SESSION

MODELING, VISUALIZATION, VIRTUAL ENVIRONMENTS, APPLICATIONS, TOOLS AND TECHNIQUES

Chair(s)

TBA

Modeling Plant Functional Traits and Elevation in the Cairngorm Mountains of Scotland

Valles, D.[1], Apple, M.E.[2], Dick, J.[3], Andrews, C.[3], Gutiérrez-Girón, A.[4], and Pauli, H.[5]
[1]Information Systems Dept., Wake Forest University, Winston-Salem, North Carolina, USA
[2]Dept. of Biological Sciences, Montana Tech, Butte, Montana, USA
[3]Center for Ecology and Hydrology, Edinburgh, Scotland
[4]Dept. of Plant Biology, Complutense University of Madrid, Madrid, Spain
[5]Austrian Academy of Science & University of Natural Resources and Life Sciences, Vienna, Austria

Abstract*. Alpine plants are sensitive to climate change and their functional traits influence their abilities to live in specific habitats on mountain slopes. In the RAPT (Researching Alpine Plant Traits) project, we collected in-situ data that was used to develop a model of existing plant functional traits along a gradient of elevation and climatic variables on Sgòran Dubh Mòr, a summit of the GLORIA (Global Observational Research Initiative in Alpine Environments) target region in the Cairngorm Mountains of Scotland. The MATLAB function polyfit(), for polynomial curve-fitting linear regression and the numerical model approach to the collected trait data were used to obtain representations of species models that corresponded well with the data on existing plant functional trait distribution. This model may be used to predict changes in the distribution of plant functional traits, and possibly species, with climate change in mountainous regions of the Earth.*

Keywords: modeling, plants, climate, traits, alpine

1 Introduction

Plants are inextricably linked with environmental constraints and thus have evolved to respond to abiotic and biotic factors in the environment [1], and characteristics of plants that influence their interactions with the environment are called plant functional traits. Examples of plant functional traits include but are not limited to: 1) overall growth form; 2) types of flowers and diaspores produced; 3) formation of clones; 4) photosynthetic pathways; 5) possession of evergreen or deciduous leaves; and 6) morphometric characteristics of leaves such as thickness, area, and complexity of shape. Particular functional traits may prove advantageous to plants that are subjected to different climatic conditions, and may then prove instrumental in determining which species are lost, become newly established in, or remain in plant communities that are subjected to climate change. Alpine plants are sensitive to climate change [2, 3] and analysis of their distributions and functional traits is an important tool in determining species-specific and functional type-specific responses of alpine plants to climate change [1, 4].

The Global Observational Research Initiative in Alpine Environments (GLORIA) program is an international network of long-term monitoring target regions established on summits of many of the world's mountain ranges [1, 2, 3, 4] with the purpose of understanding the responses of alpine plants to climate change. As functional traits are found in every plant, and as alpine plants are sensitive to climate change, and as climatic conditions change with elevation, we investigated plant functional traits along an elevational gradient downslope from the Scottish GLORIA target region. The goals of the investigation were to develop a knowledge base of the current distribution of alpine plant species with particular functional traits and to develop predictive models of species responses and plant functional type responses to alpine climate change.

Table 1: List of Plant Functional Traits and Properties

Plant Functional Traits	Units
Avg. Leaf Thickness	mm
Avg. Leaf Length	mm
Avg. Leaf Width	mm
Leaf Dry Weight	grams
Plant Height	mm
Plant Lateral Spread	mm
Leaf Area	cm^2
SLA (Specific Leaf Area)	cm^2/g

The software tool MATLAB was utilized to develop the models of each species through the data recorded from eight distinct functional traits. Table 1 shows the list of the plant functional traits, and the units in which each trait was measured. Once models are made of each trait for each species, they are used to derive overall species model functions. The MATLAB function *polyfit()* is the polynomial curve fitting function that finds the coefficients of a polynomial $P(x)$ of degree N that fits the data best in a least-squares sense. P is a row vector of length $N+1$ containing the polynomial coefficients in descending powers [7],

$$P(1) * x^N + P(2) * x^{N-1} + \cdots + P(N) * x + P(N+1) \quad (1)$$

The approach in modeling each plant species is to accurately determine the mathematical functions that describe the functional traits of each species. The data points recorded for each functional trait only describe unique characteristics of each species, but the data does not represent any certain order that can yield to a definitive model function. Therefore, the data of each functional trait is randomly positioned within its own set of values. Each function's coefficient values were collected and averaged to determine the overall coefficient values that produce each species' function model. In order to evaluate and validate the model functions, the functions were compared to the changes of functional traits with the elevation changes. It is hypothesized that by using polynomial curve-fitting linear regression MATLAB function *polyfit()* as the numerical model approach to the data collected for plant functional traits, all species model representations can be obtained.

2 RAPT Project Overview

The Researching Alpine Plant Traits (RAPT) project is an investigation of the responses of alpine plants and their functional traits to climate change in the Cairngorm Mountains of Scotland. RAPT was developed in order to investigate plant functional traits along an elevational gradient on Sgòran Dubh Mòr, a peak in the Cairngorm Mountains of Scotland. The RAPT project was established at Sgòran Dubh Mòr because it is the highest sub-summit of the Scottish GLORIA target region and because information gained on plant functional traits from Sgòran Dubh Mòr can be linked with plant functional trait data from the Scottish GLORIA target region.

2.1 Site Description

Sgòran Dubh Mòr (57°04'49.76"N, 3°48'28.85"W) is a mountain that rises steeply and continuously to 1111m from the Allt a' Mharcaidh stream within the Allt a' Mharcaidh catchment of the Cairngorms National Park of eastern Scotland [5]. Soils of Sgòran Dubh Mòr are acidic and are derived from granitic parent material. While trees exist lower in the Allt a' Mharcaidh catchment, none were present at the Sgòran Dubh Mòr site.

2.2 Species

Species of Sgòran Dubh Mòr represented in the RAPT sampling regime included vascular plants, bryophytes, and lichens. Their scientific names were abbreviated for brevity in equations and models by using the first letter of the genus name followed by a period and the first two letters of the species name. The scientific names, abbreviations in bold, and common names are as follows: *Blechnum spicant* (L.) Sm., **B. sp**, the deer fern; *Calluna vulgaris* (L.) Hull, **C.vu**, heather; *Carex bigelowii* Torr. ex Schwein, **C.bi**, Bigelow's sedge; *Deschampsia flexuosa* (L.) Trin., **D.fl**, wavy hairgrass; *Diphasiastrum alpinum* L., **D.al**, alpine clubmoss; *Empetrum nigrum* L, **E.ni**, crowberry; *Eriophorum angustifolium* Honck., **E.an**, cottongrass; *Galium saxatile* L., **G.sa**, heath bedstraw; *Huperzia selago* L., **H.se**, fir clubmoss; *Juncus trifidus* L., **J.tr**, highland rush; *Lycopodium annotinum* L., **L.an**, stiff clubmoss; *Nardus stricta* L., **N.st**, mat grass; *Rubus chamaemorus* L., **R.ch**, cloudberry; *Trichophorum cespitosum* (L.) Hartm, **T.ce**, deergrass; *Vaccinium myrtillus* L., **V.my**, bilberry; and *Vaccinium vitis-idaea* L., **V.vi,** lingonberry. The bryophytes included *Racomitrium lanuginosum* (Hedw.) Brid, **R.la**, woolly fringe moss; *Sphagnum sp.*, **S.sp**, peat moss. Lichens included *Cetraria islandica* (L.) Ach. **C.is** and *Cladonia sp.*, **C.sp**.

2.3 Sampling

Groups of four $1m^2$ vegetation plots with $10cm^2$ grids were randomly placed at the lower (750m), middle (940m), and upper slope (1105m) along an elevational gradient ascending the western slopes of Sgòran Dubh Mòr in the summer of 2013. The upper plots were positioned 6m below the Sgòran Dubh Mòr GLORIA summit to avoid trampling plants. Plots were placed inside and outside of the late snowfield areas which are found at mid-elevation (940 m). The presence and percentage surface cover of all vascular plant species, lichens, and bryophytes within each of the $1m^2$ plots were recorded along with frequency counts of all species in each plot subdivision per elevation. Plant height and lateral spread were recorded *in-situ* and lateral spread of clonal plants was measured as the maximum width of the above ground patch of a tussock.

The leaves of ten plants per species per elevation and from inside and outside of the mid-elevation snowfields were measured for shape, margins, length, width and thickness within 15 hours of collection. Length: width ratios were calculated and leaf area was measured from photographs with Image-J [6]. Specific leaf area (SLA, cm^2/g), was calculated from leaf dry weights. Plant trait information collated from taxonomic texts included overall morphology; life history strategies (longevity; clonality and the presence of rhizomes, stolons, adventitious roots or other clonal structures; phenology and type of flowering; pollination; seed or spore production, morphology, persistence, and dispersal); leaf longevity and whether the leaves were evergreen or deciduous; and rooting architecture and depth. Traits not included in this initial model are mentioned here to provide a context for the overall goals of this project, which are to develop predictive models based on large suites of trait data and biotic and abiotic drivers.

Abiotic factors measured at Sgòran Dubh Mòr included slope, aspect, and proximity to late snowfield areas for each plot at each elevation. Values for soil bulk density (g/cm^3), volumetric water content (g/cm^3), water content (g/g), porosity (%); water filled pore space (%), wet and dry weights (g), and the weight (g) and volume (ml) of stones were obtained at each elevation and inside and outside of late snowfield areas.

3 Numerical Modeling

The data collected for each functional trait consists of data sets containing ten data points. An initial approach of modeling was to utilize the data arranged in the order each fundamental trait was recorded. The data presents itself in a linear form in which

an n^{th}-order polynomial accomplishes a best-fit. As an example of *D. flexuosa*'s (D.fl) lateral spread functional trait data, Figure 1 below shows the different results of increasing the order of *polyfit()* function from 4th-order to 9th-order fit.

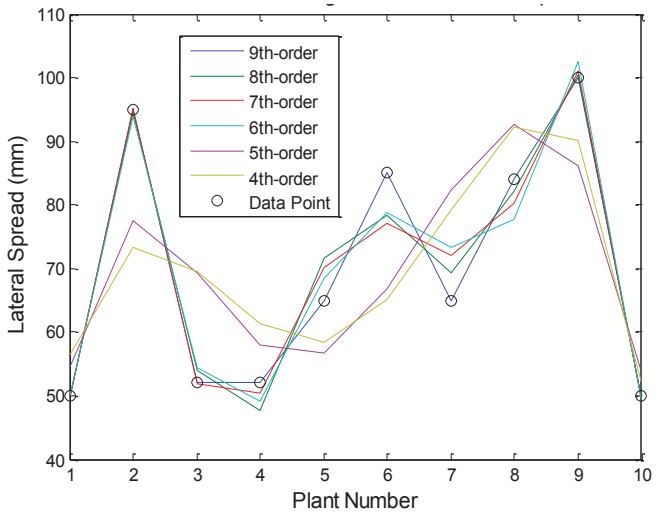


Figure 1. *D. flexuosa* – Plot 1: Lateral Spread Data, Initial Modeling Approach

This initial approach helped to evaluate the approximation of each polynomial, and its coefficient of determination R-square values. In each case of the functional traits for each species, the R-square value increases with the order of each polynomial. However, the leading coefficients of the polynomial kept approaching zero, and the functions become ill-conditioned. In containing the ill-conditioned polynomials and very small leading coefficients, each function was reduced in order by calculating the error estimation of the polynomials. The approach to the error estimation of an n^{th}-order polynomial in order to reduce the order of each function is to check when the error (e_n) is added to the n^{th}-order polynomial (f_n) case and generates $(n+1)^{th}$-order polynomial (f_{n+1}) case [8]. That is,

$$f_{n+1}(x) = f_n(x) + e_n \tag{2}$$

, in which yields to identify the order of the polynomials in where the functions will not be ill-conditioned.

3.1 Determining the Order of Polynomials

In obtaining best-order for each polynomial, each functional trait was modeled. As mentioned earlier, the data points for each functional trait describe the characteristic measurement value obtained, but the order in which the data was recorded does not represent a certain order in which the data follows. Therefore, the data was randomized for each trait at each run, the coefficient values are stored and then re-run for extra iterations of random positions of the data. In order words, we take the array of the ten data points of a function trait such as:

$$Data_{Func.Trait} = [d_0, d_1, d_2, d_3, \dots, d_{10}] \tag{3}$$

Then, the random permutation *randperm(N)* function was utilized which returns a vector containing a random permutation of the integers 1 to N [9], in where one possible outcome of the permutations changes to (3) can represent the data to:

$$Data_{Func.Trait} = [d_6, d_3, d_7, d_1, \dots, d_7] \tag{4}$$

The randomization approached helped in evaluating each functional trait in different ways. In utilizing the *polyfit()* function, the coefficients of each randomized iteration were obtained. As in examples, Figure 2 and Figure 3 show the 9th-order and 7th-order polynomial coefficients values of *D. alpinium* (D.al) for the 40-iterations that were performed respectively.

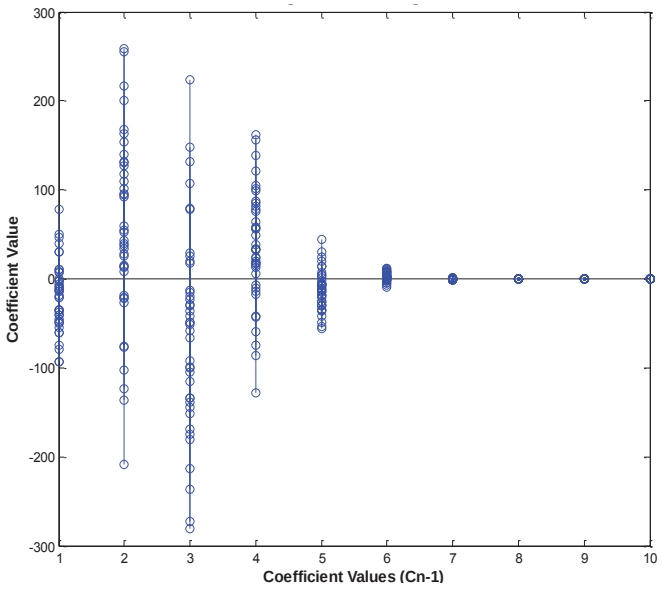


Figure 2. *D. alpinium* - Leaf-Thickness (9th-order Model): Coefficient Values Through 40-Iterations

In the case of Figure 2, it can be observed that the leading coefficients are so small that it can be assumed that they are zero, and the range of coefficient values is larger than the range found in Figure 3. In reducing the order of the model from a 9th-order to a 7th-order, the model of each iteration went from an ill-condition to a stable model, and the values of each coefficient are more concentrated as the order of the model was reduced. Therefore, the functional trait models will be represented in a 7th-order polynomial function, and the number of iterations will be increased in order to obtain more converged coefficient values.

3.2 Trait Data at Different Elevations

The form in which the function models will be compared is in how well the elevation changes are represented through different functional trait models. These elevation changes will help to evaluate and validate the approach described in the following section. However, the benchmark window is small

since only four plant species exist at different elevations within the sampling 1m^2 plots. The four plant species are: C.bi (Carex *bigelowii*), D.fl (*Deschampsia flexuosa*), V.my (*Vaccinium myrtillus*) and E.ni (*Empetrum nigrum*).

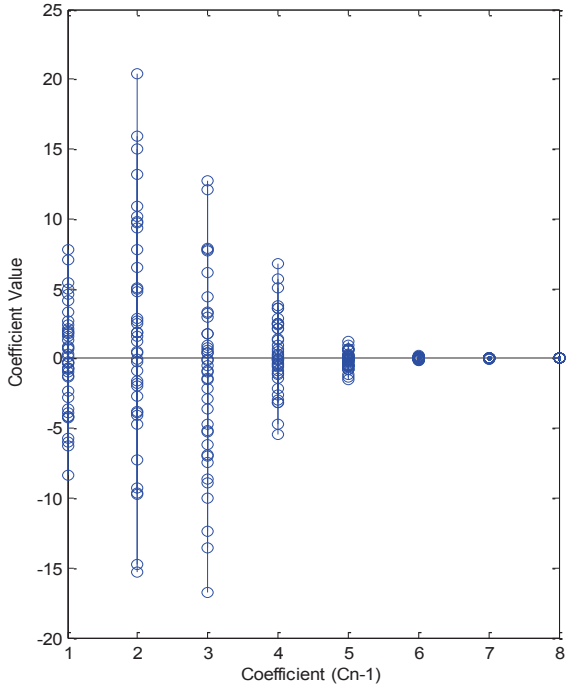


Figure 3. *D. alpinium* Leaf-Thickness (7th-order Model): Coefficient Values Through 40-Iterations

The elevation changes are represented in three levels: high- (H), mid- and low-elevations (L). Within the mid-elevation, there are the internal-snowfield (Mi) and external-snowfield (Me) areas, but they will be regarded as part of the mid-elevation for benchmark purposes. Figure 4 displays an example of how the functional trait leaf area of plant species E.ni changes from low to mid-elevations by using two sets of values: 1) the average-value of the set of leaf area data at each elevation, and 2) the maximum-value of the set of leaf area data at each elevation. For this case, the functional trait leaf area increases for E.ni as the elevation increases. Therefore, the model for the functional trait leaf area for E.ni must represent the increase as well.

4 Procedures

As mentioned before, the data points recorded for each functional trait are the description of the characteristics of each plant species, but they do not represent any certain order of position or priority within the plants. Therefore, as mentioned in the last section, the data placement within the functional trait array was randomized in order to produce different permutations of the coefficient values in a 7th-order polynomial. The number of iterations of the randomized data can also vary, in which in turn, the number of coefficient values from the *polyfit()* function can expand the range of values for each of the coefficients. Then, the number of iterations must be increased in order to: 1) utilize more of the possible existent permutations for data in every functional trait, 2) obtain a mean-value that is more accurate for each coefficient value, and 3) revise if, as the number of iterations will converge to more stable coefficient mean values.

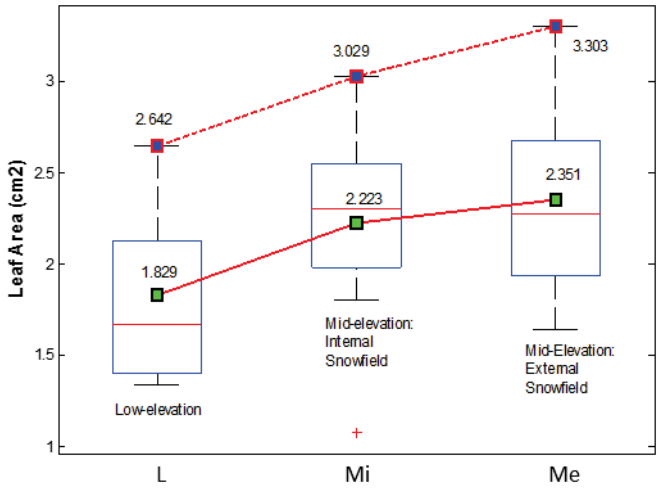


Figure 4. *E. nigrum* Leaf-Area Changes from Low to Mid-Elevation through Average and Max Values

In developing the model for the functional traits, the workflow is shown in Figure 5. Individual functional trait data array were randomized, $data_{FT}[10]$, using the *randperm()* function that produces a new random data array $data\_r_{i_{FT}}[\,]$. The new data array is then evaluated with the *polyfit()* function to produce the 7th-order curve-fitted polynomial with eight coefficient values. These coefficient values are then placed within a coefficient array, $C[10]_i$, and then added to the previous coefficient values $C[10\,]_{i-1}$. The workflow then repeats itself for the next iteration to determine the next set of coefficient values. The iterations values are 40, 4000, and 40,000. Once all iterations have been performed, the average value of each coefficient is obtained and used as the coefficient value for each functional trait model. The average value are used in order to avoid using coefficient values in spikes to a very high positive or very low negative values, and represent the coefficients values that are grouped more closely together.

In the case for developing the models for the plant species, the workflow is shown in Figure 6. Each individual functional trait data array is denoted as: $data_{FT(1:8)}[10]$; *m*-number of functional traits, and each is randomized using the *randperm()* function that produces a new random data array, $data\_r_{i_{FT(1:m)}}[10]$. The new data arrays are then evaluated with the *polyfit()* function individually and added as aggregates to all of the eight coefficient values. The aggregated coefficient values are then placed within a coefficient array $C_m[10]_i$. The workflow then repeats itself for the next iteration to determine the next aggregated set of coefficient values. Once all iterations have been performed, the average values are obtained of each

coefficient, and they're used as the coefficient values for each plant species model.

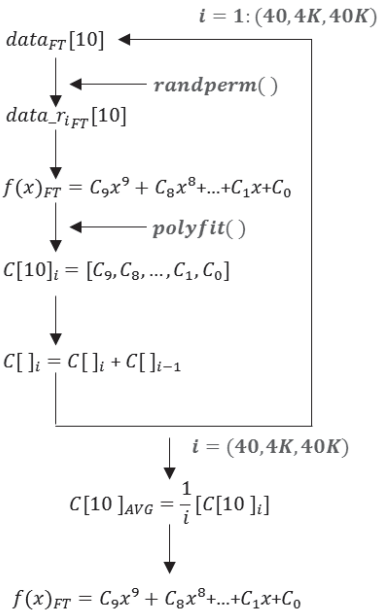


Figure 5. Workflow for Developing Functional Trait Models

In other words, each functional trait model, $f(x)_{FT}$;

$$f(x)_{FT} = C_9x^9 + C_8x^8 + \cdots + C_1x + C_0 \qquad (5)$$

, is combined by aggregate with the other functional trait models of the same plant species;

$$f(x)_{plant} = \sum_{j=1}^{m} f_{FT_j}(x) \qquad (6)$$

, in where the process is then iterated *i*-times with new randomized position of the data for each trait and taken the mean of each coefficient in which yields;

$$f(x)_{plant} = \frac{1}{i}\sum_{j=1}^{m}[C_9]_j x^9 + \cdots + [C_1]_j x + [C_0]_j \qquad (7)$$

, and the function described in (6) is now denoted as;

$$f(x)_{plant} = \frac{1}{i}\sum_{j=1}^{m} f_{FT_j}(x) \qquad (8)$$

5 Results

The workflows from Figure 5 and Figure 6 each develop functional trait and plant species models in a 7th-order polynomial respectively. In order to benchmark the model functions, the functional trait models are compared in how the functional trait data changes on average when elevation changes on average. As mentioned earlier, elevation changes are described in three levels: high-elevation (H), mid-elevations (Mi & Me) and low-elevation (L). In the mid-elevation, there are the internal-snowfield (Mi) and external-snowfield (Me) areas, and also, the small number of plant species existing in multiple elevations are mentioned in section 3.2.

