic), which is represented using solid lines. The predator requires information from both the prey population and the predator population to be decoded before it can be evaluated, while the prey population only requires the predator to be decoded, while the genotypes of the prey themselves are acted upon directly.

In Figure 6b, a third population is introduced to model a more general version of predator/prey based on the work of de Boer, Folkert, and Hogeweg[12]. The third population is composed of "scavengers" whose fitness is dependent on the fitnesses of both predator and prey, but which do not affect the fitness of either.



(a) A variation on predator/prey in which genotypes are decoded into phenotypes before being considered together.



(b) Predator/prey extended to include scavengers[12].

Fig. 6: Examples of more-advanced systems using edge labels.

The scavengers, much like their biological counterparts, exist only to "pick up the scraps" after the predator and prey have finished evaluating each other. In this system, the scavengers require the prey genotype to establish "edibility"; after all, if the scavenger cannot digest the prey, it will go hungry!

In this example, in addition to the three objective functions (one per population), there is a fourth function, F_{carc}, where "carc" is an abbreviated form of "carcass". This function takes the evaluated fitnesses of each predator/prey pair and produces a "fitness value" for the consumption of the scavenger's objective function. This value, in effect, represents whether a given prey was actually killed (and can therefore be consumed), as well as how much of the prey is left over and can be used by the scavenger (hence "carcass", a term not used in the original paper[12]). The genetic information of the prey is also forwarded to the scavenger, in order to establish edibility as described above. It should be noticed that the mpEAd clearly represents all aspects of this process, as well as indicating the asymmetrical role of the scavenger population.

3.2 Inset Arrowheads

Migration between populations is a common feature of multi-population systems and it would be remiss to exclude it from mpEAd. Yet, migration presents an apparent quandary: at first glance, it appears to break with the "information flow" model, as an actual individual is being transported rather than formless information. On deeper reflection, it becomes obvious that migration can be modelled as a transfer of genetic information followed by a state change in the receiving population. There is, however, a distinction to be made between the arrival of genetic information to be added to the population versus the arrival of genetic information to be incorporated into existing individuals in that population. The first, obviously, is migration. The second is lesser-known, although still existing in the literature, where it is known as hierarchical composition[13]. In every other instance in mpEAd, arrowheads that touch a population node affect individuals within that population. Consequently, a genotypic arrowhead that touches the node border maps more closely to hierarchical composition (which changes the individual) than migration (which adds an individual to the population). Migration, then, is modelled using inset arrows located along the edge in question (as seen in Figure 7).

3.3 Macro Boxes and Ellipsis Notation

A macro box is, much like in a programming language, a simple shorthand for repeating elements. This is typically used in mpEAd to depict the same evaluation being used independently for multiple

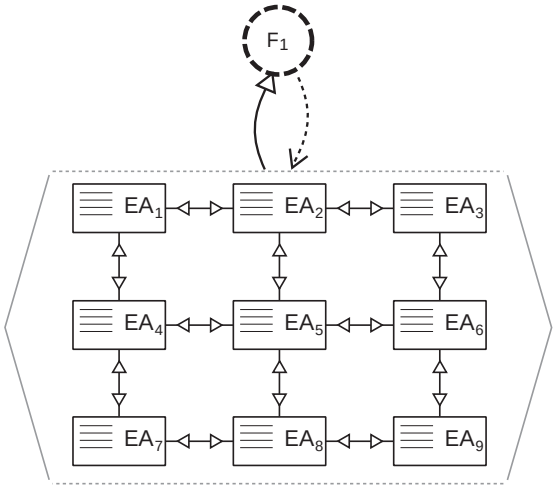


Fig. 7: A small island-model system using a 3x3 grid of populations.

populations (this being distinct from a single evaluation shared by multiple populations, which implies co-evolution).

An element that originates arrows ending at the border of the macro box is duplicated and connected to every element within the box, while elements whose arrows do not end at the macro box border are not duplicated. Macro boxes are are depicted using a elongated gray hexagon in which the long face is dashed and the remaining faces are solid. A simple example of macro box usage can be seen in Figure 7, in which a traditional island-model EA is shown with migration occurring across a 3x3 grid of populations. While macro boxes have many additional uses and properties, a full discussion of these is outside the scope of this paper.

The ellipsis (...) is used analogously to abbreviating text. It indicates the presence of many more identical components of a system, the actual drawing of which would be difficult or unwieldy. The ellipsis is always accompanied by a horizontal bar with vertical ends and an integer (value greater than 2) that indicates the total number of elements, including those actually drawn. An example of this usage is given in Figure 8, in which a 32x32 migratory grid is depicted in a compact form.

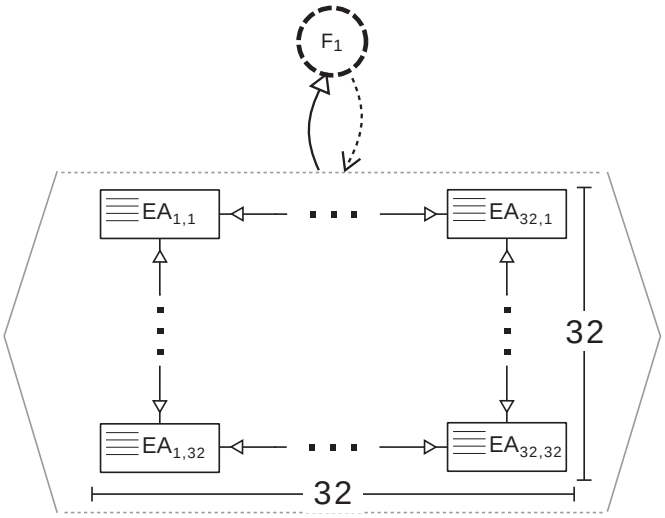


Fig. 8: A much larger island-model system, using a 32x32 grid of populations.