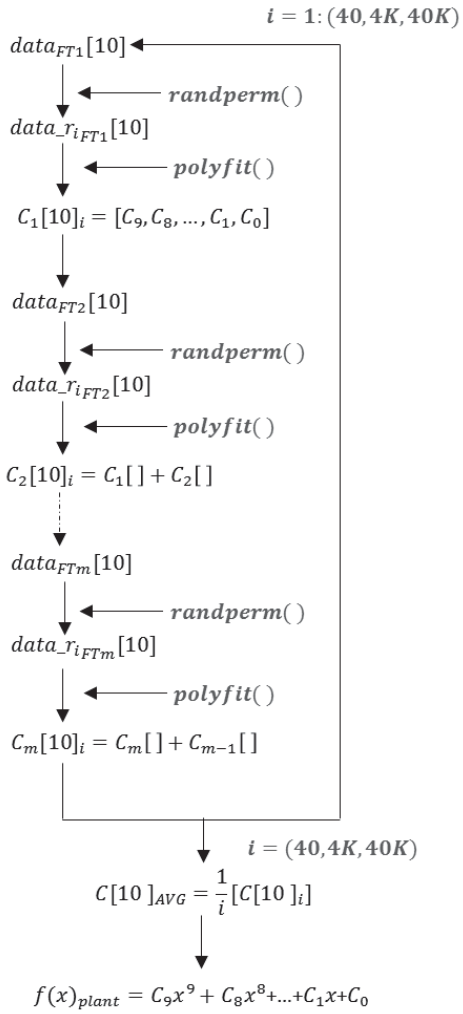


Figure 6. Workflow for Developing Plant Species Models

To display a resultant case of the functional trait workflow, Figure 7 shows the height functional trait for C.bi. (*Carex bigellowii*) as it changes elevation on average illustrating: a) the average of the height functional trait of C.bi using the recorded data, and b) by using each height functional trait model at each elevation. In this case, there are three models that represent C.bi at each plot elevation. Each of the models is evaluated from $x = 1$ to $x = 5$ or $x = (1{:}\,5)$. As it can be observed in Figure 7 b), the mean representation for each plot retains the mean values from part a) as x increases. However at $x = 5$, the three models for C.bi height functional trait begin to fail in representing the correct behavior. Meaning, as x increases for each height model, then the leading coefficients begin to grow the effect of the much larger tailing coefficients in which distort each of the three models. However, the data is still well represented with the three resultant height functional trait models by using (5) at $x = (1{:}\,4)$ as the number of iterations

increases that provides proper coefficient mean values for each model.

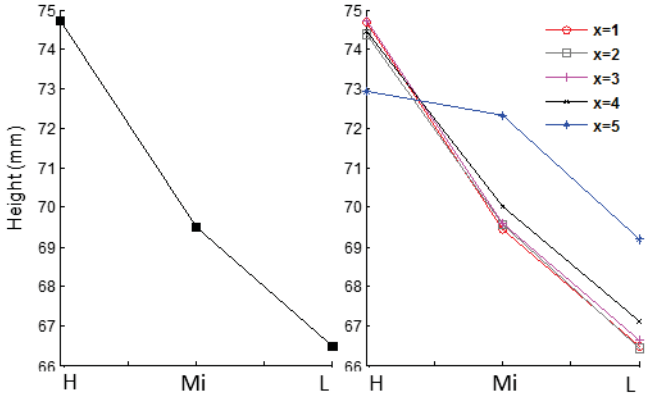


Figure 7. *C. bigellowii* Average Height at Different Elevations – a) Average of Data as Elevation Changes [left], b) Trait Model Evaluated at x = (1:5) [right]

In a separate case, Figure 8 shows the leaf area functional trait for D.fl (*Deschampsia flexuosa*) as it changes with elevation on average by illustrating a) the leaf area functional trait of D.fl using the recorded data, and b) by using each leaf area functional trait model at each elevation. In this case, there are four models that represent D.fl at each plot elevation. Each of the models is evaluated from the same x range as Figure 7. As it can be observed in Figure 8 b), the mean representation for each plot struggles to reproduce the mean values from part a) as x increases. Although the four models do not provide the exact behavior of the elevation changes, the models are contained within the correct average value ranges. Also, as x increases for each leaf area model, the elevation changes to leaf area are rectified. Similar results were obtained from the other functional traits due to elevation changes as seen in Figure 7 and 8. The trait models showed that the workflow described on Figure 5 (which determines 7th-order polynomials), such as in (5), helped to generated proper reproduction of the traits in changes of elevation from the same plant species.

In a similar manner, the plant models were obtained as described in the workflow from Figure 6. The resultant plant models are also in a 7th-order polynomial from the aggregate of all of the individual functional traits. Table 2 shows the resultant coefficient values of all plant species. From Table 2, it can be seen that the leading coefficients are very small values in most cases, and as x increases, it can cause the models to be disrupted and misrepresent each plant species. Therefore, the models will be also restricted at in the same manner as the functional trait models. In another observation, the trailing coefficients are similar in value range for the same plant species at different elevations. This projects that each overall plant model will have similar properties and elevation characteristics that describe the functional traits in the aggregate. Overall, the approach procedure in obtaining the coefficient values for individual functional trait models for the 7th-order do show reproduction of data and changes of elevation concurrent with the recorded data which helped to obtained the overall plant model coefficient values.

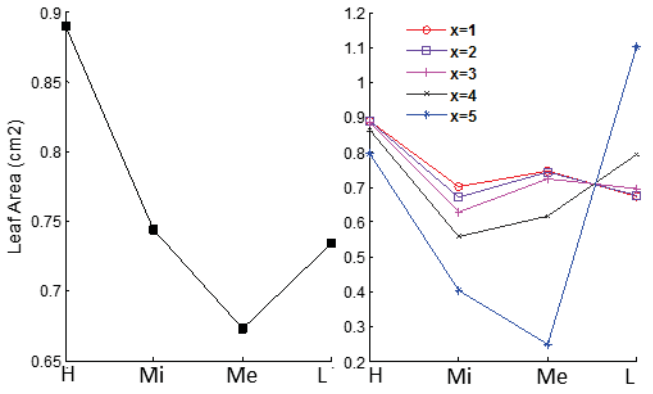


Figure 8. *D. flexuosa* Average Leaf Area at Different Elevations – a) Average of Data as Elevation Changes [left], b) Trait Model Evaluated at x = (1:5) [right]

6 Conclusions

The data collected from the RAPT Project were utilized in order to develop models of the functional traits of the plant species, and of the overall plant models through the aggregated functional trait models. The recorded data that were collected are in sets of ten entries for each of the functional traits in which the order of the recorded data does not represent any biological or organizational order. This is important because the sample number does not have biological significance *per se*. The numerical modeling approach was to utilize the data collected and randomize the position of the data within its vector-form in repetitive iterations. Each of the iterations generated a different outcome of coefficient values of the same data set for each functional trait by using the *polyfit()* function in MATLAB, the curve-fitting function. It determined the order to the polynomials by starting in the maximum 9th-order representation down to 7th-order polynomial.

The functional trait models were developed by using the same iterative approach in determining the order of the polynomials. However, the average of each coefficient was obtained in order to determine the coefficient values of each model. Then, the models of C.bi (Carex *bigelowii*), D.fl (*Deschampsia flexuosa*), V.my (*Vaccinium myrtillus*) and E.ni (*Empetrum nigrum*) were compared according to how the data change with elevation: high- elevation (Plot 1), mid-internal-snowfield (Plot 3), mid-external-snowfield (Plot 4) and low-elevation (Plot 2). The models show consistency with the changes of elevation when the models are evaluated at $x = (1{:}\,4)$. However, the other models display inconsistencies in one instance in where $x \geq 5$. Therefore, each of the model functional trait functions are defined as $x = (1{:}\,4)$.

The overall plant species models were generated by using each functional trait in the aggregate by iterative workflow. The average of the coefficient values were again taken to determine

Table 2: List of Coefficient Values for Overall Plant Species Models

Plant Model	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7
$f(x)_{D.al.\_plot3}$	74.0931	7.2720	-5.8954	2.4075	-0.5378	0.0665	-0.0043	0.0001
$f(x)_{L.an.\_plot2}$	112.9282	-2.1949	1.5697	-0.5608	0.1080	-0.0113	0.0006	0
$f(x)_{C.bi.\_plot1}$	156.3454	-1.5820	1.0344	-0.3622	0.0730	-0.0084	0.0005	0
$f(x)_{C.bi.\_plot3}$	171.6487	4.5836	-3.7129	1.5176	-0.3419	0.0429	-0.0028	0.0001
$f(x)_{C.bi.\_plot4}$	156.3454	-1.5820	1.0344	-0.3622	0.0730	-0.0084	0.0005	0
$f(x)_{Caes.\_plot3}$	95.5284	-1.5443	1.2976	-0.5201	0.1145	-0.0142	0.0009	0
$f(x)_{R.ch.\_plot2}$	227.9216	15.8560	-13.1324	5.3989	-1.2130	0.1512	-0.0098	0.0003
$f(x)_{D.fl.\_plot1}$	117.4476	1.8835	-1.6414	0.6698	-0.1461	0.0176	-0.0011	0
$f(x)_{D.fl.\_plot2}$	230.0989	-6.2883	18.0881	-11.8767	3.4088	-0.4895	0.0346	-0.0010
$f(x)_{D.fl.\_plot3}$	130.0732	0.2353	-0.2396	0.1228	-0.0326	0.0046	-0.0003	0
$f(x)_{D.fl.\_plot4}$	127.6576	3.3975	-2.6334	0.5188	-0.1803	0.0446	-0.0091	0
$f(x)_{V.my.\_plot1}$	260.8595	1.0968	-1.9159	1.0836	-0.2929	0.0412	-0.0029	0.0001
$f(x)_{V.my.\_plot2}$	281.1444	-0.2638	1.6738	-1.2081	0.3589	-0.0521	0.0037	-0.0001
$f(x)_{V.my.\_plot3}$	253.5294	4.5530	-3.9121	1.7096	-0.4056	0.0527	-0.0035	0.0001
$f(x)_{V.my.\_plot4}$	220.9643	-2.4891	2.0329	-0.8184	0.1783	-0.0214	0.0013	0
$f(x)_{E.ni.\_plot2}$	64.3202	-2.0476	1.6784	-0.6439	0.1327	-0.0152	0.0009	0
$f(x)_{E.ni.\_plot3}$	65.5195	5.2449	-4.2459	1.6635	-0.3542	0.0420	-0.0026	0.0001
$f(x)_{E.ni.\_plot4}$	68.1222	7.9333	-6.3021	2.4073	-0.4992	0.0578	-0.0035	0.0001
$f(x)_{H.se.\_plot3}$	128.0943	1.2656	-1.1207	0.4858	-0.1157	0.0153	0.0153	0
$f(x)_{B.sp.\_plot3}$	121.7526	-7.4726	5.7576	-2.1728	0.4474	-0.0514	0.0031	-0.0001
$f(x)_{N.st.\_plot3}$	66.3037	-0.2130	0.3994	-0.2184	0.0536	-0.0066	0.0004	0
$f(x)_{J.tr.\_plot1}$	139.9547	-7.7600	6.4774	-2.6212	0.5757	-0.0702	0.0045	-0.0001
$f(x)_{C.vu.\_plot2}$	50.4534	1.1568	0.3676	-0.0788	0.0095	0.0095	-0.0006	0

each model as seen in Table 2. As with the resultant models of the functional traits, the plant species models also are defined at $x = (1{:}\,4)$.

These models will be utilized in future representations that integrate a suite of plant functional traits [10] with biotic and abiotic environmental variables of Sgòran Dubh Mòr and the GLORIA target region in Scotland, and that place greater emphasis on the distribution of plant functional traits than on the distribution of species, as particular functional traits are not necessarily confined to individual species but can reflect responses to environmental conditions along elevational gradients and with distance from snowfields.

7 Acknowledgements

We acknowledge the Interact Transnational Access program; (HPC) Montana Tech of the University of Montana; (HPC) Wake Forest University; the Center for Ecology and Hydrology, Edinburgh, Scotland, and the GLORIA program.

8 References

[1] Gutierrez-Giron, A., Gavilan, R., 2013. *Plant Functional Strategies and Environmental Constraints in Mediterranean High Mountain Grasslands in Central Spain*. Plan Ecol. Div. DOI: 10. 1080/ 17550874.2013.783641.

[2] Gottfried, M., Pauli, H., et al. 2012. *Continent-wide Response of Mountain Vegetation to Climate Change*. Nature climate change 2: 111-115.

[3] Pauli, H., Gottfried, M., Dullinger, S., et al. 2012. *Recent Plant Diversity Changes on Europe's Mountain Summits*. Science. 336(6079):353-355, DOI: 10.1126/science.1219033.

[4] Venn S, Pickering C, Green K. 2014. *Spatial and Temporal Functional Changes in Alpine Summit Vegetation are Driven by Increases in Shrubs and Graminoids*. AoB PLANTS 6: plu008; doi:10.1093/aobpla/plu008.

[5] Bonjean, M. C., Hutchins, M., and Neal, C. 2007. *Acid Episodes in the Allt a' Mharcaidh, Scotland: An Investigation Based on Sub-hourly Monitoring Data and Climatic Patterns*. Hydrol. Earth Syst. Sci., 11(1):340-355.

[6] Schneider, C. A.; Rasband, W. S. & Eliceiri, K. W. (2012), *NIH Image to ImageJ: 25 Years of Image Analysis*, Nature methods 9(7): 671-675, PMID 22930834.

[7] MathWorks, (2015). *MATLAB™ (R2012a)*. [Online] Available: http://www.mathworks.com/help/matlab/ref/polyfit.html?searchHighlight=polyfit

[8] Chapra, Steven, Canale, Raymond. *Numerical Methods for Engineers*. 5th ed., McGraw-Hill, New York, NY, 2006.

[9] MathWorks, (2015). *MATLAB™ (R2012a)*. [Online]. Available: http://www.mathworks.com/help/matlab/ref/randstream.randperm.html

[10] Perez-Harguideguy, N., Diaz, S., et al. 2013. *New Handbook of Standardized Measurement of Plant Functional Traits Worldwide*. Australian Journal of Botany, 61, 16-234.

Visual Analytics of Complex Networks: A Review from the Computational Perspective

Marian Sorin Nistor[1], **Stefan Wolfgang Pickl**[1], **and Martin Zsifkovits**[1]

[1] Computer Science Department, Universität der Bundeswehr, 85577 Neubiberg, Germany

Abstract – This article is a domain review on existing tools, open-source and commercial, capable of operating with complex networks from a visual analytics perspective. The complex networks capability in computing highly complex and dynamic interactions among entities of complex information systems is widely accepted in literature. This review is aiming for a more reliable complex system analysis for decision makers in case of information overload problems in crisis situations where time plays a key role and human resources are limited.

Keywords: complex networks, visual analytics, networks visualization tools.

1. Introduction

According to [1], every day 2.5 quintillion bytes of data are created. This implies a massive amount of growing data every day. Thus, collecting raw data is no longer the most important issue. Processing and transforming it in valuable information and identifying methods and models which can generate trustworthy knowledge are nowadays challenging the time [2].

Taking a situation where an information system is overloaded. For example, an important airport during a pilots strike with several canceled flights. The airport managers, or just calling them decision makers, are now dealing with a local crisis situation. This soon extends to all the other airports which were expecting the now canceled flights, and now the local crisis situation escalates. Without the help of technology, it is very difficult for the decision makers to process the most important information in their system in order to take quick informed decisions, and this under time pressure.

Finding technologies to solve the data overload problem in a crisis situation, like described above, in many areas it is still an open problem. In literature a few questions are addressed to any technology that claims to overcome this issue: "Who or what defines the 'relevance of information' for a given task? How can inappropriate procedures in a complex decision making process be identified? How can the resulting information be presented in a decision-oriented or task-oriented way?" [3]

The aim of this article is to present the proper technologies available in literature dealing with the data overload problem in a network. Few concepts dealing with this problem will be used here to better define the problem. Therefore, in this article the concept of computational networks for a network resulted from computation will be approached [4]. The concept of a complex network as a simplified frame of a complex system, where the nodes of the network are represented by the entities of the system, and the links between pairs of nodes are the interrelations between the entities of the system [5]. Two complex systems will be used as examples in this article in order to explain the problem briefly. One is an aviation management system and the second a subway transportation system.

Another concept tackled in this article is the network visualization concept. Not network drawing, in general oriented on the visual representation of network's nodes and links, but network visualization in terms of using a network for its data visualization. The last one being more oriented on operating the hierarchies of a complex system, for various view perspectives and interactions between its entities. This concepts will be explained in more detail in Section 2, where the state-of-the-art of this topic is presented. The above introduced concepts will afterwards be interconnected for a larger overall picture. Related work and existing tools will be presented in Section 3. And in the end, Section 4 presents the key challenges and concluding remarks of this review.

2. Network visualization: State-of-the-art

The best solution to elucidate how a system runs is by understanding its structure [6]. The same holds for a network. In order to understand why a network is differently structured than another, or why a network is more vulnerable, or resilient, than the others are, it is important to understand the network's structure [7]. As a network grows in complexity, it is more difficult to handle the interactions between its nodes. Therefore, in order to understand and to work with a complex network, analysis techniques are required. According to the existing literature, the visual analytics techniques can be considered a reliable solution [3,8–17].

2.1. When a network gets complex

A network is a collection of points, which in Computer Science are considered as nodes, with a set of

lines connecting the points between them, called edges. For this article, the usage of the term network is related to a real world component which can be represented, from a mathematical perspective, as a graph, but the term graph will be reserved only for its mathematical model.

The study of networks, from a mathematical approach, has its origins in discrete mathematics. Euler treated the Königsberg Bridge problem in 1736, and his work is reported as the first paper written on the graph theory [18].

In the last decades, the research focus of networks slightly changed from analysis of small networks, and the individual properties of its elements, to large-scale networks. This was possible thanks to computers' evolution which enabled the possibility to collect and analyse data much more easily, and on a larger scale than was possible before [19].

Analysing a small network is no longer a challenge. However, when a network is getting complex with hundreds (or thousands) nodes and links, an unaided eye, even if considered a very powerful analytic tool, is not able to study the complex structure of it in a real time. Today's computers are offering high computational power, which can process complex networks in a matter of seconds. Therefore, analyzing important network properties like the small-world effect, transitivity or clustering, degree distribution, network resilience, mixing patterns, degree correlations, community structure, network navigation, etc., became possible for complex networks, too [7,19,20].

2.2. Complex networks evolved from complex information systems

As defined by [21] an "Information System is what emerges from the usage and adaptation of the Information Technology and the formal and informal processes by all of its users." This type of system most likely consists of interconnected and heterogeneous subsystems which can communicate with the external environment [22]. The complexity is given by the interactions of its entities.

The main characteristic of a complex system, in general, is that they appear to be organized without any external regulation to be used [23], and, in particular, they can fall under some of the properties: heterogeneousness, flow processing, hierarchical organization, and evolution [22]. Hence, complex networks can be built from real complex information systems, for a reliable computation, and for real case application studies like the problem tackled in here.

This article presents two examples of representing a complex information system as a complex network in an attempt to understand its properties, and structure.

2.3. Example 1: Munich subway network

The Munich subway network used for this example is a network with 96 stations (nodes) and 101 connections (links) between its stations. The network is represented in an abstract way in *Figure 1* and *Figure 2*, based on [24]. The representation of the network is a mathematical one, regardless of the topological position of its nodes.

Figure 1 is one abstract-view of the Munich subway network with representation based on the network modules. A module of a network is a subset connected part of the network [25]. The line-up of the modules and their nodes is determined by a star-view layout. For this representation, links between modules are collapsed into one, but with the width proportional with the number of the links collapsed. Nodes in modules are colored on a heat-color based. More precisely, the darker colored nodes represent a high degree, and lighter colored representations for the lower degree nodes.

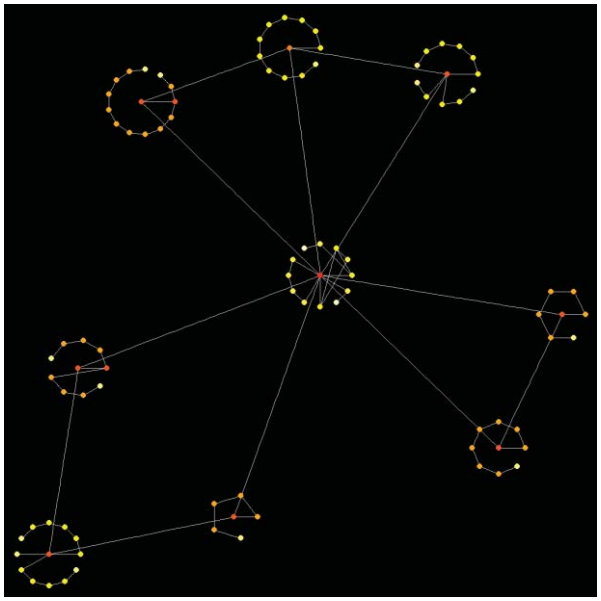


Figure 1. An abstract-view of the Munich subway network based on its modules.

Figure 2 is another way to represent an abstract modular view of the same network. The size of the nodes is proportional with the total number of nodes in a module. The same with the links, their size is proportional with the number of links between the modules.

Compared with the original network representation [26], this kind of representation might not make that much sense for a guidance use. However, for a network analyzing purpose this kind of representation is more suitable for analyzing the network structure properties. In this case it might be possible that the topological network representation is not too much helpful, because a different approach is required, a more analysis oriented one.

Figure 2. Abstract modular view of the Munich subway network.

2.4. Example 2: The worldwide air transportation network

Figure 3 represents the worldwide air transportation network [27], where the gray links represent the air traffic of passengers in the world, connecting 1,000 airports with more than 35,000 links, and the red links represent the basic structure of the network with only 1,300 links. In this case we have the more extended example mentioned in Example 1. Here is more clear the requirement of an additional aid in helping the decision maker to get a reliable interaction among the networks' items.

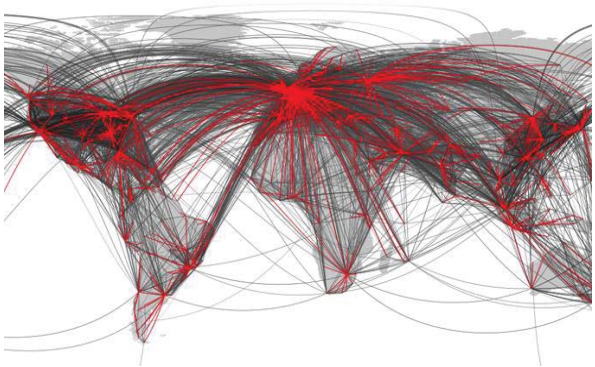


Figure 3. The worldwide air transportation network. Picture copyrights of [27].

The two examples were illustrated to get the feeling of analyzing complex structures. However, representing a complex system as a complex network, and fixing the network plotting part with the computational power of the nowadays computers, doesn't fix the problem. The reliability in analyzing the structure of such complex networks is challenging. As mentioned in the first paragraph of this section, visual analytics might be a reliable solution. The status quo of visual analytics is presented in the following subsection.

3. From Visualization to Visual Analytics

According to [28], visualization began to be studied in a scientific way in 1987. Very much progress has been made since then, and can be divided roughly in two different sciences: Scientific Visualization and Information Visualization.

While the focus of Scientific Visualization is mainly on visualization techniques for an interaction efficiency, Information Visualization emphasizes on visualization of data, with a lately stronger focus on abstract data with no explicit spatial references available [9,29].

A new research field evolved from these two sciences in 2004 [11], Visual Analytics. A first definition is given by [8] as "the science of analytical reasoning facilitated by interactive visual interfaces" and adapted later by [3] to "Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets".

This is to say that Visual Analytics is an adapted version of Information Visualization in order to deal, in an innovative way, with analysis tasks and advanced data analysis algorithms, which is more than just visualizing, and can be considered "an integral approach to decision-making, combining visualization, human factors and data analysis". [2]

The mantra of [30] "Overview first, Filter and zoom, Details on demand", which inspired systems and visualization techniques in Information Visualization, is now adapted by [31] for Visual Analytics to "Analyze first, Show the Important, Zoom, filter and analyze further, Details on demand".

A novel perspective of visualizing data is not offered to decision makers, but they are guided in a loop where data can be interactively operated to offer new perspectives on them, and their descriptions, in a knowledge discovery process. By letting the decision maker at the command of the system, with different view perspectives, and combination of automatic and visual analysis methods, will lead to a better understanding of the complexity of the analyzed system.

This process is the visual analytics process introduced by [3] and exemplified in *Figure 4*. This process distinguishes from the other processes from the very beginning by offering the chance to transform (sort, delete, modify etc.) the input data. Then, the decision makers are able to choose between the automated data analysis, or so called performance optimization, and the visual data exploratory branches.

By choosing the performance optimization branch, models based on the input data will be generated automatically. Parameters refinement are enabled for better results. A gain of knowledge can be followed now from the automated analysis, or after the models are sent for a visual interaction with data.

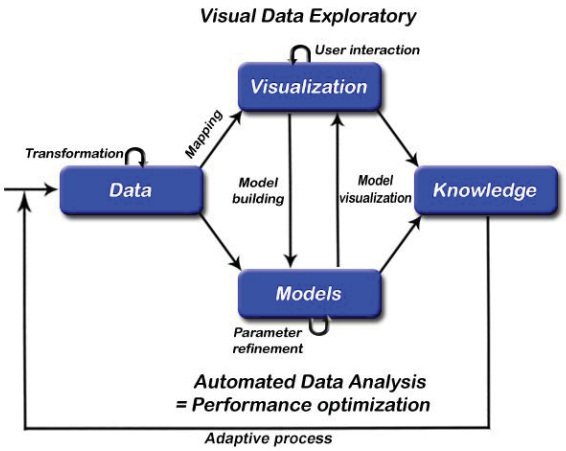


Figure 4. The Visual Analytics Process, adapted from [3].

For a visual data exploratory selection, the user is able to use visual analysis methods (e.g. zoom, filters) for a reliable data exploration, and to generate optimal (e.g. structure understanding of the input data) results for a direct gain of knowledge, or to build models for more insights in between.

This process is an adaptive process, and anytime the user can restart the loop in case the results are not sufficiently useful.

In the end of this section, the list of the existing visualization tools and toolkits is presented, based on recent (in the last three years) comparative reviews available in literature. In order to be eligible for this review, the visualization tool, or toolkit, has to provide functionality for network data.

A review of visualization tools was perform by [32] in 2013. The selection criteria for the tools were to be in an active development (published after 2009, or a new version release in the last two years), and to handle networks with more than 10,000 nodes. Some of them are free to use, some others only enclosed in research papers, and not available for tests. The tools are: Cytoscape [33], Tulip [34], WiGis [35], CGV [36], VisANT [37], ipgraph library [38], Pajek [39], Gephi [40,41], In Situ Framework [42], and Honeycomb [43].

Another review [44], from 2012, finds and compares from a functional point of view (visualization functions, analysis capabilities, and development environment) an important number of available open source visual analytics toolkits. From this list only the toolkits able to visualize network data too will be presented.

The toolkits are: birdeye [45], Flare [46], GraphViz [47], Improvise [48], IVTK [49], JIT [50], JGraph [51], JUNG [52], NetworkX [53], Prefuse [54], Protovis [55], Titan [56], WilmaScope [57], and Zest [58].

A complementary review [12] of the previous one, from the same year, finds and compares from a functional point of view (system and architecture, data management, automatic analysis methods, and visualization techniques) the available commercial visual analytics toolkits. For consistency, the selection criteria of functionality for visualizing network data is applied. The eligible commercial toolkits are only QlikView [59], TIBCO Spotfire [60], JMP [61], Visual Analytics Incorporated [62], and Centrifuge [63].

4. Key challenges and concluding remarks

In this article the data overload problem was presented in relation with existing visual solutions. Such information overload problems often generate crisis situations by not handling the complex structure of important systems properly.

A few concepts were used in this paper in order to better define the problem. The concept of computational networks, complex networks, and networks visualization with emphasizes on visual analytics. Complex systems were represented as complex networks, and the visual analytics techniques presented from literature were considered a reliable solution for this problem.

With the two examples illustrated in Section 2 we demonstrated that the handling of such complex systems is not an easy task for decision makers. The need for the development of new technologies that are overcoming this problem is growing proportionally with the technologies that generate more and more raw data.

Acknowledgements

The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA Grant Agreement Number 317382.

References

[1] IBM, What is big data, available at http://www-01.ibm.com/software/data/bigdata/what-is-big-data.html (accessed on June 20, 2014).
[2] Visual analytics: Combining automated discovery with interactive visualizations, Springer, 2008.
[3] D. Keim, Mastering the information age: Solving problems with visual analytics, Eurographics Association, Goslar, 2010.
[4] L.J. Mueller, M. Dehmer, F. Emmert-Streib, Network-Based Methods for Computational Diagnostics by Means of R, in: Z. Trajanoski (Ed.), Computational Medicine: Tools and Challenges, Springer Vienna, 2012, pp. 185-197.
[5] A. Geiger, A. Kroll, Modeling and analyzing technical systems as complex networks: Detecting inverse response, in: 2013 IEEE Symposium on Computational Intelligence in

Control and Automation (CICA), Singapore, Singapore, pp. 89–96.
[6] Y. Bar-Yam, C. Ramalingam, L. Burlingame, C. Ogata, Making things work: Solving complex problems in a complex world, NECSI, Knowledge Press, Cambridge, MA, 2004.
[7] M. Dehmer, F. Emmert-Streib, Analysis of complex networks: from biology to linguistics, John Wiley & Sons, 2009.
[8] J.J. Thomas, K.A. Cook, Illuminating the path: The Research and Development Agenda for Visual Analytics, IEEE Computer Society, Los Alamitos, Calif., 2005.
[9] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, G. Melançon, Visual Analytics: Definition, Process, and Challenges, in: A. Kerren, J.T. Stasko, J.-D. Fekete, C. North (Eds.), Information Visualization, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 154–175.
[10] R. May, P. Hanrahan, D.A. Keim, B. Shneiderman, S. Card, The state of visual analytics: Views on what visual analytics is and where it is going, in: 2010 IEEE Symposium on Visual Analytics Science and Technology (VAST), Salt Lake City, UT, USA, pp. 257–259.
[11] Pak Chung Wong, Jim Thomas, Visual Analytics, IEEE Computer Graphics and Applications 24 (2004) 20–21.
[12] L. Zhang, A. Stoffel, M. Behrisch, S. Mittelstadt, T. Schreck, R. Pompl, S. Weber, H. Last, D. Keim, Visual analytics for the big data era — A comparative review of state-of-the-art commercial systems, in: 2012 IEEE Conference on Visual Analytics Science and Technology (VAST), Seattle, WA, USA, pp. 173–182.
[13] G. Andrienko, Space, time and visual analytics, International journal of geographical information science.
[14] D. Hutchison, M.H. Böhlen, T. Kanade, J. Kittler, J.M. Kleinberg, F. Mattern, A. Mazeika, J.C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, S.J. Simoff, Visual Data Mining: Theory, Techniques and Tools for Visual Analytics, Springer, Berlin, Heidelberg, 2008.
[15] J. Kohlhammer, T. Ruppert, J. Davey, F. Mansmann, D. Keim, Information visualisation and visual analytics for governance and policy modelling, Bibliothek der Universität Konstanz, Konstanz, 2010.
[16] Maciejewski R, Rudolph S, Hafen R et al. 2010. IEEE. A visual analytics approach to understanding spatiotemporal hotspots Transactions on Visualization and. Comput.Graphics,16,205-220 (2010).
[17] Sun G, Liang R. 2013. Wu F, Qu H. A web-based visual analytics system for real estate data. Science China (13). Info.Sci.,56,052112 (2013).
[18] N. Biggs, E.K. Lloyd, R.J. Wilson, Graph theory, 1736-1936, Clarendon Press, Oxford [Oxfordshire], New York, 1986.
[19] M.E.J. Newman, The structure and function of complex networks, SIAM review 45 (2003) 167–256.
[20] M. Dehmer, A. Mowshowitz, F. Emmert-Streib, Advances in network complexity, John Wiley & Sons, 2013.
[21] R.J. Paul, Challenges to information systems: time to change, Eur J Inf Syst 16 (2007) 193–195.
[22] D. Bihanic, T. Polacsek, Models for Visualisation of Complex Information Systems, in: 2012 16th International Conference on Information Visualisation (IV), Montpellier, France, pp. 130–135.
[23] C. Zhao, L. Wang, Computational Modeling of Complex Systems: Case Study in Real World, in: 2009 Ninth International Conference on Hybrid Intelligent Systems, Shenyang, China, pp. 33–38.
[24] S. Tripathi, M. Dehmer, F. Emmert-Streib, NetBioV: an R package for visualizing large network data in biology and medicine, Bioinformatics 30 (2014) 2834–2836.
[25] L.O. James, R.G. Stanton, D.D. Cowan, Graph decomposition for undirected graphs (1972) 281--290.
[26] MVV-Muenchen, Netzpläne, available at http://www.mvv-muenchen.de/de/netz-bahnhoefe/netzplaene/index.html (accessed on March 31, 2015).
[27] Northwestern University, Researchers develop method that shows diverse complex networks have similar skeletons, available at http://phys.org/news/2012-06-method-diverse-complex-networks-similar.html (accessed on June 22, 2014).
[28] B.H. MCCORMICK, T.A. DeFanti, M.D. Brown, Visualization in scientific computing, IEEE COMPUTER SOC 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 7, 1987.
[29] R. Spence, A. Press, Information visualization (2000).
[30] Shneiderman B. The. 1996. IEEE. eyes have it: A task by data type taxonomy for information visualizations. In Proc. the Symposium on Visual Languages, Sept., pp.336-343. (1996).
[31] D.A. Keim, F. Mansmann, J. Schneidewind, H. Ziegler, Challenges in Visual Data Analysis, in: Tenth International Conference on Information Visualisation (IV'06), London, England, 05-07 July 2006, pp. 9–16.
[32] S. Jarukasemratana, T. Murata, Recent Large Graph Visualization Tools: A Review, Information and Media Technologies 8 (2013) 944–960.
[33] Cytoscape, available at http://www.cytoscape.org/ (accessed on June 23, 2014).
[34] Tulip, available at http://tulip.labri.fr/TulipDrupal/ (accessed on June 23, 2014).
[35] Gretarsson, B., Bostandjiev, S., O'Donovan, J. and Höllerer, T. 2009. WiGis: A Framework for

Scalable Web-Based Interactive Graph Visualizations, in 17th International Symposium, Graph Drawing, pp. 119-134. (2009).
[36] CGV, available at http://www.informatik.uni-rostock.de/~ct/software/CGV/CGV.html (accessed on June 23, 2014).
[37] VisANT, available at http://visant.bu.edu/ (accessed on June 23, 2014).
[38] igraph library, available at http://igraph.org/redirect.html (accessed on June 23, 2014).
[39] Pajek, available at http://pajek.imfm.si/doku.php (accessed on June 23, 2014).
[40] Gephi, available at https://gephi.org/ (accessed on June 23, 2014).
[41] Bastian, M., Heymann, S. and Jacomy M. 2009. Gephi: an open source software for exploring and manipulating networks, in International AAAI Conference on Weblogs and Social Media. (2009).
[42] Hadlak S, Schulz H J, Schumann H. In. 2011. IEEE. situ exploration of large dynamic networks Transactions on Visualization and. Comput.Graphics,17,2334-2343 (2011).
[43] F. van Ham, H.-J. Schulz, J.M. Dimicco, Honeycomb: Visual analysis of large scale social networks, in: Human-Computer Interaction–INTERACT 2009, Springer, 2009, pp. 429–442.
[44] P.C. Wong, J.R. Harger, P.J. Crossno, D.L. Kao, M.C. Hao, C. Chen, R. Kosara, M.A. Livingston, J. Park, I. Roberts, Comparison of open-source visual analytics toolkits, in: IS&T/SPIE Electronic Imaging, Burlingame, California, USA, SPIE, 2012, pp. 82940E.
[45] birdeye, available at https://code.google.com/p/birdeye/ (accessed on June 23, 2014).
[46] Flare, available at http://flare.prefuse.org/ (accessed on June 23, 2014).
[47] E.R. Gansner, S.C. North, An open graph visualization system and its applications to software engineering, Software Practice and Experience 30 (2000) 1203–1233.
[48] Improvise, available at http://www.cs.ou.edu/~weaver/improvise/ (accessed on June 23, 2014).
[49] The InfoVis Toolkit, available at http://ivtk.sourceforge.net/.
[50] JavaScript InfoVis Toolkit, available at http://philogb.github.io/jit/ (accessed on June 23, 2014).
[51] JGraph, available at http://www.jgraph.com/ (accessed on June 23, 2014).
[52] JUNG, available at http://jung.sourceforge.net/ (accessed on June 23, 2014).
[53] NetworkX, available at https://networkx.github.io/ (accessed on June 23, 2014).
[54] Prefuse, available at http://prefuse.org/ (accessed on June 23, 2014).
[55] Protovis, available at http://mbostock.github.io/protovis/ (accessed on June 23, 2014).
[56] Titan, available at http://www.kitware.com/news/home/browse/VTK%3F2008_03_18%26Titan+Informatics+Toolkit+Update (accessed on June 23, 2014).
[57] WilmaScope, available at http://wilma.sourceforge.net/ (accessed on June 23, 2014).
[58] Zest, available at http://www.eclipse.org/gef/zest/ (accessed on June 23, 2014).
[59] QlikView, available at http://www.qlik.com/ (accessed on June 23, 2014).
[60] TIBCO Spotfire, available at http://spotfire.tibco.com/ (accessed on June 23, 2014).
[61] JMP, available at http://www.jmp.com/ (accessed on June 23, 2014).
[62] Visual Analytics Incorporated, available at http://www.visualanalytics.com/ (accessed on June 23, 2014).
[63] Centrifuge, available at http://centrifugesystems.com/ (accessed on June 23, 2014).

Nonlinear second-order dynamical systems on Riemannian manifolds: Damped oscillators

Simone Fiori and Andrea Bonci
Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche, Ancona, Italy

Abstract— *Linear as well as non-linear mathematical systems that exhibit an oscillatory behavior are ubiquitous in sciences and engineering. Such mathematical systems have been used to model the behavior of biological structures, such as the pulsating contraction of cardiac cells, as well as the behavior of electrical and mechanical components. Chaotic oscillators are currently being used in the secure transmission of information. The state of such classical dynamical systems evolve in the Euclidean space $\mathbb{R}^n$ (typically, $n = 1, 2, 3$). The current paper aims at proposing a principled mathematical technique to design second-order nonlinear dynamical systems over curved Riemannian manifolds and to discuss a numerical simulation framework that is compatible with the structure of such spaces.*

Keywords: Second-order dynamical systems, Non-linear dynamics, Oscillatory behavior, Chaotic systems.

1. Introduction

A wide range of nonlinear phenomena have been studied in the scientific literature by analyzing the harmonic behavior regime of different models of nonlinear oscillators such as the Lorenz model, the Duffing model, the van der Pol model and the Colpitts model. The state of nonlinear dynamical systems may evolve over time in a complex way, by showing non-repeating paths (albeit deterministic), and most of such dynamical systems present a chaotic behavior for specific values of their parameters. Chaos is considered to be an irregular random-seeming behavior, strongly influenced by the initial conditions of a non-linear system and by the values of its parameters. Chaos can be encountered in a number of complex physical processes in various scientific fields such as secure communications [1], [2], cryptography [3] and robotics [4].

Non-linear dynamical systems are currently employed to model the behavior of a variety of complex physical, chemical and biological systems. In mathematical biology, biological processes are often represented by linear or nonlinear oscillators, such as in the modeling of the heartbeats [5], [6], of the metabolism, of the breathing and of constructive effects in genetic systems [7].

In electrical engineering, the van der Pol oscillator explains electrical circuits governed by a second order differential equation but it can also describe how a pacemaker controls the irregular heartbeat of human heart, where the whole cardiac system can be modeled as a working van der Pol oscillator. In mechanical engineering, the Duffing nonlinear oscillator is used to model magneto-elastic mechanical systems (e.g., the motion of a sinusoidally-forced structure undergoing large elastic deflections), whereas, in electrical engineering, the Duffing oscillator can explain the behavior of non-linear electrical circuits [8].

Such classical dynamical systems involve a few variables in low-dimensional Euclidean spaces, such as the real line $\mathbb{R}$ (see, e.g., the van der Pol oscillator), the real plane $\mathbb{R}^2$, and the ordinary space $\mathbb{R}^3$ (see, e.g., the Lorenz system). The current paper aims at proposing a principled mathematical technique to design and to numerically simulate second-order nonlinear dynamical systems over *curved spaces* that take the structure of Riemannian manifolds. The present paper focuses on three specific aspects: on extending previous studies on nonlinear damped oscillator theories on Euclidean spaces to high-dimensional curved (Riemannian) manifolds; on deriving a geometrically-sound framework to numerically integrate the differential equations on manifold that describe the dynamics of the devised general nonlinear oscillator; on illustrating, via computer simulations, interesting patterns over a simple curved manifold, the unit sphere, that affords graphical rendering of the numerical results.

The manuscript is organized as follows. In Section 2, second-order dynamical systems on Riemannian manifolds are introduced. The formulation of the continuous-time equations via an extended stationary-action principle is presented in Subsection 2.1, whereas potential energy functions and damping terms are introduced in the Subsection 2.2. The notion of numerical simulation of the developed dynamical systems on manifolds via geometric-integration techniques is presented in the Section 3: in particular, time-discretization of the system and numerical simulations on the unit hypersphere are respectively discussed in the Subsections 3.1 and 3.2. Concluding remarks are presented in Section 4.

2. Second-order dynamical systems on manifolds

A prototypical second-order dynamical system, whose trajectory evolves in the space $\mathbb{R}^3$, is described by the Newton's law of motion of a particle of mass m subjected to a total force $F : \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{I} \to \mathbb{R}^3$. The trajectory $\gamma : \mathbb{I} \to \mathbb{R}^3$ followed by such particle is computed as the

solution of the Newton's equation:

$$m\frac{\mathrm{d}^2\gamma}{\mathrm{d}t^2} = F\left(\gamma, \frac{\mathrm{d}\gamma}{\mathrm{d}t}, t\right), \tag{1}$$

equipped by appropriate initial/boundary conditions. The term $\ddot{\gamma}(t)$ denotes the instantaneous acceleration of the particle as a function of the time $t \in \mathbb{I}$, the mass term m accounts for the inertia of the particle and the external force F depends on the instantaneous position $\gamma(t)$, on the instantaneous velocity $\dot{\gamma}(t)$ and may depend explicitly on the time t. The total force F accounts for both *internal forces* – typically, due to active or passive damping effects – as well as *external forces*.

In the equation (1), the trajectory is described by three functions $\gamma_1(t), \gamma_2(t), \gamma_3(t)$ that constitute the entries of the vector-valued function $\gamma(t)$. In fact, for every value of the time-parameter t, the function $\gamma(t)$ may be visualized as a point of the space $\mathbb{R}^3$. Likewise, the instantaneous velocity $\dot{\gamma}$ belongs to the ordinary space $\mathbb{R}^3$ and the phase-space associated with the system is $\mathbb{R}^3 \times \mathbb{R}^3$. Whenever the space where the dynamics takes place is more involved than the ordinary space $\mathbb{R}^3$, as a curved high-dimensional surface or even an abstract space, both the representation of the trajectory and the structure of the phase-space become more complicated.

In the present contribution, we treat the case that the underlying space is a Riemannian manifold M. Some smooth manifolds of interest in the scientific literature are the unit hypersphere (with applications in signal processing), the Stiefel manifold (namely, the space of 'tall-skinny' orthogonal matrices, with applications in signal processing and machine learning), the manifold of symmetric positive-definite matrices (with applications in computational mechanics) and the orthogonal group (namely, the space of rotations/reflections, with applications to robotics and signal processing).

At a point $x \in M$, the tangent space to the manifold M is denoted as T_xM and represents the vector space spanned by all tangent vectors to all smooth curves on M passing through the point x. It pays to think of a tangent space T_xM as a local linearization of the manifold M around a point $x \in M$. The trajectory $\gamma(t)$ generated by a dynamical system evolving on a Riemannian manifold M entirely belongs to such manifold as it holds that $\gamma(t) \in M$ for every value of $t \in \mathbb{I}$. By definition of tangent space, the instantaneous velocity $\dot{\gamma}(t)$ belongs to $T_{\gamma(t)}M$ at any time. The symbol TM denotes the tangent bundle associated with the manifold M, which is defined as:

$$TM \stackrel{\mathrm{def}}{=} \{(x, v) | x \in M, v \in T_xM\}. \tag{2}$$

The tangent bundle plays the role of phase space. A Riemannian manifold M is endowed with a bilinear, positive-definite form $\mathbb{G}_x : T_xM \times T_xM \to \mathbb{R}$, termed inner product. Note that an inner product may be applied only to tangent vectors and that, in general, it depends (smoothly) on the point on the manifold $x \in M$.

A few recent contributions appeared in the scientific literature that aimed at extending the classical notion of second-order dynamical system (1). In this regard, notable is the contribution [9], that suggests to employ geometric numerical integration techniques to study dynamical systems subjected to equality constraints. A contribution on the line of extending linear smothering to non-linear (time-independent) stifling on dynamical systems is [10].

2.1 Formulation of the equations via an extended stationary-action principle

In the present paper, the formulation of second-order dynamical systems on Riemannian manifolds is based on the dynamics of a point-wise particle sliding on a manifold M endowed with an inner product $\mathbb{G}$. The stationary-action principle used to formulate the dynamics of such a particle in a dissipative context is based on the notions of Riemannian kinetic energy function, potential energy function and external driving force. The *kinetic energy function* for a point-wise particle associated with the metric $\mathbb{G}$ is denoted by $K : TM \to \mathbb{R}$ and is defined by:

$$K_x(v) \stackrel{\mathrm{def}}{=} \frac{1}{2}\mathbb{G}_x(v, v) \text{ for } (x, v) \in TM, \tag{3}$$

where the mass of the particle has been assumed unitary for simplicity. On a Riemannian manifold, the metric tensor is positive-definite, hence, on every trajectory $\gamma : \mathbb{I} \to M$, it holds that the kinetic energy takes non-negative values, namely $K_\gamma(\dot{\gamma}) \geq 0$. The *potential energy function* $V : M \to \mathbb{R}$ depends on the coordinate $x \in M$ only. In the absence of any external solicitation, the dynamical system generates a trajectory $\gamma : \mathbb{I} \to M$ that follows the landscape of the potential energy function. In addition, in the context of dynamical systems subjected to *external driving forces on manifolds*, a force at a point $x \in M$ is a tangent vector, namely:

$$f_x : T_xM \times \mathbb{I} \to T_xM. \tag{4}$$

At a point $\gamma(t) \in M$ of a trajectory, the external force may depend upon the instantaneous velocity $\dot{\gamma}(t) \in T_{\gamma(t)}M$ and on the time t.