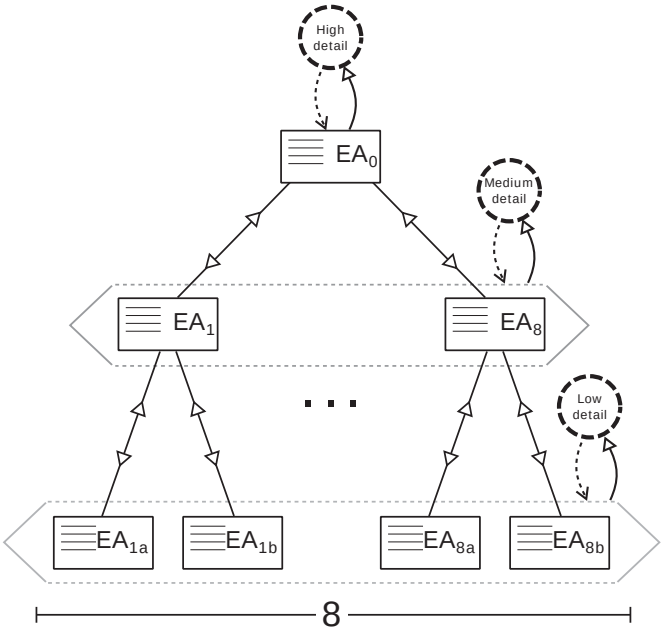


Fig. 9: An extended version of the hierarchical GA created by Sefrioui and Périaux[4].

The full power of the combination of these two concepts can be found in Figure 9, which depicts an extended version of the hierarchical GA created by Sefrioui and Périaux[4]. This system is used to find solutions for an problem for which the objective function is computationally expensive to evaluate, but can be approximated using functions that require less computational effort at the expense of precision. Sefrioui and Périaux first use a number of coarsely-evaluated populations to search for promising solutions, which are then passed upwards in order to be more finely evaluated. They also allow solutions to be passed down, in order to assist the lower-level populations and keep them "current". While the original version takes the form of a binary tree of populations, this variant adds additional subtrees below the "high detail" node, resulting in a system of 25 nodes rather than the original 7. Such a system would be able to more effectively cover the entire search space than the original model, potentially accelerating the process of locating a solution to a complex problem.

4. Conclusion

The mpEAd formalism is a graphical notation designed to permit the depiction of large, complex multi-population EA systems. Designed with the goals of being as intuitive, consistent, distinctive, and simple as possible, mpEAd is a powerful modelling tool for systems often considered too complex to describe clearly. Even with a system as small as two populations, we have seen that, through the use of mpEAd, it is possible to envision not only existing systems, but to diagram and reason about systems not in the literature in an easy and clear way. In the future, even more elements may be added to mpEAd as EC itself grows and matures, necessitating unenvisioned interactions and relationships. Ultimately, it could be possible to develop an "mpEAd IDE", where complex systems are created visually before being rendered down to source code automatically. While these pursuits remain in the future, it is clear that mpEAd has practical applications in the here and now, depicting the exceedingly complex in a simple manner.

References

[1] Darrell Whitley, Soraya Rana, and Robert B Heckendorn. "The island model genetic algorithm: On separability, population size and convergence". In: *Journal of Computing and Information Technology* 7 (1999), pp. 33–48.

[2] W Daniel Hillis. "Co-evolving parasites improve simulated evolution as an optimization procedure". In: *Physica D: Nonlinear Phenomena* 42.1 (1990), pp. 228–234.

[3] Mitchell A Potter and Kenneth A De Jong. "A cooperative coevolutionary approach to function optimization". In: *Parallel problem solving from nature-PPSN III*. Springer, 1994, pp. 249–257.

[4] Mourad Sefrioui and Jacques Périaux. "A hierarchical genetic algorithm using multiple models for optimization". In: *Parallel Problem Solving from Nature PPSN VI*. Springer Berlin Heidelberg. 2000, pp. 879–888.

[5] Mehmet Gulsen and Alice E Smith. "A hierarchical genetic algorithm for system identification and curve fitting with a supercomputer implementation". In: *Evolutionary Algorithms*. Springer. 1999, pp. 111–137.

[6] Kenneth A De Jong. *Evolutionary computation: a unified approach*. MIT Press, 2006.

[7] QS Li et al. "A multilevel genetic algorithm for the optimum design of structural control systems". In: *International Journal for Numerical Methods in Engineering* 55.7 (2002), pp. 817–834.

[8] R Paul Wiegand and Jayshree Sarma. "Spatial embedding and loss of gradient in cooperative coevolutionary algorithms". In: *Parallel Problem Solving from Nature-PPSN VIII*. Springer. 2004, pp. 912–921.

[9] Ludo Pagie and Paulien Hogeweg. "Information integration and red queen dynamics in coevolutionary optimization". In: *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*. Vol. 2. IEEE. 2000, pp. 1260–1267.

[10] Kim-Fung Man, Kit Sang Tang, and Sam Kwong. *Genetic algorithms: Concepts and designs*. Springer Science & Business Media, 2012, pp. 65–74.

[11] James Rumbaugh, Ivar Jacobson, and Grady Booch. *Unified Modeling Language Reference Manual, The*. Pearson Higher Education, 2004.

[12] Folkert K de Boer and Paulien Hogeweg. "Co-evolution and ecosystem based problem solving". In: *Ecological Informatics* 9 (2012), pp. 47–58.

[13] Richard A Watson. "Compositional evolution". PhD thesis. Brandeis University, 2002.

ISBN: 1-60132-438-3, CSREA Press ©