On a Riemannian manifold M, whose tangent bundle is endowed with an inner product $\mathbb{G}$, the extended stationary-action principle that governs the evolution of a dynamical system reads:

$$\delta \int_{\mathbb{I}} (K_\gamma(\dot{\gamma}) - V_\gamma)\mathrm{d}t + \int_{\mathbb{I}} \mathbb{G}_\gamma(f_\gamma, \delta\gamma)\mathrm{d}t = 0, \tag{5}$$

where symbol δ denotes variation. The integrand in the leftmost integral represents the classical *Lagrangian function* of the particle and its integral represents the *total action* of the particle. The rightmost integral represents the variation of the particle's energy due to the external driving force

and represents an extension of the classical stationary-action principle introduced in the advanced analysis of fluid dynamics [11]. On each interior point of the trajectory $t \in \mathbb{I}$, the variation $\delta\gamma \in T_\gamma M$ is arbitrary, while at the boundaries of the trajectory it vanishes to zero. (For a reference on the calculus of variation on smooth manifolds, readers might want to consult the paper [12].)

Applying the calculus of variation to the extended stationary-action principle (5) leads to the equation of motion of a particle sliding on a smooth Riemannian manifold. In particular, in the spacial case that the external forcing term is absent, namely $f_\gamma = 0$, and the potential energy function is everywhere constant, the stationary-action principle (5) leads to the second-order differential equation:

$$\bar{\nabla}_{\dot\gamma}\dot\gamma = 0, \tag{6}$$

where the symbol $\bar\nabla$ is known as 'covariant derivative' in differential geometry and represents the amount of change of a tangent vector field along a given tangent direction. The formal condition (6) may be phrased as "the instantaneous velocity of a trajectory stays constant along its own direction" or, equivalently, the trajectory γ, which is the solution of the equation (6), translates parallely to itself. This is precisely the characteristic of a straight line in the space $\mathbb{R}^3$, which stays the same when translated along its own direction. Hence, the above equation admits, as a solution, the mathematical entity that is equivalent to a straight line on a curved manifold, termed *geodesic line*. In physical terms, a geodesic line is the path followed by a particle sliding on a curved manifold freely (i.e., without any solicitation) under the effect of the initial momentum and represents a uniform motion on manifolds. A geodesic line starting from a point $x \in M$ with initial velocity $v \in T_xM$ may be written as $\gamma(t) = \exp_x(tv)$, for $t \in [0,\ a]$, with $a > 0$. The function $\exp$ is termed *exponential map*. Since a geodesic line minimizes the kinetic energy, it also represents the shortest path joining two given points on a connected Riemannian manifold.

Upon embedding the manifold M into an Euclidean space (of sufficient dimension), the differential equation (6) may be written as the dynamical system:

$$\left\{\begin{array}{rcl} \dot x &=& v, \\ \dot v &=& -\Gamma_x(v,v), \end{array}\right. \tag{7}$$

where the term Γ denotes the 'Christoffel symbol'. Upon performing the embedding, the variables $x(t)$ and $v(t)$ become free to evolve on the Euclidean space and, in order for the trajectory to lay over the manifold M, the acceleration $\dot v$ needs to adapt point-by-point through the term $-\Gamma_x(v,v)$. In fact, the whole term $\dot v + \Gamma_x(v,v)$ is termed *geometric acceleration of an embedded geodesic*. Note that, since on a geodesic the instantaneous geometric velocity is constant, the instantaneous geometric acceleration is zero.

When the potential energy function V and the external force are present, they manifest themselves in the equation of motion through two additional terms:

$$\left\{\begin{array}{rcl} \dot x &=& v, \\ \dot v &=& -\Gamma_x(v,v) - \nabla_x V_x - f_x. \end{array}\right. \tag{8}$$

The term $\nabla_x V_x \in T_xM$ represents the *Riemannian gradient* that is used to compute the directional derivative of a function on a manifold. (Namely, given a point $x \in M$ and a tangent direction whatsoever $w \in T_xM$, the directional derivative of the potential V_x along the direction w would compute as $\mathbb{G}_x(\nabla_x V_x, w)$.)

For comparison purpose, the dynamical system (8) may be rewritten as:

$$\ddot\gamma + \Gamma_\gamma(\dot\gamma,\dot\gamma) = -\nabla_\gamma V_\gamma - f_\gamma(\dot\gamma, t). \tag{9}$$

By comparing the second-order differential equation (9) on an embedded manifold with the Newton's law (1), it can be readily appreciated how the term $\ddot\gamma + \Gamma_\gamma(\dot\gamma,\dot\gamma)$ represents the geometric acceleration of the particle sliding on the manifold M, the term $-\nabla_\gamma V_\gamma$ denotes conservative forcing, and the term f_γ accounts for a possibly dissipative (or braking) force acting upon the particle.

In the special case that $M = \mathbb{R}^n$, hence $T_xM \cong \mathbb{R}^n$, and $\mathbb{G}_x(v,w) = v^T w$, it holds that $\Gamma = 0$ everywhere on M, $\nabla_x V_x = (J_V)_x$ (the classical Euclidean gradient or Jacobian). Hence, the dynamical system (8) simplifies into:

$$\left\{\begin{array}{rcl} \dot x &=& v, \\ \dot v &=& -(J_V)_x - f_x. \end{array}\right. \tag{10}$$

The dynamical system (10) accounts for well-known oscillator models, such as the simple pendulum (with $M = \mathbb{R}$):

$$\left\{\begin{array}{rcl} \dot x &=& v, \\ \dot v &=& -\frac{dV_x}{dx}, \end{array}\right. \tag{11}$$

with potential energy function $V : \mathbb{R} \to \mathbb{R}$ defined as $V_x \stackrel{\text{def}}{=} \Omega_0^2(1 - \cos x)$; the van der Pol oscillator:

$$\left\{\begin{array}{rcl} \dot x &=& v, \\ \dot v &=& -\frac{dV_x}{dx} - \mu(x^2-1)v, \text{ with } \mu > 0, \end{array}\right. \tag{12}$$

where the potential $V : \mathbb{R} \to \mathbb{R}$ is defined as $V_x \stackrel{\text{def}}{=} \frac{1}{2}\Omega_0^2 x^2$ and $\Omega_0 > 0$; the Lorenz oscillator:

$$\left\{\begin{array}{rcl} \dot x &=& v, \\ \dot v &=& -C_x v,\ C_x \stackrel{\text{def}}{=} \left[\begin{array}{ccc} \sigma & -\sigma & 0 \\ x_3 - \rho & 1 & x_1 \\ -x_2 & -x_1 & \beta \end{array}\right], \end{array}\right. \tag{13}$$

where $x \stackrel{\text{def}}{=} [x_1\ x_2\ x_3]^T$, and σ, β and ρ are constant parameters, for some values of which the Lorenz system becomes chaotic.

2.2 Potential energy functions and forcing terms

In order to complete the picture of non-linear oscillators on Riemannian manifolds offered within the present paper, it is necessary to discuss on the form of the involved potential energy function V_x and of the external forcing term f_x.

In the present context, the force $f_x = f_x(v,t) \in T_xM$ will essentially represent damping effects. In particular:

- *Friction-type damping*: This kind of damping generalizes the Rayleigh damping and is expressed by the forcing term:

$$f_x = -\mu \|v\|_x^{2(\epsilon-1)} v, \tag{14}$$

  with $\epsilon \geq 1$ being a damping coefficient and $\mu \geq 0$ being a viscosity coefficient.
- *Non-linear damping*: It generalizes the nonlinear damping term that appears in the van der Pol system and in the Lorenz oscillator, and assumes the expression:

$$f_x = -\varphi_x(v), \tag{15}$$

  where the map φ is such that, for each $x \in M$, $\varphi_x : T_xM \to T_xM$ is a linear map (endomorphism).

In order to design potential energy functions that might give rise to complex trajectory patterns, it pays to retrace the classical potential energy functions and to extend such potentials to a general Riemannian manifold. To this aim, it is necessary to recall the notion of Riemannian distance between two points on a Riemannian manifold. The Riemannian distance $d(x,y)$ between $x \in M$ and $y \in M$ is obtained by means of a rectification formula that returns the length of the shortest path (namely, the geodesic line) on the manifold joining the points x, y, which may be calculated via the inner product. For this reason, d is also termed *geodesic distance*. The Riemannian gradient of the squared Riemannian distance function with respect to its first argument reads:

$$\nabla_x d^2(x,y) = -2 \log_x(y), \tag{16}$$

where the symbol $\log$ denotes the *logarithmic map* and is defined as the inverse function of the exponential map. In general, the logarithmic map is defined only locally around a given point $x \in M$.

A well-documented potential is the one arising in the study of the simple pendulum. In the classical case that $M = \mathbb{R}$, it reads $V_x \propto 1 - \cos x$. Such potential energy function may be extended to a general Riemannian manifold M, endowed with a Riemannian distance function d, as:

$$V_x^{(\mathrm{pen})} \stackrel{\mathrm{def}}{=} \kappa(1 - \cos d(x,r)), \tag{17}$$

with $\kappa > 0$ being a constant parameter and $r \in M$ denoting a reference point. Upon rewriting such potential energy function as $V_x^{(\mathrm{pen})} = \kappa - \kappa \cos[(d^2(x,r))^{\frac{1}{2}}]$ and by using the calculation rule (16), it is readily seen that its Riemannian gradient reads:

$$\nabla_x V_x^{(\mathrm{pen})} = -\kappa \frac{\sin d(x,r)}{d(x,r)} \log_x(r). \tag{18}$$

The potential energy function introduced to state the van der Pol dynamical system in the case $M = \mathbb{R}$ is a quadratic function, namely $V_x \propto x^2$. Such potential energy function may be extended to a Riemannian manifold M as:

$$V_x^{(\mathrm{pol})} \stackrel{\mathrm{def}}{=} \frac{1}{2} \kappa d^2(x,r), \tag{19}$$

with $\kappa > 0$ being a constant parameter and $r \in M$ denoting a reference point. According to the calculation rule (16), its Riemannian gradient reads:

$$\nabla_x V_x^{(\mathrm{pol})} = -\kappa \log_x(r). \tag{20}$$

Likewise, the *Keplerian system* mentioned in [13] may be extended to a general Riemannian manifold by defining the Keplerian potential as:

$$V_x^{(\mathrm{kep})} \stackrel{\mathrm{def}}{=} -\frac{\rho}{d(x,r)} + \eta d(x,r), \tag{21}$$

with $\rho, \eta > 0$. Rewriting the Keplerian potential as $-\rho[d^2(x,r)]^{-\frac{1}{2}} + \eta[d^2(x,r)]^{\frac{1}{2}}$ and invoking again the calculation rule (16) together with the rule of derivation of composite functions, its Riemannian gradient was calculated to be:

$$\nabla_x V_x^{(\mathrm{kep})} = -\left[\frac{\rho}{d^3(x,r)} + \frac{\eta}{d(x,r)}\right] \log_x(r). \tag{22}$$

The shape of the potential energy function has a strong influence on the evolution of the state of the non-linear damped oscillator (8).

3. Simulation of the non-linear oscillator on manifold

In order to implement the devised theory on a computing platform, it is necessary to discretize in time the second-order dynamical system (8). In particular, it is necessary to devise a numerical solver that approximates numerically the solution of the two differential equations in the variables x and v: Such a numerical solver will output a sequence $k \mapsto (x_k, v_k)$ of tangent-bundle pairs at any snapshot k. Recall that the variable x does not belong to a 'flat' space like $\mathbb{R}^n$, but it belongs to a curved space, hence, the classical Euler-like approximations cannot be invoked.

3.1 Time-discretization of the continuous-time oscillator

The Euler rule to solve numerically a differential equation of the type $\dot{x} = v$ prescribes to move forward the point x_k to the point x_{k+1} along a straight line of direction v_k. By replacing the notion of straight line with the notion of

geodesic line, one gets a so-called Euler-Riemann stepping method, that reads:

$$x_{k+1} = \exp_{x_k}(\Delta t \cdot v_k), \tag{23}$$

where the time-interval $\Delta t > 0$ denotes a time-discretization stepsize.

In order to solve the differential equation on the variable v, we need to introduce a further differential-geometric notion, namely *parallel transport* along a geodesic line. Parallel transport on a tangent bundle associated with a Riemannian manifold serves to move around tangent vectors by making sure that they stay tangent. If we start with a tangent vector $w \in T_xM$ at a point $x \in M$ of an embedded manifold M and we translate *rigidly* such tangent vector within the embedding space, along a geodesic line, to another point $y \in M$, most probably the result of such rigid translation will not belong to the tangent space T_yM. In order to make sure that a vector gets translated along a curve in a way that keeps it tangent to the manifold at any point, a special operator needs to be introduced. The parallel transport operator $\mathrm{P}^{x\to y} : T_xM \to T_yM$ associated with the geodesic curve connecting the endpoints $x, y \in M$ is a linear isomorphism that depends smoothly on its arguments and is such that $\mathrm{P}^{x\to x}$ is the identity map in T_xM and $\mathrm{P}^{z\to y} \circ \mathrm{P}^{x\to z} = \mathrm{P}^{x\to y}$ for every $x, z, y \in M$ belonging to the same geodesic line. By definition, the parallel transport is an isometry, namely, given tangent vectors $u, w \in T_xM$, it holds that $\mathbb{G}_y(\mathrm{P}^{x\to y}(u), \mathrm{P}^{x\to y}(w)) = \mathbb{G}_x(u, w)$. On the basis of the above-defined parallel transport operator, the second differential equation of the system (8) may be solved numerically through the simulation rule:

$$v_{k+1} = \mathrm{P}^{x_k\to x_{k+1}}(v_k + \Delta t \cdot (-\nabla_x V_x - f_x)). \tag{24}$$

Note that the Christoffel term is not necessary anymore because the numerical solver ensures the vector v_{k+1} to be tangent to the manifold M at the point x_{k+1}.

3.2 Numerical simulation: The unit hypersphere

The unit hypersphere is defined as:

$$\mathrm{S}^{n-1} \stackrel{\mathrm{def}}{=} \left\{ x \in \mathbb{R}^n | x^T x = 1 \right\}, \tag{25}$$

where the superscript ^{T} denotes matrix transpose. The manifold S^1 coincides with the unit circle in the plane $\mathbb{R}^2$, while the manifold S^2 coincides with the three-dimensional sphere in $\mathbb{R}^3$ (which constitutes its 'natural embedding'). The tangent spaces read:

$$T_x\mathrm{S}^{n-1} = \{ v \in \mathbb{R}^n | v^T x = 0 \}. \tag{26}$$

Since a vector x in an embedded hypersphere may be regarded as a radial vector, a tangent space at x is readily seen to be spanned by all of those vectors that are normal to the radial vector x, namely, which are tangent to the sphere at the point x. A number of engineering applications are based on the manifold S^{n-1} as, for instance, blind channel deconvolution [14], robust constrained beamforming algorithms [15] and modeling of MIMO broadcast channels under linear filtering [16].

For the unit-hypersphere endowed with the canonical metric $\mathbb{G}_x(w, u) = w^T u$, it holds:

$$\Gamma_x(v, v) = x\|v\|^2, \tag{27}$$

$$\mathrm{P}^{x\to y}(w) = \left[I_n - \frac{(I_n - xx^T)yy^T}{1 + x^T y} - xy^T \right] w, \tag{28}$$

where symbol $\|\cdot\|$ denotes vector 2-norm, symbol I_n denotes a $n \times n$ identity matrix, $x \in \mathrm{S}^{n-1}$, $w, v \in T_x\mathrm{S}^{n-1}$, in the expression of the geodesic it is assumed that $v \neq 0$ and it is assumed that $x^T y \neq -1$ in the expression of the parallel transport.

By gathering the expressions of the Christoffel operator and of the Riemannian gradient of the potential energy function, the following dynamical system with non-linear damping is obtained:

$$\left\{ \begin{array}{rcl} \dot{x} & = & v, \\ \dot{v} & = & -\|v\|^2 x - \nabla_x V_x - \mu\|v\|^{2(\epsilon-1)} v - \varphi_x(v), \end{array} \right. \tag{29}$$

where $x(0) = x_0 \in \mathrm{S}^{n-1}$ and $v(0) = v_0 \in T_{x_0}\mathrm{S}^{n-1}$.

According to the expressions of the geometric quantities of interest recalled above, the dynamical system (29) may be simulated through the equations:

$$\left\{ \begin{array}{l} x_{k+1} = x_k \cos(\Delta t \cdot \|v_k\|) + v_k \|v_k\|^{-1} \sin(\Delta t \cdot \|v_k\|), \\ v_{k+1} = \mathrm{P}^{x_k\to x_{k+1}}(v_k - \Delta t \cdot [\mu\|v\|^{2(\epsilon-1)} + \varphi_{x_k}(v_k) \\ \qquad\qquad + \nabla_{x_k} V_x]), \end{array} \right. \tag{30}$$

with $k = 0, 1, 2, \ldots$.

In the case of the hypersphere, the squared Riemannian distance between two sufficiently-close points $x, y \in \mathrm{S}^{n-1}$ may be expressed in closed form as:

$$d^2(x, y) = \arccos^2(x^T y), \tag{31}$$

where the symbol '$\arccos$' denotes the inverse cosine function. Likewise, both the exponential map and its inverse, the logarithmic map, may be expressed in closed form as:

$$\exp_x(v) = x\cos(\|v\|) + \|v\|^{-1}\sin(\|v\|)v, \tag{32}$$

$$\log_x(y) = (I_n - xx^T)y \cdot d(x, y) \cdot \csc d(x, y), \tag{33}$$

where the symbol 'csc' denotes the cosecant function. The above formulas help working out the expression of the Riemannian gradient of potential energy functions that depend on the squared Riemannian distance (such as the 'quadratic' potential $V_x = \frac{1}{2}d^2(x, r)$, with $r \in \mathrm{S}^{n-1}$ being a reference point on the hypersphere).

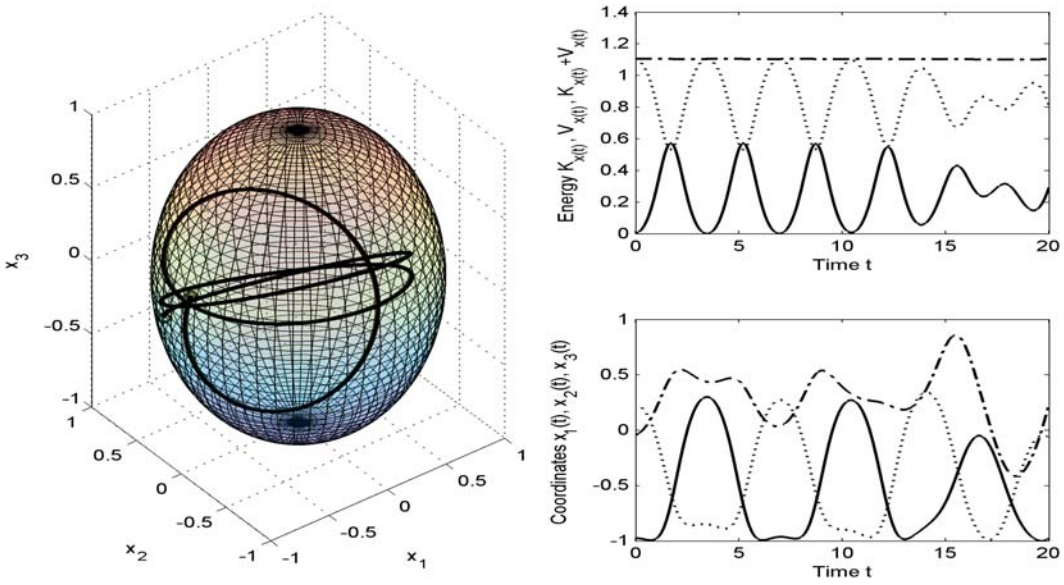


Fig. 1: Exemplary numerical simulation result. A-periodic dynamics on the manifold $M = \mathrm{S}^2$ embedded in the Euclidean space $\mathbb{R}^3$. The left-hand panel illustrates the trajectory over the sphere (the circle denotes the initial point and the diamond denotes the final point). The upper panel on the right-hand side illustrates the values of the kinetic energy (solid line), the potential energy (dotted line) and the total energy (dot-dashed). The lower panel on the right-hand side illustrates the trajectory in terms of the three embedded coordinates in x.

3.3 Numerical simulation on the sphere

The present Subsection illustrates the behavior of the discrete-time dynamical system (30) over the sphere S^2. Such manifold affords graphical rendering of the resulting trajectories. In all the following simulations, the time-discretization interval was set to $\Delta t = 0.001$, the damping coefficient was chosen as $\epsilon = \frac{3}{2}$, the initial state x_0 was randomly chosen over the sphere and the initial velocity v_0 was randomly chosen in $T_{x_0}\mathrm{S}^2$.

The Figure 1 illustrates a configuration of the oscillator that generates a non-periodic trajectory. In this simulation, the potential energy function was set to $V_x = \frac{1}{2}x^T S x$, with S being a random symmetric, positive-definite matrix. The damping term contains again the purely dissipative term $-\mu\|v\|v$, with $\mu = 0.001$. It is readily appreciated how the oscillator, driven by a quadratic potential energy function in the embedding space, generates a trajectory over the manifold S^2 exhibiting a complex pattern.

The Figure 2 illustrates a configuration of the oscillator (30) that generates a periodic, self-sustaining motion. In this simulation, the potential energy function was set to $V_x = \frac{\kappa}{2}d^2(x, r)$, with r randomly chosen over the unit sphere and $\kappa = 2$. The external forcing term contains again the purely dissipative term $-\mu\|v\|v$, with $\mu = 0.001$, and an active van der Pol damping term $-\rho(d^2(x, r) - 1)v$, with $\rho = 0.8$. For this reason, the corresponding oscillator, after an initial transient, exhibits an oscillating periodic trajectory.

4. Conclusions

The presented research endeavor aimed at extending classical second-order dynamical systems to Riemannian manifolds in order to generalize the notion of non-linear oscillators. The equations governing the time-evolution of a dynamical system on a curved Riemannian manifold were derived on the basis of a classical minimal-action variation principle modified to include non-conservative terms. Such kind of non-linear dynamical systems are able to generate deterministic trajectories exhibiting complex patterns on curved manifolds.

In order to simulate on a computer platform the devised non-linear oscillators, the continuous-time equations were discretized in time through a numerical solver tied to the underlying geometric structure of the manifold that the system's trajectory lays on. The presented numerical experiments illustrate the behavior of the non-linear oscillator on the unit sphere embedded in the ordinary space and showed two kinds of trajectory, namely a periodic self-sustained trajectory and a non-periodic trajectory.

Future research will be directed along the study of possible configurations for the potential energy function V_x and for the forcing term f_x that can give rise to self-sustained non-periodic oscillations, possibly by means of

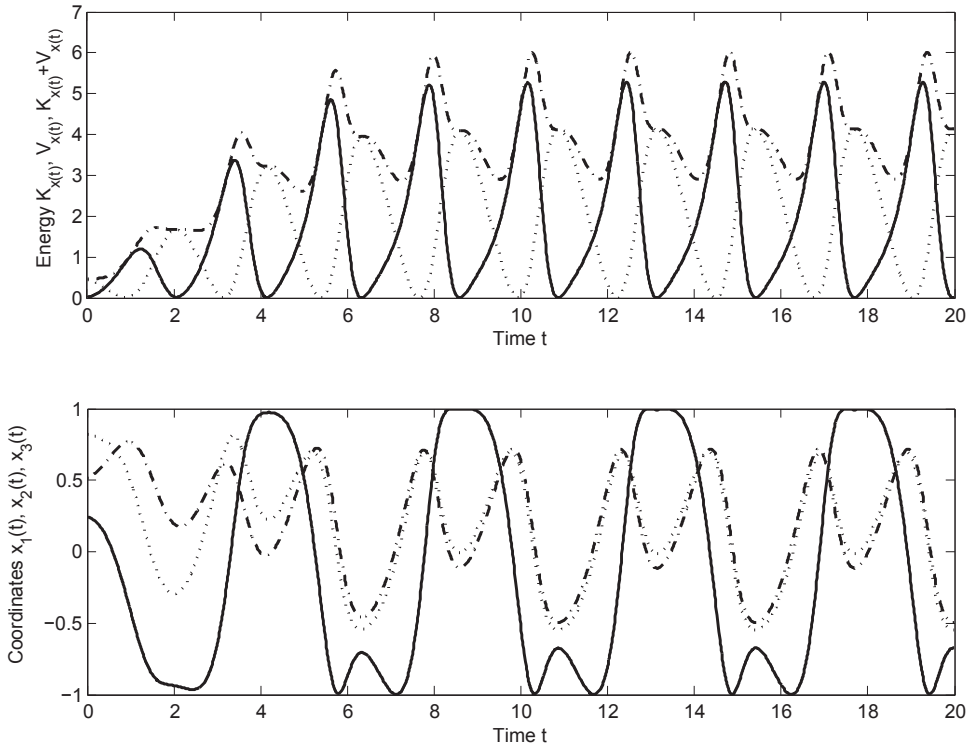


Fig. 2: Periodic, self-sustained, dynamics of a generalized van der Pol oscillator on the manifold $M = \mathrm{S}^2$ embedded in the Euclidean space $\mathbb{R}^3$. The upper panel shows the values of the kinetic energy (solid line), the potential energy (dotted line) and the total energy (dot-dashed line). The lower panel illustrates the trajectory in terms of the three embedded coordinates.

active damping terms. Of particular interest are those choices of the parameters that generate *chaotic* oscillations.

Acknowledgment

The authors thank C. Leonori and M. Leonori, from the School of Informatics and Automation of the Università Politecnica delle Marche, who contributed in the coding of the software that simulates the non-linear oscillator system on the unit sphere.

References

[1] C.K. Tse and F. Lau, *Chaos based digital communication systems: operating principles, analysis methods, and performance evaluation*, New York: Springer-Verlag 2003.

[2] L.A.B. Torres and L.A. Aguirre, "Transmitting information by controlling nonlinear oscillators," *Physica D*, vol. 196, pp. 387 – 406, 2004.

[3] G. Jakimoski and L. Kocarev, "Chaos and cryptography: block encryption ciphers based on chaotic maps," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, pp. 163 – 169, 2001.

[4] W. Zhao, Y. Hu, L. Zhang, and L. Wang, "Design and CPG-based control of biomimetic robotic fish," *IET Control Theory and Applications*, vol. 3, pp. 281 – 293, 2009.

[5] B. van der Pol and J. van der Mark, "The heartbeat considered as a relaxation oscillator and an electrical model of the heart," *Philosophical Magazine Supplement*, vol. 6, pp. 763 – 775, 1928.

[6] K. Grudziński and J.J. Żebrowski, "Modeling cardiac pacemakers with relaxation oscillators," *Physica A: Statistical Mechanics and its Applications*, vol. 336, pp. 153 – 162, 2004.

[7] M.R. Roussel and A.U. Igamberdiev, "Dynamics and mechanisms of oscillatory photosynthesis," *Biosystems*, vol. 103, pp. 230 – 238, 2011.

[8] J.O. Maaita, I.M. Kyprianidis, Ch.K. Volos, and E. Meletlidou, "The study of a nonlinear Duffing type oscillator driven by two voltage sources," *Journal of Engineering Science and Technology Review*, vol. 6, pp. 74 – 80, 2013.

[9] D.M. Kaufman, and D.K. Pai, "Geometric numerical integration of inequality constrained, nonsmooth Hamiltonian systems," *SIAM Journal on Scientific Computing*, vol. 30, pp. 134 – 147, 2007.

[10] K. Modin, and G. Söderlind, "Geometric integration of Hamiltonian systems perturbed by Rayleigh damping," *BIT Numerical Mathematics*, vol. 51, pp. 977 – 1007, 2011.

[11] S. Shorek, "A stationarity principle for non-conservative systems," *Advanced Water Resources*, vol. 7, pp. 85 – 88, June 1984.

[12] S. Fiori, "Solving minimal-distance problems over the manifold of real symplectic matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 32, pp. 938 – 968, 2011.

[13] F.J. Molero, M. Lara, S. Ferrer, and F. Cèspedes, "2-D Duffing oscillator: Elliptic functions from a dynamical systems point of view," *Qualitative Theory of Dynamical Systems*, vol. 12, pp. 115 – 139, April 2013.

[14] S. Fiori, "Geodesic-based and projection-based neural blind deconvolution algorithms," *Signal Processing*, vol. 88, pp. 521 – 538, March 2008.

[15] S. Fiori, "Neural minor component analysis approach to robust constrained beamforming," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 150, pp. 205 – 218, August 2003.

[16] R. Hunger, P. de Kerret, and M. Joham, "An algorithm for maximizing a quotient of two Hermitian form determinants with different exponents," in *Proc. of ICASSP 2010*, 2010, pp. 3346 – 3349.

Maximizing the Speed of Influence in Social Networks

Yubo Wang and Robert Chun
Department of Computer Science, San Jose State University, San Jose, California, US

Abstract-*Influence maximization is the study of seed-node selection in a social network in order to achieve the maximized number of influenced nodes. Previous studies focused on three areas, i.e., designing propagation models, improving seed-node selection algorithms and exploiting the structure of social networks. However, most of these studies ignored the time constraint in influence propagation. Here, I studied how to maximize the speed of influence propagation. I extended the classic Independent Cascade model to a Continuous Dynamic Extended Independent Cascade (CDE-IC) model. In addition, I proposed a novel heuristic algorithm and evaluated the algorithm using two large academic collaboration data sets. The new algorithm is 10%-20% faster in influence propagation than previous classic heuristic algorithms on the CDE-IC model. Furthermore, I gave solutions to calculate propagation probability between adjacent nodes by exploiting the structure of social network.*

Keywords: Social network, Influence speed maximization, Diffusion model, heuristic algorithm

1. INTRODUCTION

Social networks provide a great opportunity to promote new products or ideas because of the large number of users and the high frequency of communication. In viral marketing strategy, a company invites some initial users, i.e. the seed nodes, to try its new products or technologies and hope that these initial users would give a positive feedback. By the power of word-of-mouth, these users may affect their neighbors in a social network. These affected neighbors may subsequently affect their own neighbors. Therefore, in a fashion similar to viral infection, the influence propagates in the social network. The challenge in viral marketing strategy is how to select the seed nodes to maximize the number of affected nodes.

The large scale of social networks and their complicated structures made it challenging to select the right seed nodes. Influence maximization was first proposed as an algorithm problem by Domingos and Richardson in a study of viral marketing[1, 2]. Kempe, Kleinberg and Tardos provided a foundation to solve this problem[3]. They proved that the problem of optimization in seed-node selection is NP-hard. They also presented the first provable approximation solution, which is within 63% (1-1/e) of optimal. One big drawback in this solution is low efficiency. Even finding a small seed-node set in a moderately large network (e.g. 15000 nodes) would take days to finish. Several following studies have been carried to improve the efficiency of seed-node selection algorithms. Leskovec, Krause and Guestrin proposed a nearly optimal algorithm called Cost Effective Lazy Forward (CELF) algorithm[4], which was 700 times faster. Goyal, Lu and Lakshmanan proposed an algorithm called CELF++, which was 35%-55% faster than CELF[5].

Chen, Wang and Yang[6] tackled the efficiency issue of seed-node selection by improving heuristics methods. Their new heuristics method achieved a nearly matched result comparing to greedy algorithms, but with a running time of more than six orders of magnitude faster.

All the previous studies focused on the space maximization of influence, i.e. how to maximize influence propagation in a social network without time constraint. In this paper, I will study the speed maximization of influence propagation, i.e. how to maximize influence propagation in a social network in a given time frame.

Here, I showed how I extended a classic IC model to Continuous Dynamic Extended IC (CDE-IC) model. In addition, I proposed a novel heuristic seed-node selection algorithm. I then gave formulas to calculate propagation probability between adjacent nodes. In order to test my new algorithm, I compared this algorithm with three other most popular heuristic algorithms in two data sets. The result showed that even a small modification to the existing algorithms could lead to a big boost to the qualification of seed nodes selection.

2. RELATED WORK

Studies on influence maximization in social networks are mainly focused on three areas, designing propagation models, improving seed-node selection algorithms and exploiting social networks structure.

2.1 Propagation Models

The two basic models of social networks are Linear Threshold (LT) model and Independent Cascade (IC) model. Many other models extended from these two basic models under different conditions.

2.1.1 Linear Threshold (LT) Model

LT Model was proposed by Granovetter and Schelling to simulate influence propagation in social networks [7, 8]. In LT model, each node u has a threshold θ_u ($0<\theta_u<1$) that defines the minimum requirement of its active neighbor nodes set. When the total weights of all active neighbor nodes are greater than θ_u, an inactive node u switches to an active status.

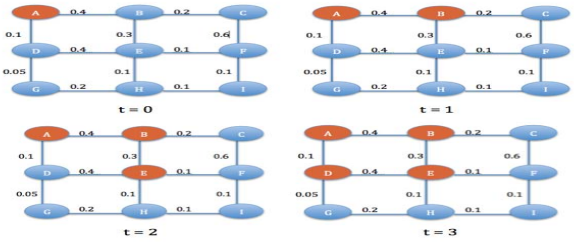


Figure 1: Influence propagation in LT model

The process of influence propagation in LT mode is shown as in Figure 1. Each node has the same threshold value of 0.3. The weight between any two nodes is labeled at the edge. At time 0, node A is selected as the seed node with an active status. Node A will try to propagate influence to its neighbor nodes of node B and Node D. Edge AB has a weight value of 0.4 and edge AD has a weight value of 0.1. In this condition, only node B satisfies the condition of switching from an inactive status to an active status as the total weight of active neighbor nodes of Node B is greater than the threshold of 0.3. Therefore, Node B turns into an active status at time 1. The total weights of active neighbors of node D are 0.1 and node D remains inactive at time 1. The following steps are similar until there are no more nodes that satisfy the condition to switch and the process of influence propagation stops.

There are two interesting observations in this propagation process. First, although some people will not accept new technologies at first, they are likely to change their minds as more of their neighbors accept the new technologies. These people are represented by node D in the example. Secondly, although the weight between node C and node F is higher than the threshold, they have no chance to switch to active status. Their active neighbor nodes are not powerful enough to propagate the influence to them. Node C and node F represent the users in social networks who are eager to accept new products. However, because inappropriate seed nodes are selected, the propagation process fails to discover these users. We have to carefully consider this during the seed nodes selection phase.

2.1.2 Independent Cascade Model

The Independent Cascade (IC) model was proposed by Goldenberg, Libai, and Muller[9]. In IC model, the process of influence propagation in social networks can be illustrated as the following steps. An initial seed-node set is selected at time 0. If node A becomes active at time t, it will try to affect its inactive neighbors at time t+1 with a probability of p. Node A will only have one chance to affect its neighbors, and will not try again whether it succeed or not. If an inactive node is connected with more than one newly active node, these newly active nodes will try to influence the inactive node in a random sequence. In addition, the result of influence propagation between two nodes is not affected by actions of other nodes.

The process of influence propagation In IC model is shown in Figure 2 and Figure 3, each representing an outcome from a single run. In both scenarios, node A is selected as the seed node. However, the end results are different because of the randomness in influence propagation in IC model. The influence reaches to 4 other nodes (D, E, F and H) in one run (Figure 2) and 5 other nodes (B, C, E, F and H) in the other run (Figure 3). Therefore, the experiments need to run multiple times in IC model to obtain an accurate estimation of influence propagation with a specific seed node selection.

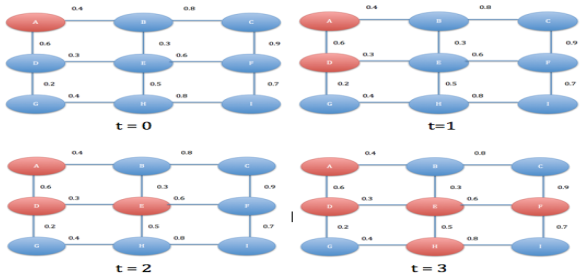


Figure 2: Influence propagation in IC 1st try

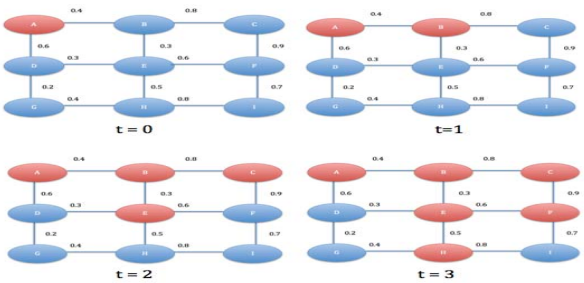


Figure 3: Influence propagation in IC 2nd try

2.1.3 Extensional IC Model

Wang, Qian, and Lu proposed an Extensional Independent Cascade (EIC) model[10]. In the EIC model, the influence propagation between two adjacent nodes is divided into two phases involving a spreading phase from an active node and an adopting phase from the inactive node. In the spreading phase, an active node decides whether to spread the influence to its neighbor nodes based on a spreading probability of p_s. If the active node decides to spread out the influence to its neighbor nodes, the adopting phase is similar as in the original propagation process in classic IC model. In the adopting phase, each inactive node decides independently whether to adopt the influence based on an adopting probability of p_a. The EIC model added one more step in the process of influence propagation. The new propagation probability (p_p) in EIC model is the product of probability in spreading phase (p_s) and probability in the adopting phase (p_a), as in formula 1 $p_p = p_s \times p_a$.

2.2 Seed-node Selection Algorithms

Seed-node selection algorithms play a key role in influence maximization. There are two metrics to measure a seed node selection algorithm, efficiency and quality.

2.2.1 Hill-Climbing Algorithm

Kempe et al.[3] used a greedy algorithm named Hill-Climbing algorithm. They proved that the Hill-Climbing algorithm guarantees to achieve an approximation solution with a factor $(1- 1/e - \varepsilon)$ to the optimal solution in both the LT Model and the IC Model. Here e is the natural logarithm base and ε is any small positive real number. This algorithm is based on the theory of submodular functions[11]. In order to obtain an accurate estimation, Kempe et al. run Monte Carlo simulation sufficient number of times (20,000) for selecting each seed node. Although greedy algorithm guarantees the quality of seed nodes selection, it is not efficient enough for large-scale social networks. Many later studies tried to improve the efficiency of this algorithm, including CELF Algorithm and CELF++ Algorithm.

2.2.2 CELF and CELF++ Algorithm

The biggest drawback of the Hill-Climbing greedy algorithm is low efficiency. Leskovec et al. tried to solve this problem using an improved greedy algorithm named Cost-Effective Lazy Forward (CELF)[4]. Their experiment showed that they would get a near optimal solution while being 700 times faster than the Hill-Climbing greedy algorithm. Further effort of reducing the running time of the greedy algorithm was carried by Goyal et al.[5]. They proposed an improved CELF algorithm called "CELF++" that further decreases the running time by 35%-50% compared to the CELF algorithm. Despite those improvements, use of greedy algorithms in large data sets is still limited by the relatively low efficiency.

2.2.3 Heuristic Algorithm

In contrast to greedy algorithms, heuristic algorithms may not provide the best result, but they can obtain an acceptable result in much less time. Two widely used heuristic algorithms are Degree-Centered and Distance-Centered.

In Degree-Centered algorithm, the nodes that have a large number of connections in a social network, i.e. the high degree nodes, are deemed as influential nodes. The more out-neighbors a node has, the more influence it is believed to have in a social network. This is an intuitive assumption. However, a known phenomenon in social network is that high degree nodes tend to connect to each other. Therefore, if only high degree nodes are selected as seed nodes, these seed nodes will have a large overlap with each other. The overlapped nodes will not bring any additional value to the set of seed nodes to maximize influence propagation. In Distance-centered algorithm, influential nodes are the nodes that have a smaller average distance to other nodes.

Although Degree-Centered algorithm is better than other heuristic algorithms[3], it is still not as good as greedy algorithms. In general, heuristic algorithms were not studied extensively in research field because of the low expectation of quality. However, Chen et al. proposed an improved heuristic algorithm in IC Model[6]. This improved heuristic algorithm performed comparably to greedy algorithms. Importantly, this algorithm significantly reduced the time in the seed-node selection with six orders of magnitude.

In this new heuristic algorithm, Chen et al. introduced the concept of "discount" to the degree of a node. The logic is that when a node is selected as a seed node, its neighbor nodes will become less influential to the social network. Therefore, there should be a discount on the neighbor nodes. They proposed two methods to discount the degree of a node. The first method is called "Single Discount", in which the degree of all neighbor nodes of a selected seed node is reduced by one. The second method is more complicated in that the effect of each newly activated node to its inactive neighbor nodes will be calculated individually.

3. SOLUTION FRAMEWORK

In this section, I study how to maximize the speed of influence propagation in social networks. I extended the classic IC model to a Continuous Dynamic Extended Independent Cascade (CDE-IC) model. In addition, I proposed a novel heuristic algorithm comparing with previous classic heuristic algorithms on the CDE-IC model in two large academic collaboration data sets. Furthermore, I discuss how to decide the parameters in the propagation model by exploiting the structure of social networks.

3.1 CDE-IC Model

The Extensional Independent Cascade (EIC) model extends the process of influence propagation from one phase in classic IC model to two phases involving a spreading phase and an adopting phase[10]. The EIC model with two phases in propagation process is a good extension to IC model, and it is more close to reality in social networks. However, The EIC model has the same drawbacks as the classic IC model, in that the propagation process is one-time and static. One-time refers to the assumption that a newly activated node will only try to propagate the influence to its neighbors once. If a node turns into active status in step t, it will try to propagate its influence to its inactive neighbor nodes in step t+1. Static in EIC model means that the probability of propagation does not change with time.

In this paper, I propose a new improvement on the EIC model to take into consideration of continuous influence, and also the dynamic nature of influence propagation. In this Continuous Dynamic Extended IC (CDE-IC) Model, an active node will keep propagating influence to its neighbor nodes until there are no more inactive nodes. Furthermore, the probability of propagation between two nodes will change with time. The process of influence propagation in CDE-IC model is shown in Figure 4. Suppose all edges in Figure 4 have a probability of ½. At time 0, node A is selected as a seed node. At time 1, node B turns active under node A's influence. At time 2, node B propagates influence to its neighbor node C and node E. At time 3, although node A fails to influence node D at time 0, it gets another opportunity and succeeds. Node D becomes active, so does node F. The influence process stops at time 3.

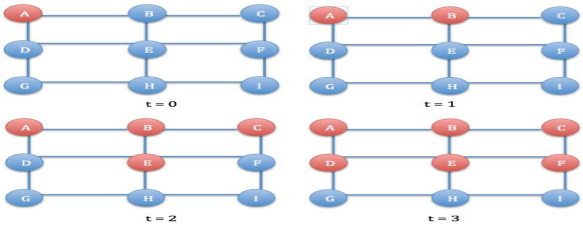


Figure 4: Influence propagation in CDE-IC model

The key difference between CE-IC Model with classic IC Model or EIC model is that an active node will have multiple changes to propagate its influence. If it fails the first time, it still has a second or a third chance to propagate. In CE-IC model, all activated nodes need to be considered in each step.

3.2 Zero-Discount Algorithm

The Hill-Climbing algorithm and its improvements can provide guaranteed approximation solution to optimal seed node selection, but they are not efficient. Traditional heuristic algorithms such as Degree-Centered algorithms and Distance-Centered algorithms are efficient in seed nodes selection, but they do not have comparable results to greedy algorithm. The quality of Degree-Centered algorithm is lower than that of Hill-Climbing algorithms because there is a high degree of overlapping between high degree nodes. If we only select nodes with the highest degree into the seed nodes set, they will greatly weaken each other in the process of influence propagation.

To minimize overlapping between high degree nodes, I propose Zero-Discount algorithm. In this algorithm, if a node is selected as a seed node, the degree of all of its out-neighbor nodes is set to zero. In CDE-IC model, an active node will continuously propagate its influence to its neighbor nodes. If the probability of propagation between two nodes is p, it only needs 1/p on average attempts to succeed. Therefore, we can safely remove all out-neighbor nodes of a seed node without impairing performance significantly. At the same time, a more evenly distributed seed node set increases the chance of influence propagation in social networks.

3.3 Estimation of Propagation Probability

The propagation probability can be set randomly or given by data mining technology. However, it will be either inaccurate or costly. Here, an estimated probability value is assigned to each edge by analyzing the characters of social networks. Influence sources are categorized into three classes, Star Effect, Peer Pressure and Social Trend.

In CDE-IC model, the propagation probability is the product of spreading probability and adopting probability. A different spreading probability is given to each node according to its degree in social networks. High degree nodes will have higher spreading probability than low degree nodes. A maximum spreading probability p_{max} is given to the node with the highest degree (p_{max}<1) and an isolated node will have a spreading probability of 0. The other nodes will be assigned a probability value based on a linear function. The spreading probability is decided as the following formula 2 by out degrees.

$$p_s = p_{min} + (p_{max} - p_{min}) \quad (Degree(node) - Degree_{min}) / (Degree_{max} - Degree_{min})$$

Adopting probability is estimated according to the source of influence in the following three steps. Firstly, check all in-neighbor nodes of a node. Secondly, calculate how much energy each in-neighbor node spends on it. Thirdly, normalize the total energy. In addition, it has been demonstrated that the probability of influence shows an exponential decay behavior[13]. To model this decay behavior, I introduce a time factor into the estimation of adopting probability in my model. The decay of adopting probability is decided as in the following formula 3:

$$p_a(t) = p_a(t_0) \; e^{-(t-t_0)/\tau}$$

The nodes with high out-degree in social networks represent celebrities. They are active in social networks and more likely to spread new influence to the public. Such nodes have a high spreading probability. However, the probability of adoption is relatively low when people receive influence from such nodes. In contrast, the nodes with low out-degree are average people and they are the main component of most social networks. Such nodes will not publish as many posts as celebrities, but their posts have a bigger influence on their connections. Social Trend also has an influence on the propagation probability. When a new technology was first introduced to the public, it is difficult for people to accept it immediately. As more and more people accept the new technology, the propagation probability will also increase with time. The effect of Social Trend can be estimated by the percentage of active nodes in a social network. The effect of Social Trend will also be added into the estimation of propagation probability.

3.4 Data Sets

In this experiment, two real world data sets from www.arXiv.org, High Energy Physics (hep) data set and Physics (phy) data set, were chosen to test algorithms on the CDE-IC model. The hep data set has 15k nodes and 59k edges, and the phy data set has 37k nodes and 231k edges.

4. EXPERIMENTS

In this paper, I studied how to maximize speed of influence propagation in social networks. I compared four heuristic methods, Random, Distance-Centered, Degree-Centered and my novel Zero-Discount method on the Continuous Dynamic Extended IC (CDE-IC) Model.

I studied the CDE-IC model in two steps. Firstly, I extended the EIC model to Continuous Extended IC (CE-IC) model that studies the continuous influence of an active node. Secondly, I improved the CE-IC model to CDE-IC model, in which the probability of propagation will change with time.

The process of influence propagation is assumed to be discrete, which means each active node can only propagate influence one step further in a unit time. By recording how many nodes are affected in each step, we know both the instant speed and the average speed of the whole process. Each method is run 100 times to obtain an average result.

4.1 Results in CE-IC Model

4.1.1 Comparison of Efficiency

The efficiencies of four algorithms are analyzed by comparing the running time in seed-node selection phase. Seed nodes size is 20 and the time unit is millisecond.

Table 1. Running time of Seed-node Selection Algorithms

	hep	phy
Random	1	1
Distance-Centered	13613	186193
Degree-Centered	45	70
Zero-Discount	143	230

Random method is the fastest among the four methods. It is only related to size of seed nodes. Distance-Centered is the slowest in these four heuristic methods and does not scale well. It takes O(n) for a root to reach all other nodes. The total time to calculate average distance of all nodes is O(mn). Degree-Centered algorithm is much more efficient than the Distance-Centered algorithm. Time complexity of is O(m + n). My novel Zero-Discount heuristic algorithm is an improvement over the Degree-Centered algorithm. It runs slower than Degree-Centered algorithm to check more nodes, but it is still much efficient than Distance-Centered algorithm. In Zero-Discount algorithm, if a node is selected as a seed node, the degree of all its out-neighbor nodes will be set to zero and put to the end of queue at each step. The time complexity of Zero-Discount is O(mlgk+n).

The experiments in seed-node selection phase show that my novel Zero-Discount algorithm has a comparable efficiency to the Degree-Centered algorithm, and significantly better efficiency than Distance-Centered algorithm. Zero-Discount is about 100 times faster than Distance-Centered method.

Table 2. Time Complexity Comparision

Algorithm	Time Complexity
Random	O(k)
Distance-Centered	O(mn)
Degree-Centered	O(m+n)
Zero-Discount	O(mlgk+n)

4.1.2 Comparison of Quality

In this section, I will compare the quality of seed-node selection algorithms by analyzing how fast influence propagates in a social network. I run experiments on two data sets with seed nodes of size 20 and 50 respectively. Each experiment will run in 40 unit time. The total number of active nodes at the end of step 40 reflects the average speed over the process of influence propagation. The number of newly activated nodes in each step reflects the instant speed of influence propagation.

Comparison of the average speed of influence propagation using four heuristic algorithms is shown in Figure 5. The size of seed node is 20. I used hep data set to test the four algorithms in CE-IC model. As expected, Random algorithm has the slowest average speed. Although randomly selecting seed nodes is fast in the seed node selection phase, this method is useless in promoting influence propagation. Degree-Centered is regarded as the best heuristic algorithm and is widely used. It is about 15% faster than Distance-Centered algorithm (Figure 5). Zero-Discount algorithm has the fastest average propagation speed among the four algorithms. Although my algorithm is slower in the seed node selection phase than Degree-Centered algorithm, it increased the propagation speed by 10%.

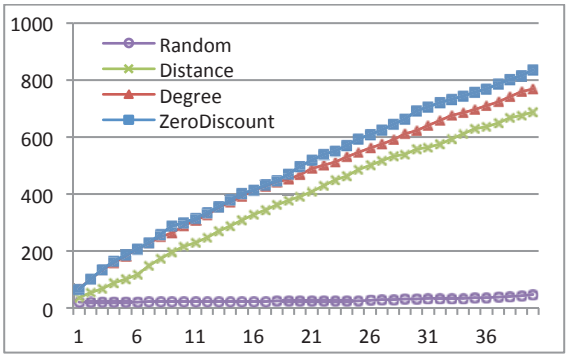


Figure 5: Average Speed in CE-IC Model (hep k=20)

The instant propagation speed is shown in Figure 6. Random selection is still the slowest. There are two important features in this graph. Firstly, both Degree-Centered and Zero-Discount algorithms have significantly fast instant speed at the initial steps of influence propagation. This is generated by inclusion of high degree nodes in the seed nodes and is the major contributor to the fast average speeds seen in Figure 5. At the later steps of influence propagation, the instant speed of Degree-Centered algorithm declined and was similar to Distance-Centered algorithm.

Secondly, although the instant speeds of Zero-Discount algorithm also declined at later steps, there were several minor spikes during influence propagation such like data point 20 and 30. The difference in instant propagation speed between Degree-Centered and Zero-Discount algorithm is that Degree-Centered algorithm only selects high degree nodes, and there is a large overlapping between the high degree nodes. As the influence propagation proceeds, these high degree nodes add little additional value to the propagation process. In contrast, Zero-Discount algorithm tries to diversify seed nodes selection by eliminating neighbor nodes of seed nodes, and adding high degree nodes from isolated sub-networks as seed nodes. As influence

propagation proceeds, these small isolated sub-networks can be connected. That is why Zero-Discount algorithm performs better than Degree-Centered algorithm at later steps of influence propagation.

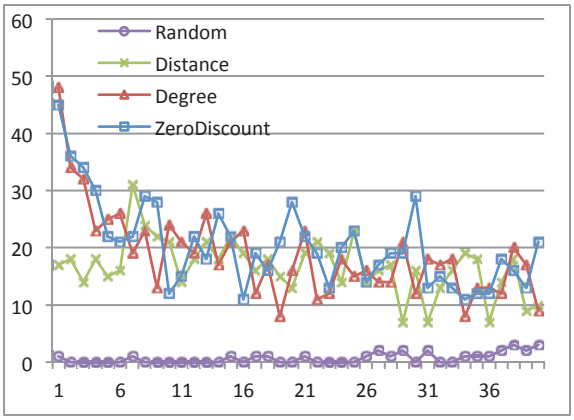


Figure 6: Instant Speed in CE-IC model (hep k=20)

Next, I explored how seed-node size affects the speed of influence propagation. Seed-node size of 50 was used for the same data set hep. The average propagation speeds of the four algorithms are shown in Figure 7. Obviously, increase of seed node size significantly increased propagation speed of all algorithms (Figure 7 vs. Figure 5). Similar to results obtained with a seed-node size of 20, Zero-Discount algorithm has the fastest average propagation speed among the four algorithms. In addition, it is about 20% faster than Degree-Centered algorithm, even better than the result with the seed node size of 20. This result indicates that the drawback of Degree-Centered algorithm is more prominent with increasing of seed-node set. The Zero-Discount algorithm has more advantage over Degree-Centered algorithm in larger seed-node selection.

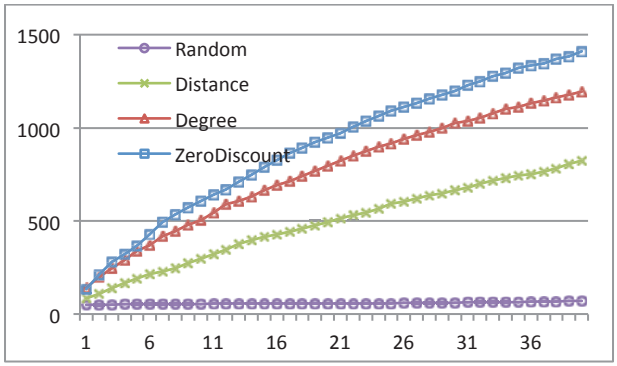


Figure 7: Average Speed in CE-IC Model (hep k=50)

The instant propagation speed is shown in Figure 8. It is more obvious that even Zero-Discount loses more speed at the first step, however, Zero-Discount can almost beats Degree-Centered in the following steps.

To test if the faster propagation speed of Zero-Discount algorithm is also true for other data sets, similar experiments were run in a larger data set, the phy data set. The average and instant propagation speeds are shown in Figure 9 and Figure 10, respectively. Consistently, Zero-Discount algorithm has the fastest average and instant propagation speeds among all four algorithms. Distance-centered algorithm performed much worse in this experiment. One reason of this low performance could be that the selected seed nodes have a lower propagation probability.

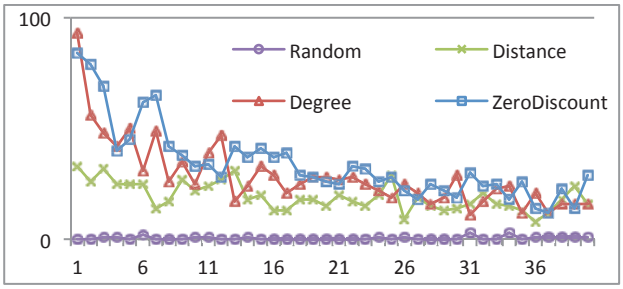


Figure 8: Instant Speed in CE-IC Model (hep k=50)

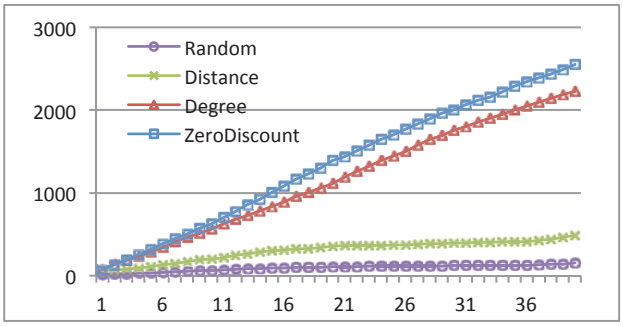


Figure 9: Average Speed in CE-IC Model (phy k=20)

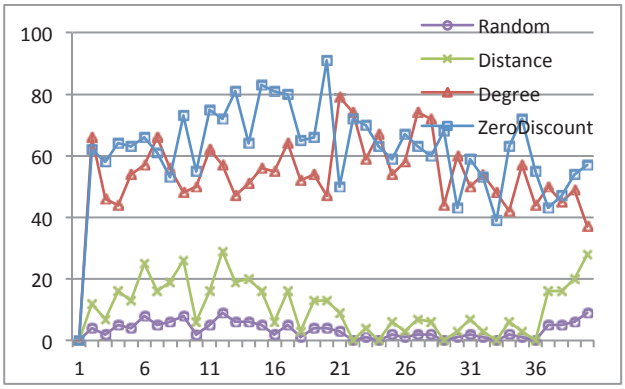


Figure 10: Instant speed In CE-IC model (phy k=20)

4.2 Results in CDE-IC Model

In the above experiments, Zero-Discount algorithm always has the fastest propagation speed among the four heuristic algorithms in the CE-IC model. Next step, I will add dynamic property into model to test if Zero-Discount is still the best in CDE-IC model when propagation probability changes with time. According to formula 3, the adoption probability of a node will decrease with time in exponential number. I set the half time period to 20, which is half of experiment units. I will also record at which step each active

node tries to propagate influence at initial time for relative edges. I will test CDE-IC model with 50 seed nodes set.

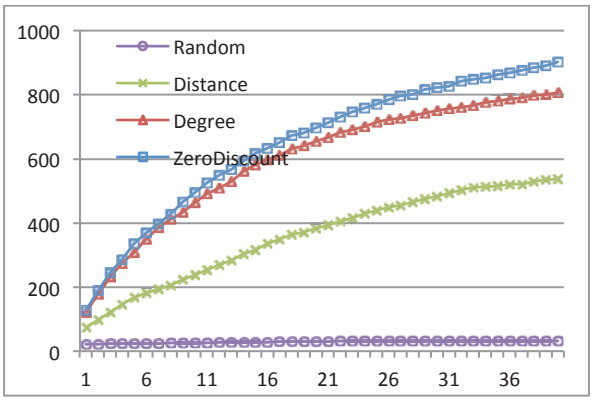


Figure 11: Average Speed in CDE-IC Model (hep k=50)

In Figure 11, we can infer similar conclusion as in CE-IC model. We can also see clearly the effect of exponential decreasing in adoption probability. First, the total number of affected nodes in CDE-IC model is about 1/3 less than CE-IC model. Second, the curves in CE-IC model is more like straight line, which means the newly affected nodes is increasing at steady pace. Contrastingly, we can see the decrease in trend of the curve in CDE-IC model. Both models show that my novel Zero-Discount can always beat all the other heuristic methods with best quality of seed nodes set. Finally, we can add a small correction factor to probability if considering Social Trend. However, this value is not significant until a large part of the social networks is under influence. Since we are studying speed of influence propagation, which is more meaningful in a short period, when not so many nodes under influence, we can safely ignore it.

5. CONCLUSION

In this paper, I studied how to maximize speed of influence propagation in social networks. I proposed a new Continuous Dynamic Extended IC (CDE-IC) Model, which is an improved modification of the Extensional Independent Cascade (EIC) Model. The original EIC model has two drawbacks: first, an active node can only try to propagate its influence to its neighbors once; second, the propagation probability does not change between nodes. Both of these problems are solved in my CDE-IC model. I ran four algorithms in CDE-IC Model, Random, Distance-Centered, Degree-Centered and my novel Zero-Discount method. Experiments on two data sets with different sizes of seed nodes all showed that my Zero-Discount method performed better than any other heuristic methods. The previous best method, Degree-Centered, was 10%-20% slower than my method.

6. REFERENCE

[1] Domingos P, Richardson M: Mining the network value of customers. In: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. San Francisco, California: ACM; 2001: 57-66.

[2] Richardson M, Domingos P: Mining knowledge-sharing sites for viral marketing. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. Edmonton, Alberta, Canada: ACM; 2002: 61-70.

[3] Kempe D, Kleinberg J, #201, Tardos v: Maximizing the spread of influence through a social network. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. Washington, D.C.: ACM; 2003: 137-146.

[4] Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N: Cost-effective outbreak detection in networks. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. San Jose, California, USA: ACM; 2007: 420-429.

[5] Goyal A, Lu W, Lakshmanan LVS: CELF++: optimizing the greedy algorithm for influence maximization in social networks. In: Proceedings of the 20th international conference companion on World wide web. Hyderabad, India: ACM; 2011: 47-48.

[6] Chen W, Wang Y, Yang S: Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. Paris, France: ACM; 2009: 199-208.

[7] Granovetter M: Threshold Models of Collective Behavior. The American Journal of Sociology 1978, 83(6):23.

[8] Schelling TC: Miromotives and Macrobehavior. 1978.

[9] Jacob Goldenberg BL, Eitan Muller: Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth. Marketing Letters 2001, 12(3):12.

[10] Wang Z, Qian Z, Lu S: A probability based algorithm for influence maximization in social networks. In: Proceedings of the 5th Asia-Pacific Symposium on Internetware. Changsha, China: ACM; 2013: 1-7.

[11] Gerard Cornuejols aLF, George L Nemhauser: Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms. Management Science 1977, 23(8):21.

[12] Goyal A, Bonchi F, Lakshmanan LVS: Learning influence probabilities in social networks. In: Proceedings of the third ACM international conference on Web search and data mining. New York, New York, USA: ACM; 2010: 241-250.

Perspectives of Modeling in Metallurgical Production

K. Aksyonov[1], **E. Bykov**[1], **E. Sysoletin**[1], **O. Aksyonova**[1], **A. Nevolina**[1], **and N. Goncharova**[1]
[1]Department of Information Technology, Ural Federal University, Ekaterinburg, Russian Federation

Abstract - *The paper introduces the architecture of the automated system for metallurgical production. We focus on the interaction of the sub-systems, dive into the principles that the simulation module is based on, analyze the alternatives and look at the perspectives of modeling for the problem domain.*

Keywords: Simulation application; data exchange; message exchange; agent-based simulation

1 Introduction

Effectiveness of metallurgical production is tightly interconnected with perfection of technological, as well as logistical and business processes. The problems that can benefit from modeling are listed in [1].

1. Industrial
    a. Irregularity of production load over a monthly period and other scheduling intervals,
    b. Mistrimmed production on interconnected technological systems in the shops and between the ones,
    c. Suboptimal load of production units of main technological process,
    d. As a result of the previous, suboptimal load of auxiliary and transportation equipment,
    e. Unavailability of operative modification of the schedules and production graphs on technological reasons, related to the downtime of equipment on various reasons, interruption of stock supply,
    f. Unavailability of operative modification of the schedules and production graphs due to business reasons, including rejection, emergence of new orders, modification of specifications, modification of volumes, etc.
2. Predictive modeling, planning of development, including upgrades and deployment of information technology and automation systems:
    a. Analysis of effectiveness of suggested actions on new technological decisions,
    b. Analysis of effectiveness of the innovative events on energy and resource saving,
    c. Analysis of suggested actions on new business decisions at the level of processing facilities control, shop and inter-shop control and interaction,
    d. Analysis of results of possible operative re-planning in case of unexpected failures, accidents and other emergencies.

Thus, interaction of technological, logistical and business models is an important aspect for improvement of effectiveness of a metallurgical enterprise as a whole.

2 Model Interaction

The most effective way of interaction of model integration system and automated information system of an enterprise consists in automated obtaining the data required for modeling directly from the automated information system (Figure 1). In order to implement this method, we suggest using the Messages queue system, which itself constitutes the architecture and intermediate level software, which collects, stores and distributes messages between subscribers.

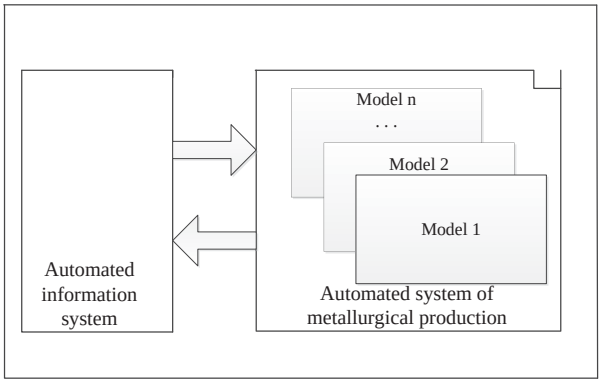


Figure 1. Interaction of integration module with the corporate information system

Criteria	Redis	RabbitMQ	ActiveMQ	Socket.IO
Performance	high	high	high	high
Scalability	high	high	high	high
Clustering	no	yes	yes	no
Java support	yes	yes	yes	yes
Ease of use	high	average	average	high

Table 1. Comparison of queuing brokers

Existing message brokers have been analyzed during research. All of them provided guaranteed message delivery between applications. Analysis results are presented in Table 1.

Since implementation, based on REDIS [2] and Socket.IO message exchange [3], is simpler, they were selected for data exchange between the automated information system and automated system of metallurgical production.

3 Interaction of Simulation Module with Data Exchange Module

The first option of interaction between the simulation module and the module for data exchange with the enterprise automated system suggested the use of "publisher – subscriber mechanism". In this case the simulation module subscribed for a specific subject, and the data exchange module published all occurring events, distinguishing these by the subject. Thus, the simulation module obtained only the required events.

A significant disadvantage of this option is that the simulation module has to reserve its own resources for the channel listening and awaiting the new information (see Figure 2).

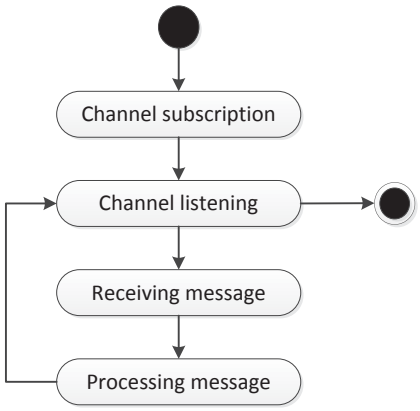


Figure 2. State chart for modeling the behavior of simulation modeling module while using MQ mechanism

The main ideology of the system under discussion is the asynchrony of operation and the use of non-blocking operations of input-output, which means that the awaits mechanism is appropriate. Correspondingly, the interaction mechanism between simulation and data exchange modules has been replaced during technical engineering in favor of asynchrony and elimination of waits. At this moment simulation module uses the events mechanism, which is identical for all parts of the system. The event exchange is based on the Socket.IO protocol, on top of which the Java framework netty.io is implemented. Netty.io is the event oriented library for the implementation of network exchange using various protocols, including UDP, HTTP/HTTPS, and Socket.IO. Data exchange module receives the emerging events on any of the listed protocols, providing the maximum versatility of its operation. At the same time all events are produced using the only protocol, which is Socket.IO. Such approach allows maximum unification of event exchange, providing the feature of receiving messages both from Java based modules and from in-browser clients, written in JavaScript. Work through Socket.IO requires registration of callback procedures. Any waits in this case are eliminated.

Use of Socket.IO provides the following features [3]:

- Sending and receiving events,
- Sending messages to connected users, using *send()* and *emit()* functions, callback methods are available for acknowledgement of received messages from the client,
- Broadcasting the messages using the option *Broadcasting = true*,
- Grouping users and sending messages to a group,
- Use of events, related to user sign-in and sign-out.

Apart from this, use of event mechanism conforms to the Model-View-Controller concept, since the processes of receiving, possible data pre-processing, model execution and display of results are fully isolated. This allows multiple views of the same working model at the same time, e.g. the work may be represented in 2D and 3D, or the specific user may have the aggregated information on the shop or the full detail on a certain mechanism.

4 Loading the Model to the Simulation Module

Before the interaction with data exchange module, the simulation module needs to register by providing its unique identifier and name to the data exchange module. From the point of view of a system as a whole, the simulation module is

the same participant as all other modules and workstations. Correspondingly, the simulation module entity with the specified name needs to have the required permissions set and the required roles assigned beforehand. These settings are made by the system administrator using the visual user GUI. Since the permissions are validated before each event, the operation of simulation module would be impossible without preliminary permissions configuration. Even the attempt of simulation module registration in the system would fail, apart from any other step.

Multiple concurrent simulation modules are allowed in the system. The physical location of a simulation module does not matter. No model is executed within the simulation module immediately after start. Nevertheless, the module notifies all participants of the system about the fact of its start and its server CPU load ratio. Data exchange module logs the start of simulation module. Since at this stage of development the system does not allow delayed model start, no other action follows. The simulation module is ready and waits for commands from the users and system administrators.

Model start within the simulation module is initiated by the user. This is done by broadcasting the event from user's workstation via the data exchange module. This event is interpreted by all operation simulation modules as a command to provide information on their state at the current moment of time. Each simulation module generates its own message that contains the module name, identifiers of currently executed models, server CPU load ratio, etc. Thus, users receive information about available simulation modules.

After selecting the specific simulation module and launching the model, an *Event_Model_Start* is generated for the selected simulation module. Model launch is divided into two stages. First, the simulation module attempts to load the model from the database. In case of any errors, they are returned to the user. If the model is loaded correctly, the user receives the original *Event_Model_Start* event, which indicates that the model launch goes to the second stage. Simulation module actions after loading the model depend on the features of the loaded model and the mode of its operation. Model operation results are displayed on visual forms on the workstations. Note that model representations may vary between the users.

5 Receiving Data from the Data Exchange Module

If model operation permits the real-time mode, the simulation module subscribes for the necessary parameters. Subscription checks previously loaded models, which avoids double subscription for the same parameter. After receiving the new parameter value from the data exchange module, the simulation module transfers it into the model (Figure 3).

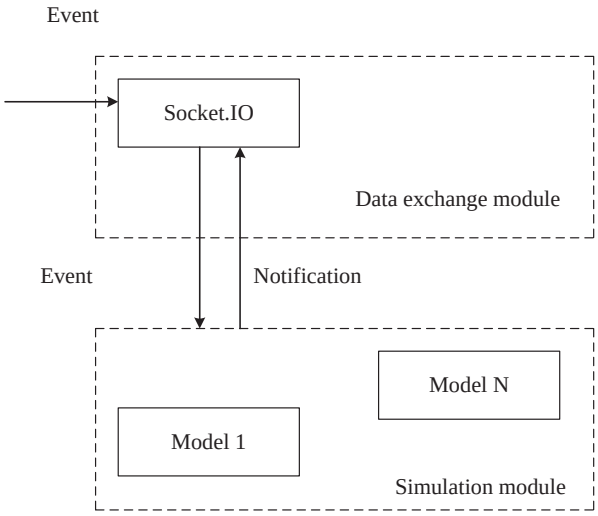


Figure 3. Receiving data from the data exchange module

6 Transferring Data from the Simulation Module to the Data Exchange Module

Simulation module launches the models and transfers the results of their operation into the data exchange module. On the first step the fully loaded and executed test model modifies the state of its inputs. Next the parameter is searched in the database. When the data exchange module receives the *Event_ParamValue*_Got event, it runs a number of checks. The received value must be of expected type (string, integer, etc.). The values themselves are written in the pre-defined form, depending on parameter kind (vector, scalar, linked list, synthetic). Thus, before the «parameter value changed» event is received, the parameter itself needs to be stored in the database as an object with certain features. If this is not done, the processing of received value will cause an error.

7 Interaction of Simulation Module with the Database

Simulation module interacts with the database in two cases:

1. When loading the model, since all information is stored in the database,

2. When model operation requires processing of statistical data, since this data is obtained beforehand and stored in the database.

In the first case the simulation module receives a "Start" signal, which has the unique model identifier. Knowing it, the simulation module accesses the database and selects the corresponding model. In the second case the simulation module receives the required parameter identifiers from the model as well as the desired time interval. Next it selects the

values of parameters from the database and forwards them to the model.

8 Representation of the Simulation Model as an Agent

The comparison of properties of cognitive and reactive agents, presented in [4], has been mapped to the features of simulation module. The results are presented in Table 2.

9 System Structure Definition

Simulation module may logically be divided into the following parts:

- *Talker* – communication with the data exchange module via the network,
- *Main process* – processing broadcast messages and events, events, directed to a specified simulation module, provides general information about the model, generates simulation time-steps and informs all executing models about these, monitors changing model statuses.

Messages, received from the data exchange module, are processed in two stages. Stage one is processed by the Talker. It processes physical errors in the connection channel and acknowledges message delivery.

For each message, sent by the simulation module, an acknowledgement is required. Each message has its own unique identifier and the identifier of the sending simulation module. The Talker controls acknowledgment process, i.e. it monitors notifications of message delivery. If the acknowledgement is not received within a specified interval, the Talker repeats the message to the data exchange module. There is a total of three attempts.

The model definition class provides the following information on the model, loaded from the database: general model status, model state, model operation requirements, modeling time. Model can have one of the following statuses: undefined, loaded to memory, unloaded from memory, executing, stopped, suspended.

An array of all loaded models is stored within the main process.

System structure definition is presented on Figure 4.

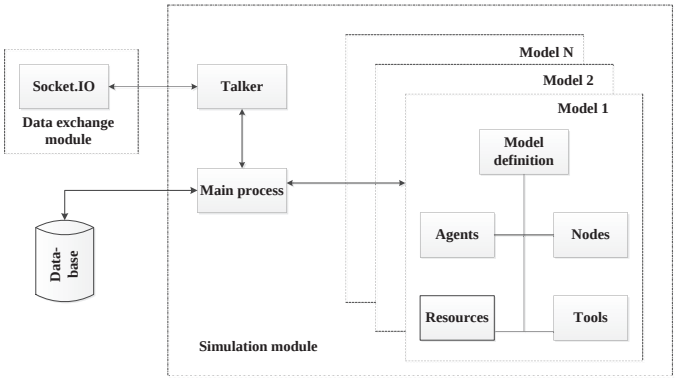


Figure 4. System structure

10 Conclusions

We have introduced the perspectives od modeling in metallurgical production. We compared the functionality of simulation module with various agent architectures. We may conclude that the simulation module is essentially an

Feature	Cognitive agents	Reactive agents	Simulation module
Internal model of the environment	Advanced	Primitive	Has the definition of model structire, ways of representation of modeling results for the users
Discussions	Complex and reflexive discussions	Simple one-step discussions	Can identify, which parameters are required for the specified model, can validate model consistency
Motivation	Advanced motivation system, including beliefs, desires, intentions	Simple motives, based on survival	Behavior is described with one-step conclusions
Memory	●	○	●
Reaction	Slow	Fast	Fast
Adaptiveness	Low	High	High
Module architecture	●	○	●

Table 2. Comparison of Agent Properties

intelligent agent that builds the infrastructure for the execution of logistical, technological and business models.

11 Acknowledgment

Research is conducted under the terms of contract № 02.G25.31.0055 (project 2012-218-03-167) with financial support from the Ministry of Education and Science of the Russian Federation.

12 References

[1] Vlasov S. A., A. L. Genkin, I. V. Nikulina, "Integrated control of automated systems of metallurgical production based on predictive simulation modeling, Proceedings of the 4th Russian scientific conference on simulation modeling and its application in the science and the industry «Simulation modeling. Theory and practice», IMMOD-2009, p. 48.

[2] http://redis.io/

[3] http://socket.io/

[4] Tarasov V. B., Agents, multi-agent systems, virtual communities: strategic vision in informatics and artificial intelligence, AI News, vol. 2, 1998, pp. 5-63]

Development of a Tool for Semi-automatic Creation of Virtual Environments

Nurkkala, V-M. & Kalermo, J.
Kajaani University of Applied Sciences, Kajaani, Finland

Abstract – *Due to digitalization the use of virtual environments in different products and services has increased in many fields. For decades military and aviation simulators have shown way of utilizing VE's for training. The increasing need of virtual environments that are based on real environments has been recognized in several areas. For example healthcare, sports and forest industry have woken up to the possibilities to utilize VE's in their businesses. However, the markets are lacking a solution to generate high quality virtual environments cost-effectively based on different kind of databases from real environments. In this paper we present the software that utilizes data from different sources to generate virtual environments based on real environment cost-efficiently.*

Keywords: Virtual environment, digitalization, terrain data, geological data, GPS.

1 Introduction

Brooks [1] defines a virtual reality experience "as any in which the user is effectively immersed in a responsive virtual world". It is hard to find a difference between terms virtual reality and virtual environment. Virtual environment could be said to be more "concrete" and virtual reality more "abstract". In this article we define the term virtual environment (VE) as real-looking virtual landscape, which is based on real environment data or on imaginary environment.

For decades military as well as driving and working machine simulator manufacturers have used virtual reality in their training. The expectations of virtual reality have always been enormous, but so far the technology and applications have not made possible the final breakthrough. Digitalization, growth of entertainment business, technology development and more sophisticated tools to create low-cost, but high quality virtual environments may finally open the doors to apply VE's to new areas in an enormous scale. Oculus Rift and other virtual headsets will have a remarkable role in this revolution. Augmented reality is following quickly due to technological developments and may bring even a bigger revolution than VE's in the future. Still, it is good to remember the difference of these technologies and that they are not direct competitors with each other. They have their own strengths and will be used in a most of the cases for different purposes.

An increasing need to create high quality virtual environments based on real land areas has been recognized in different businesses. Nevertheless, virtualizing a real land area is time consuming and rather expensive. There are usually many kinds of geological and terrain data available which could be utilized in the creation of virtual environment. For the need to decrease the development costs and time we started to develop ScenicoTool software which will semi-automate the creation of virtual environment. We succeeded to develop software that uses different kind of geological and other data to generate high quality virtual environments in very short time.

2 The use of virtual environments

In the last couple of years the use of virtual environments (VE's) has expanded to many new fields. Typical applications have been flying and driving training simulators, entertainment (especially games), architecture, education, medicine, healthcare, sports and arts. Usually the purpose of utilizing VE's to these areas has been to improve safety, skills and/or efficiency and to save time and/or money. Product developers and service providers want to offer close to real-life experiences by using VE's. For educational purposes VE's are used especially to achieve more efficient learning results, and entertainment business aims at providing people enjoyment and hooking moments e.g., by playing games. There is lot of scientific evidence of the use of VE's to e.g., achieve good learning results [2] and increase the physical activity level [3].

Even though virtual environments have been utilized for several fields, it has most likely been just a scratch of the possibilities VE's provide. It is likely that new innovations that utilize VE's will be announced in rapidly growing phase in the forthcoming years. Market analysis of the use of virtual environments and augmented reality forecast remarkable growth. The annual growth of these markets is expected to be about 15% in 2013-2018 and reach $1.06 Billion in 2018 [4].

Recently the use of VE's has increased drastically by game industry due to the enormous growth of the industry. Games such as SimCity and Cities: Skylines are good examples of games that are built to simulate high quality virtual cities and worlds. Construction industry, architecture and environment planning are utilizing VE's in their planning projects. VE's have been utilized to present prototypes of the buildings and rooms, but can be utilized also for an iterative, user-informed process throughout the entire design and development cycle [5]. Forest and mining industry have been

utilizing VE's in their simulators (e.g., companies such as Ponsse, Tenstar and MeVEA). Also driving simulator manufacturers utilize high quality VE's in their simulators (e.g., Oktal, Realtime Technologies, Eca Faros). Exercising and athlete training have been provided with new innovations [6] in which VE's provide new tools and motivating content into training [3]. Tourism services are taking advantage of 3D visualizations in their operations and services (e.g., Vatican offers 3D tour in Sistine Chapel). These and many other fields benefit from VE's that are representing the real environments in high and accurate enough quality.

Due to improvements in technological capabilities of hardware and software, such as continuously growing processing power, image resolution and more powerful graphic cards as well as communication bandwidth, the quality of virtual environments have improved quickly. Also in gaming industry the game engines such as Unity3D and Unreal Engine provide good tools for game developers to create high quality virtual environments. Moreover, a great number of graphic designers provide good quality 3D models, texture packages and virtual environments for reasonable costs.

Despite the above-mentioned improvements in technologies, the development of VE's based on real land areas is still very challenging. The development requires most often lot of handwork and is very time consuming. There are software that enable creation of virtual environments (e.g., Virtual Terrain Project and WorldComposer) by utilizing different databases (e.g., Open Street Map, point clouds, road data databases). Still, existing software are utilizing only a part of the available data, which could be used to add more details to VE. Also support to import created VE's to other software, such as simulator software is missing.

3 Landscape data

In the transit of digitalization, huge data masses concerning e.g., geological and forest data are stored in cloud services and other storages. Vast information of landscape data from all over the Globe is available for organizations to build up new services. Some of the data are freely available for research and educational but also for commercial use. On the other hand, some of the data may be rather expensive.

4 The background of the project in Kajaani University of Applied Sciences

During 2010-2012, along with the research and development of driving simulators we learned, for example, how to create high quality virtual environments, how to build and use multiscreen CAVE environment and how to control and fine tune different motion platforms [7]. During the 2013-2014 we have developed a new kind of exergaming simulator for gym training, fitness testing and rehabilitation. The target was to develop a simulator, which integrates different exercising, and rehabilitation devices, immersive virtual environment, games, and advanced motion controllers in order to bring these devices to a new level [6]. During these projects the idea of semi-automatization of virtual environments in a cost-effective way for generating new VE's was recognized. The development of VE's has required rather much handwork and it used to be time consuming especially when the virtual environment has been made to be one-to-one copy of the real environment. Much of the development of virtual environments was done manually by graphic designers with help of numerous pictures of the environment and Google Street View.

In discussions with organizations in the municipality of Kainuu and in further Finland, it was recognized that numerous organizations saw benefits of utilizing virtual environments in their services, especially if the VE's were based on the real environments. However, these (VE's) should be achieved easily and with reasonable costs and should be able to import different software. As no such software that would enhance the VE creation process were recognized, a software development project was initiated at Kajaani University of Applied Sciences. The objective of the project was to develop a tool for semi-automatic creation of virtual environments. The competent software developers at the University were recruited to work for semi-automating the creation of VE's with software that use different databases to create the VE based on real terrain and other data.

5 Development team

The R&D team consisted of team lead and software team with three talented developers and two graphic designers. Unity3D game engine was chosen for the platform of development, as it was already familiar to the team and its features suit well also to other operations than game development. It offers lots of ready 3d objects and features for instance for modeling different weather conditions and seasons.

The project was done in collaboration with several organizations that represented different fields, such as sports, forest industry and rehabilitation. In the beginning of the project, reviews of available data and formats as well as possible costs of the data were explored. It was also important to discuss with forest industry experts to understand what kind of data are available for example from forest databases.

6 ScenicoTool software

The developed ScenicoTool software is able to generate quickly virtual forest environments that are based on real Finnish environments. Currently the data provided by National Land Survey of Finland (NLS; freely available data) and by Finnish Forest Centre is used. The terrain is based on the Digital Elevation Models that the NLS provides on the accuracy of two meters. The undergrowth, big rocks, paths and tracks, roads etc. are also captured from the data provided by NLS. The different types of trees are put on their places on 16×16 meter raster's based on the data from Finnish Forest Centre or respectively, from Metsähallitus, which is a state

enterprise that administers majority of state-owned land and water areas in Finland.

After the first phase of software development was ready, the development of GPS data feature was started. The aim was 1) to use GPS data to generate virtual environments in which the recorded route matches the real world equivalent but all the surrounding nature is based on imaginary environment and 2) to add GPS routes to the VE that is created based on the databases of real environment. The objective for utilizing GPS data was to open new possibilities to create real world routes even there is no other data available. This improvement will open up new possibilities specially to use VE's e.g., in various simulators and sports.

7 Pilot cases for testing the tool

During the R&D project, ScenicoTool software was tested in several cases, targeted to sectors of forest, sports, and rehabilitation. The pilot cases are presented next.

FOREST. Finnish Forest Center played an important role in the project. Their expertise on forest and GIS data helped to understand how to effectively utilize the forest data. Other partners in this work package represented forest machine simulator manufacturer and forest machine manufacturer, who utilize virtual environments in the training simulators. The game engine enabled exporting the virtual environment that was created with Unity3D to match the requirements of the simulator software.

Specific land area from Kajaani, Finland (Pöllyvaara recreation area), was chosen for pilot testing. After generating the environment, it was tested in Athene Exergaming simulator (Fig.1).

Figure 1. Walking in virtual forest environment.

SPORT. The ScenicoTool was used to create virtual Kontiolahti Stadium and skiing tracks, as well as the terrain and forest based on the real Kontiolahti area. The graphic designers modeled the 3d objects such as buildings and other assets. The tracks of Biathlon 2015 World Championships were created in the virtual environment and tested at Vuokatti Testing Center in December 2014. The virtual Kontiolahti was attached with a large treadmill and the athletes were able to walk, run and roller ski through the competition tracks.

Figure 2. Virtual Kontiolahti was developed for Athletes training.

REHABILITATION. The partners in this pilot case represented rehabilitation. These partners saw value in being able to provide virtual environments as part of e.g., physiotherapy with a recumbent bike or a treadmill so that the rehabilitees would exercise in places that are familiar to them. In the project, THERA-Trainer Tigo restorator bike was used with virtual environments in the rehabilitation exercises (Fig.3). Created VE's were tested with a rehabilitation pilot group with good feedback of the increased motivation and enjoyment for exercising.

Figure 3. Virtual environments were tested for rehabilitation use.

8 Conclusions

The software will be a valuable tool for many purposes. Sports and health care will benefit of getting access to virtual environments that are based on real areas, which is very valuable for instance in preparations and coaching for forthcoming championships, exercise testing, training and rehabilitation. The forest owners and forest machine operators can be provided with a visualization of how the forest property would look like e.g., in the case of logging and also

move in the virtual forest. The virtual environments are useful also for, e.g., forest machine training simulators. For tourism, applications to show the nature could be rather quickly be initiated.

ScenicoTool software development has taken the first steps. As the software is created as a plug-in to Unity3D engine, it is very straightforward to export the created VE's into various software and products. The next steps will be looking at the data from different countries in order to enhance the software to develop environments from all over the globe, and later on commercialize the software.

Acknowledgements

The project was funded by Finnish Ministry of Education and Culture and by funding provided by the co-operative companies.

References

[1] Brooks Jr, F. P. (1999). What's real about virtual reality? Computer Graphics and Applications, IEEE, 19(6), 16–27.

[2] Tendick, F., Downes, M., Goktekin, T., Cavusoglu, M. C., Feygin, D., Wu, X., Eyal, R., Hegarty, M. & Way, L. W. (2000). A virtual environment testbed for training laparoscopic surgical skills. Presence: Teleoperators and Virtual Environments, 9(3), 236–255.

[3] Monedero, J., Lyons, E. J., & O'Gorman, D. J. (2015). Interactive Video Game Cycling Leads to Higher Energy Expenditure and Is More Enjoyable than Conventional Exercise in Adults. PloS one, 10(3), e0118470.

[4] MarketsAndMarkets (2014). Augmented Reality & Virtual Reality Market by Technology Types, Sensors (Accelerometer, Gyroscope, Haptics), Components (Camera, Controller, Gloves, HMD), Applications (Automotive, Education, Medical, Gaming, Military) & by Geography - Global Forecast and Analysis to 2013–2018.

[5] Drettakis, G., Roussou, M., Reche, A., & Tsingos, N. (2007). Design and evaluation of a real-world virtual environment for architecture and urban planning. Presence: Teleoperators and Virtual Environments, 16(3), 318–332.

[6] Nurkkala, V. M., Kalermo, J., & Järvilehto, T. (2014). Development of Exergaming Simulator for Gym Training, Exercise Testing and Rehabilitation. Journal of Communication and Computer, 11, 403–411.

[7] Koskela, K., Nurkkala, V. M., Kalermo, J., & Järvilehto, T. (2011). Low-cost Driving Simulator for Driver Behavior Research. Proceedings of the 2011 International Conference on Computer Graphics and Virtual Reality, CGVR (Vol. 11, 18–21).

Other sources

Simcity: www.simcity.com/en_US/buy/ep1
Sistine Chapel in 3D: http://vatican.com/tour
Virtual Terrain Project: http://vterrain.org/
WorldComposer: www.terraincomposer.com/worldcomposer/

Usability Study of Leap Motion Controller

Alberto Scicali[1] and Hans-Peter Bischof[1 2]
[1]Computer Science Department, Rochester Institute for Technology, 102 Lomb Memorial Dr., Rochester, NY 14623, USA
[2]Center for Computational Relativity and Gravitation, Rochester Institute for Technology, 102 Lomb Memorial Dr., Rochester, NY 14623, USA
{afs2842, hpb}@cs.rit.edu

Keywords: Usability Study, 3-Dimensional Input Devices, Unity

Abstract: The qwerty keyboard and a mouse are still the main input devices used today. The last upgraded mouse was introduced by Apple in 2009; the Magic Mouse. The Magic Mouse is a multi touch mouse and can be used to recognize simple gestures. Computer games and visualization environments are typically played out in a 3D world projected on a 2D screen, or a 3D environment. A 2D mouse is not very useful in a 3D environment. On July 2013 Leap Motion released a device which can be used as a 3 dimensional mouse. This paper presents the usability experiments and first results.

1 INTRODUCTION

A long time go a keyboard was the only mechanism to communicate with computer programs. Operations we have no problem to perform today, such as Cut and Paste operations, took a while learn if you have been using an editor like vi(William Joy, 1977). Programming tools, like the vi, was written for developers by developers and a perfect tool for creating code, but a terrible tool for any non-developer.

The introduction of graphical computer interfaces changed the way in which we interact with computers completely, especially after the mouse came along. The usage of a computer became so much easier for most users after the release of Microsofts mouse compatible version of Word. The next milestone came in 1984 when Apple released Macintosh 128 with a update version of the Lisa mouse (Pang, 2002).

The world of interaction with a computer has not changed for the next 25 years or so, because there was apparently no need for it. The qwerty keyboard layout, introduced to the market 1874, is the most common keyboard up till today, despite the fact the reason for this layout, jamming keys, does not exist anymore. Tetris could have been played with a keyboard, but games in a 3D world required something more sophisticated, such as console game controllers. However, moving freely in 3D space was a challenge, even when using a game controller.

The Kinect(Zhang, 2012) was another game changer. The Kinect allowed interaction with a game without a game controller, or to better describe it, the human body and voice are used as the game controller. The Kinect uses a motion sensing device and voice recognition system to interact with games.

The next device, which became available using sensing technology was the LeapMotion Device (Weichert et al., 2013). The LeapMotion is an inexpensive optical sensor capable of recognizing position of finger tips, palm vectors up to 0.2mm accuracy. This device allows to study gesture recognition algorithms in a detailed way.

This paper discusses scenarios in which these motion sensing devices outperform a keyboard, or a mouse, and also discusses their limitations.

2 PROBLEM DESCRIPTION

The inputs coming from a mouse or a keyboard is discrete and do not need any or very limited interpretations. A mouse down event is a discrete event at a given position on the screen and carries depending on the environment information about the position of mouse pointer, time of the click, etc. A double, or a triple click is in event over time, and will only count as double/triple click if the two events happen in a defined time window. A mouse can move on a limited 2D space and the basic functionality has not changed for a long time. Up until Apples Magic Mouse(Loyola, 2010) opened the window to 2D gestures a bit; swipe between pages or full screen appli-

cations and a double tap which gives access to mission control.

A mouse is a appropriate input device, if a correlation between between the action and reaction can be created. Tetris(Burgiel, 1997), the 2D tile matching puzzle game, can easily played with a keyboard. Rotating of a tile, move to the left, right, and down are the only needed operations to play this game, and these operation can easily be mapped to keys. A keyboard or a mouse is a less then optimal input device, if you want to play table tennis in a 3D world, because many parameters must be controlled at the same time. The position of the racket, orientation, speed and direction must be controlled at the same time. It is simply to difficult to map these analog signal to discrete events like a click, and movement of a mouse on a 2D surface. On the other side, a hand can be easily moved in 3D space to control a racket in the desired way.

The Leap Motion(Weichert et al., 2013) device is motion sensing device for hands. It senses objects near by like the hands.

This paper discusses scenarios in which a motion sensing devices outperform a keyboard, or a mouse, and also discusses their limitations.

3 LEAP MOTION DEVICE

The LeapMotion device is a very small and inexpensive usb device. It tracks the position of objects in a space roughly the size of the top half of beach ball. The precision of what the sensors is up to 0.01 mm. This means the sensor can identify 700,000,000 points in its view area, which is extremely precise.

The Leap Motion device uses three infrared camera to detect the reflection of light from users hands and arms. It also senses in on direction, therefore it can not see objects which are behind each other. The SDK V2 introduced a detailed skeleton model which allows a better prediction of the position of objects, like hidden fingers. The API supports access to the stream of frames or can be used to detect predefined gestures. We use both for levels for the implementation of the software.

4 EXPERIMENTS

4.1 Visualization

Visualization of data comes is the art of creating meaningful images, video, or audio out of a n-dimensional data set(Bischof and Dong, 2011). The process includes typical a bit of experimenting with the attributes of the lights like positions, color, and directions etc. The position, direction, and up vector of the viewing point camera - and other parameters require typically some experimenting in order to create a meaning full visualization.

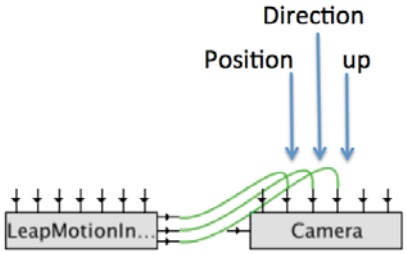


Figure 1: Leap Motion used as an Input Device

We used the Leap Motion device just to control the camera. Most visualization systems are using a 3D space, which means that camera, lights must be moved along the x, y, z axis. Defining the values of the coordinates with sliders, or defining the values in a text field typically does this. This is a cumbersome and not a very explorative undertaking. With the Leap Motion Device we can take the virtual camera and position it space and focus the camera at a particular position. We implemented this and used the left hands palm position and direction to arrange the camera in space. The up position was determined with the palm direction of the right hand. The Spiegel programming framework is easily extensible and we added just one component to include the Leap Motion device, see Figure 1. The LeapMotion component sends out new values along isa output channels if any of the value changes and the camera component reads the new value and applies the value to is state.

4.1.1 Observations

It was relatively easy to control the camera with hand, it got a bit more difficult to use two hands for the complete control of the camera. After a while it felt like hard work to keep both hands up and moving then them around.

The Leap Motion device sends information as soon as it detects anything, which at the beginning is more or less noise. We used a Finite State Machine(Brand and Zafiropulo, 1983) to control when the device starts and ends sending information.

A camera might be used inside a cube of length 1, or in a cube of much greater length, like 1000. We employed also a FSM to determinate a scalar value which mapped to our rather limited hand motion range.

The use of system was no far away from easy

to use. To many things needed to be remembered and most people using the system have a very difficult time to move their hands independent form each other.

The Leap Motion device sends out up to 200 frames per second. The processing of 200 frames per second would be of little value, because the changes between each frame would be extremely minimal, because you can not move your hand in much1/200 of a second. In order to reduce the workload on the system we process only frames which are at least n seconds apart and n is variable. Experiments have shown that 1/10 second is a reasonable value.

4.2 Unity3D Games

The games created in the Unity3D game engine were intended to further exemplify the advantages of control that is offered with the Leap Motion device, especially when compared to a computer mouse. Each game was created with specific tasks to be completed that demonstrates the need for input devices that allows software to be controlled with 3-D controllers.

4.2.1 Interactive Globes

Two interactive globe applications were created in order to test varying methods of gesture controls with the Leap Motion device. One globe application, created with Java and jMonkey, is controlled with firm gesture controls that, once recognized, began traversing the globe; i.e. keeping your hands parallel and over 20cm would begin to move the camera away from the globe, effectively zooming out. The second globe application, created with JavaScript and the Google Earth plugin, implemented a complete one-to-one mapping between a users hand movements and how the camera moved around the globe; when a user moves their hand along the XYZ axis the camera will too, i.e decreasing the hand's Y axis will zoom the camera closer to the globe, while increasing the hand's Y axis will move the camera away from the globe. Then, movement in the X axis controls longitudinal panning and Z axis movement pans along the latitude.

4.2.2 Observations

Through experimenting with over one hundred anonymous users during Imagine RIT 2014, it was discovered that the one-to-one globe was much more intuitive for users to understand the controls to, with the instant visual feedback between the movement of their hand and the panning movements of the globe camera, users could easily relate what motions of their

Figure 2: GoogleEarth with LeapMotion integration

hands controlled the camera. While the first globe application, using the firm gestures, was difficult for users to understand the controls. When gesture controls are limited to a firm gesture, with no active feedback, it becomes less intuitive for users. The way in which the Leap Motions controls are utilized is a primary component in allowing the Leap Motion and other 3-Dimensional input devices in becoming more pertinent for users.

4.2.3 Paddle Game

A paddle and ball game was created early on using jMonkeyEngine and Java along side the Leap Motion API to create unique three dimensional experience. The task of the paddle ball game was to continuously bounce a tennis ball into the air with a racket. The virtual racket can be controlled through all axis, X, Y and Z. This allowed experimenters to reach into the 3-Dimensional world; interacting with a game in a way that is not possible with a typical 2-dimensional mouse. This experiment was an early example of the advantages that the Leap Motion device provided over a mouse.

4.2.4 Swatting Game

The first Unity3D game is the simplest, where a ball must be swatted into a goal using a 3-D hand model that is controlled and mapped to the user's hand. This game uses the X, Y and Z axis. A typical computer mouse only allows movement in two dimensions, by creating a game that allows a user to control objects in 3-dimensions. The user is capable of swatting the ball vertically, side to side and towards the far end of the designated arena. Without the 3-Dimensional capabilities of the Leap Motion, a mouse is incapable of swatting the ball into the goal without extra controls being added. The Leap Motion allows full 3-dimensional input controls without only one device, this allows for a more intuitive control and a lower learning curve for the controls.

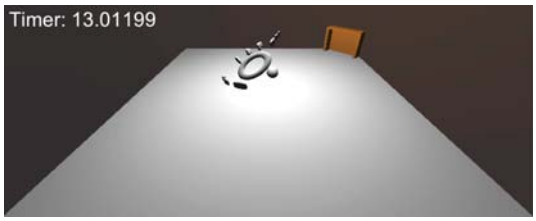


Figure 3: Swatting Unity3D Game

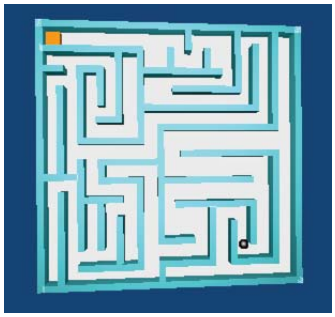


Figure 4: Labyrinth Unity3D Game

4.2.5 Observations

Without adding additional input controls to the game, a mouse isnt capable of effectively traversing the 3-D space. Using the LeapMotion, users were able to swiftly swat the ball into the goal. If the users were new to LeapMotion, they would take a little longer to adjust to the interaction between their hand and the controller. But the end result would always be with the user successfully completing the game.

4.2.6 Labyrinth Game

The Labyrinth game is a more difficult challenge, demanding users to navigate a sphere through a maze. The maze is directly connected to the hand, pitching and rolling as the user's hand does the same. This game tests the intuitive methods to control 3-Dimensional object by the rotation and pitch of a user's hand. As a hand rolls left or right and pitches forward and backwards, the maze does the same. Giving the user full control of the terrain and allowing gravity to pull the ball down the labyrinths corridors as the user tilts the floor. This game allows users to match the exact movements of their hand to the tilt-angle of the maze. A 2-Dimensional mouse would have to map out X and Y axis movement to the angular tilt, rather than directly using the user's hand. The absolute positions of the mouses Y and X axis controls the pitch and roll, respectively.

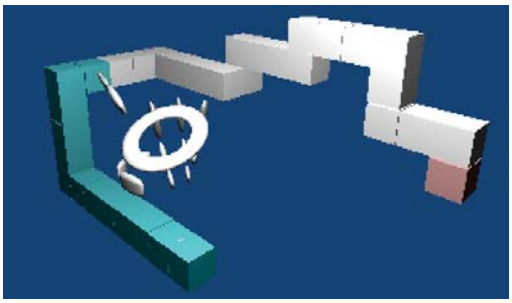


Figure 5: Trace Unity3D Game

4.2.7 Observations

This game was playable with both a mouse and the LeapMotion, however the when users used the LeapMotion the controls were much more intuitive; because of the direct 1-to-1 control of pitching and rolling their hand, the labyrinth mimicked their hand movement as expected. When users played the game with the mouse, it took more time to understand and effectively use the controls.

4.2.8 Trace Game

The final game, Trace, tasks the user to trace a 3-Dimensional path with a 3-Dimensional hand that acts as their own. This task forces the user to use all 3 dimensions to their fullest extent, as well as testing the accuracy of the Leap Motion device. A 2-D mouse is incapable of carrying out this task to completion, the mouse would have to accurately move a 3-Dimensional object along all 3 axis. To allow a 2-D computer mouse to complete the task, extra inputs would have to be added in order to allow the mouse to traverse the 3 axis; adding more inputs other than the mouse introduces more complications by forcing the user to use a combination of inputs, rather than just the mouse, or in the final case the LeapMotion. The Leap Motion device makes 3-Dimensional traversal of the path possible and accurate.

4.2.9 Observations

Users that played Tracer were able to successfully complete the task upon familiarizing themselves with the 3-D environment and the given 3-D Path. With the 1-to-1 mapping between the users hand and the rendered 3-D hand, users were able to smoothly control their hand movement with accuracy and trace the path. One noticeable negative was the inability for users to sense if they truly touched the path objective. When a block on the path is touched, the color changes to teal, however there is no limit on how far the hand/fingers can go through the block objects.

Two solutions to this issue would be to add more feedback, whether through auditory, tactile or more visuals, or to add collision properties between the 3-D hand and the blocks; this way the 3-D rendered hand model wont be able to move through the blocks.

5 RESULTS

With all experiments we have tried to create natural user interface to perform one task. We did not try to create a virtual drum set and ask a drummer if this could be used. we have been interested in the question if the Leap Motion Device could make it easier to perform a task in 3D space.

In all Unity3D game cases, the Leap Motion device was able to more achieve the game task. A computer is only able to be used fully in the labyrinth game, and that is with the necessity of remapping 2 axis to the tilt angle of the maze. The Leap Motion device becomes a more useful and familiar device for 3-Dimensional programs that need to access all 3 axis at a single time with a user's hand/s. There is simply no good way to control the movement of the mouse with the use of discrete events of a mouse or keyboard in an easier way.

The hundreds of users controlling the globe has shown that very simple gestures outperforming more powerful, but more complicated gestures. Most users could not perform to different movements with their hands in order to achieve a desired task.

The Spiegel experiment has shown that the Leap Motion is very useful in order to explore the visualization of a data set. It has also been shown that it is not as simple as find the palm position and direction and using it. FSM, start and end markers need to be used to be somewhat efficient. It is less precise than using exact coordinates, but precise enough to make it very useful.

6 CONCLUSION

Programs that are based in a 3-Dimensional environment are much more accessible with 3-D controllers such as the Leap Motion. A standard 2-Dimensional mouse will not be able to effectively manipulate and traverse the 3-Dimensional landscape without having to add extra inputs. The Leap Motion allows direct 1-to-1 mapping that gives users of varying skill the familiarity of using their own hand. However, these 3-Dimensional devices are only as useful and friendly to user as the user experience allows. If the controls to a program are created with hard-coded gestures that must be fulfilled within a certain degree, the learning curve to effectively using these devices increases. With the ability to access all 3-Dimensions, it is important to allow the controls to be intuitive to use and informative; giving feedback to the user so that they may quickly adapt their hand movements.

7 FUTURE WORK

Our experiments have strongly indicted that a 3 dimensional input device, like the Leap Motion Device, can be successfully used to control objects in a 3 or more dimensional space. It is not clear at this point which kind of gestures work well, or not so well. Future study will bring more clarity.

Additionally the previous work now leads to applying these techniques and uses to a real world, physical applications. The current goal is to map the Leap Motion device to a robotic arm so that it may be controlled with one-to-one hand movements. This could allow users to move a robotic arm with precision, over long distances.

We also will study how which aspects of a visualization environment can be successfully controlled by this kind of device. Under current consideration is lighting, paths of the viewpoints and lights, speed of the objects on their paths etc.

REFERENCES

Bischof, H.-P. and Dong, A. (2011). Directing a visualization ala kubrick. In *Proceedings of International Conference on Computer Graphics and Virtual Reality*. CSREA Press.

Brand, D. and Zafiropulo, P. (1983). On communicating finite-state machines. *J. ACM*, 30(2):323–342.

Burgiel, H. (1997). How to lose at tetris. *Mathematical Gazette*, 81:194–200.

Loyola, R. (2010). Apple's magic mouse offers multitouch features. Copyright - Copyright PC World Communications, Inc. Jan 2010; Document feature - Photographs; Last updated - 2012-07-19.

Pang, A. S.-K. (2002). The making of the mouse. *American Heritage of Invention and Technology*, 17(3):48–54.

Weichert, F., Bachmann, D., Rudak, B., and Fisseler, D. (2013). Analysis of the accuracy and robustness of the leap motion controller. *Sensors*, 13(5):6380–6393.

William Joy, M. H. L. (1977). An introduction to display editing with vian introduction to display editing with vi.

Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE MultiMedia*, 19(2):4–10.

A Calibration Framework of a Mixed-traffic Signal Optimization Model by Multi-objective Evolutionary Approach

H.-J. Cho[1,2], **M.-C. Hwang**[1], **and C.-C. Hsu**[1]
[1] Department of Transportation and Logistics Management, National Chiao Tung University, Hsinchu, Taiwan, R.O.C.
[2] Chairman of the board, China Engineering Consultants Inc., Taipei, Taiwan

Abstract - *Scooter-mixed traffic flow models have been investigated for decades since the motorcycle is one of the major modes for daily traffic in some Asian countries. But the calibration process of most of them are based on a single performance metric. In this study, a framework for parameter optimization of the mixed-traffic model with multi-objectives has been presented and a NSGA-II algorithm is coupled with the mixed-traffic flow model to find the non-dominated front in the objective space. An application example has been presented that illustrates the potential use of the proposed calibration framework.*

Keywords: Mixed Traffic Flow; NSGA-II; Multi-objective Evolutionary Approach

1 Introduction

1.1 Background

Scooter-vehicle mixed traffic flows is one of the most important traffic features in some Asian developing countries, such as China, Taiwan, Malaysia and Vietnam, etc. Due to scooter's maneuverability, agility, and low price compared to passenger car, it has become the primal transportation mode for commuters, especially in urban areas [7]. But unlike passenger car that usually moves along a specific lane and changes lanes only for overtaking or turning purpose, scooters moves in a rather irregular and erratic manner, which consequently creates more chaotic to the congested urban traffic flows. For the purpose of transportation system planning, design and control, it is critical to gain deep insights into the motorcycle behaviors and to develop appropriate traffic flow models. In the last decade, few papers concerning this topic in theoretical research [2, 3, 8, 9, 11] and some of them couple the traffic flow model with signal control model together to optimize signal timing plans [3, 8]. In all of these studies, the calibration procedure is based on a single performance metric or calibration criterion [1, 2, 6, 9]. The tradeoff between different objectives has been ignored. Therefore in this study, a multi-objective evolutionary algorithm known as Non-dominated Sorting Genetic Algorithm II (NSGA-II) [4] has been used to establish a calibration procedure for the mixed-traffic signal optimization model. Such algorithm employs a population-based approach to find multiple optimal solutions, or so-called Pareto optimal solutions.

The paper attempts to develop a calibration framework to optimize the parameters of a mixed-traffic flow model by using NSGA-II. The paper is organized as follows. First a short review to the scooter-mixed traffic flow model is provided. Then the parameter optimization problem and calibration framework is proposed. Section 3 describes the case study site and results of applying the proposed framework. The final section concludes this paper.

1.2 Scooter-Mixed Traffic Flow Model

The scooter-vehicle mixed traffic flow model used in the study was developed by [8, 9]. Consider a typical approach with mixed-traffic flows, the model conceptually divides the traffic dynamics into the following stages: a) Upstream arrivals and propagation to the end of queue, b) lane choice and merge into existing queue, c) discharging process from the queue. In the first stage, the model calculates the upstream arrival rate and estimated travel time from the upstream stop-line to the downstream end-of-queue. Then a lane choice model, as shown in (3), distributes the arrivals into different lanes according to each lane's queue length and saturation flow rate.

$$Q_v^{l,arr}[t] = Q_v^{arr}[t] \times \gamma_{m,v} \times \left[\frac{\tilde{L}^l[t] \times \tilde{S}^l[t-1]}{\sum_{l \in LG_m} \tilde{L}^l[t] \times \tilde{S}^l[t-1]} \right], \quad (1)$$

$$\tilde{L}^l[t] = \frac{\left(L^l[t]\right)^{-1}}{\sum_{l \in LG_m} \left(L^l[t]\right)^{-1}} \text{ and } \tilde{S}^l[t-1] = \frac{S^l[t-1]}{\sum_{l \in LG_m} S^l[t-1]},$$

where $Q_v^{l,arr}[t]$ is the flow rate of type-v (v=1 is scooters, v=2 is passenger cars) arriving the end of queue of lane l during time step t; $Q_v^{arr}[t]$ is the flow rate of type-v vehicles arriving the end of queue during time step t, $\gamma_{m,v}[t]$ is the turning ratio of type-v vehicles with movement m; LG_m is the set of lanes available to vehicles with movement m; $\tilde{L}^l[t]$ and $\tilde{S}^l[t-1]$ denote the normalized factor (between 0 and 1) of queue length and saturation flow rate on lane l, respectively.

If an approach has a scooter-waiting area, the model assumes that the scooters with through movement merge into the waiting area directly if there is space available. Once the waiting area is fully occupied, scooters will start to queue on available lanes. Let $Q^{swa}[t]$ denote the number of scooters merging into the waiting area, then (1) is rewritten as follows:

$$Q_1^{l,arr}[t] = (Q_1^{arr}[t] \times \gamma_{m,1} - Q^{swa}[t]) \times \left[\frac{\tilde{L}^l[t] \times \tilde{S}^l[t-1]}{\sum_{l \in LG_m} \tilde{L}^l[t] \times \tilde{S}^l[t-1]} \right] \quad (2)$$

Given the average storage space of a scooter waiting area χ, $Q^{swa}[t]$ is calculated by the follows:

$$Q_1^{swa}[t] = \min\left\{ Q_1^{arr}[t] \times \gamma_{thru,1}, \chi - X^{swa}[t] \right\} \times \left(1 - g^{thru}[t]\right) \quad (3)$$

where $X^{swa}[t]$ is the number of scooters in the scooter waiting area at time stage t and $g^{thru}[t]$ is a binary variable representing a green phase for through movement.

The vehicle queues on each lane keep cumulating until the traffic light turns to green. In the discharging process, the model describes how different types of vehicle interact and generates a mixed-traffic discharging rate weighted with the number of each type of vehicles in the queue. The average discharge rate $S^l[t]$ is expressed as follows:

$$S^l[t] = \left(\frac{\sum_v X_v^l[t] \cdot O_v \cdot h_{m,v}}{\sum_v X_v^l[t] \cdot O_v} \right)^{-1} \quad (4)$$

where $X_v^l[t]$ is the number of type-v vehicles in queue on lane l at time step t; O_v is the occupied space of type-v vehicles and $h_{m,v}$ is the average discharge headway of type-v vehicles with movement m. The actual discharging rate for each vehicle type and movement can be calculated as follows:

$$S_v^l[t] = S^l[t] \times \frac{X_v^l[t]}{\sum_v X_v^l[t]} \quad (5)$$

2 Multi-Objective evolutionary approach for parameter optimization

2.1 Parameter Optimization

The purpose of model calibration can be generally stated as estimation of model parameters so that model simulations and traffic behavior match closely. In the study, the estimation of the parameter values is formulated as a multi-objective optimization problem where two objectives, the absolute errors between the observed and simulated average number of passenger cars and scooters in queue, are considered. The optimization problem is written as follows:

$$\text{Minimize} \quad \left(\left| \bar{x}_{scot}^{obs} - \bar{x}_{scot}^{est}(p) \right|, \left| \bar{x}_{pc}^{obs} - \bar{x}_{pc}^{est}(p) \right| \right) \quad (6)$$

$$\text{Subjected to} \quad p_{LB} \le p \le p_{UB} \quad (7)$$

Where, $\bar{x}_{scot}^{obs}$ and $\bar{x}_{scot}^{est}(p)$ denote the observed and estimated average number (of all movements) of scooter in queue, respectively; $\bar{x}_{pc}^{obs}$ and $\bar{x}_{pc}^{est}(p)$ are same values of passenger cars. The vector of parameters is represented by p, with a lower bound p_{LB} and upper bound p_{UB}. The objectives (6) is to minimize the absolute error of average number of scooter in queue and the same of passenger cars. For each approach, the parameter vector has seven elements, including the average discharging headway $h_{m,v}$ for three different movements and two vehicle types and the average storage space of a scooter waiting area χ. The value of p_{LB} and p_{UB} are given according by the field survey data.

2.2 Calibration Framework

The proposed framework of calibration procedure for the mixed-traffic flow model is shown in Fig. 1. The main optimization process is on the left side, which can be any kind of multi-objective evolutionary algorithms (MOEA), i.e., Genetic Algorithm, Particle Swarm, etc. In this study, NSGA-II approach is used. In a sense, these population-based approaches have similar solution structure. At the beginning, a pool of solution candidates are generated and evaluated. Then a procedure of reproduction is revoked and a new generation of solution set is created. This procedure is iterated until the stop conditions are met.

On the other hand, the evaluation procedure is depicted on the right side of Fig. 1. Since the purpose of parameter optimization is to minimize the estimation error, the mixed-traffic model is revoked when a solution is evaluated. The input file of the mixed-traffic model is modified based on the set of parameters represented by the solution being evaluated. Then the model output is analyzed and compared with field survey observations. The differences between estimations and observations are objective values of the solution.

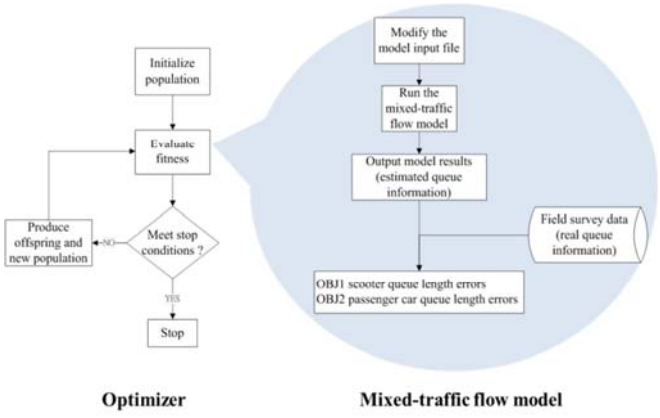


Figure 1 The model calibration framework

3 Application Example

3.1 Survey Area and Data

The survey field site for this study is a major intersection in Hsinchu County, Taiwan. The data was collected on Oct. 30, 2014 between 05:00 PM and 08:00 PM. The peak-hour demand during 05:35 PM to 06:35 PM is used as the model input, as shown in Table 1. The number of passenger cars and scooters in the queue on each lane were collected in intervals of fifteen seconds. The data was aggregated to the form of average number of vehicles in queue of each movement, as shown in Table 2.

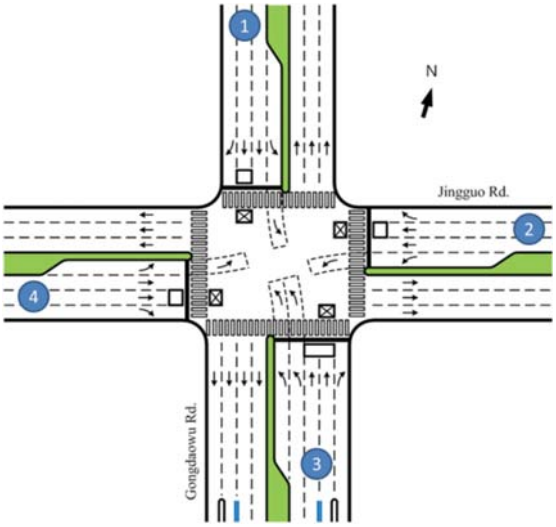


Figure 2 The survey field site

Table 1 Peak-hour demand (Unit: PCUs)

Approach	Movement / v-type	Left	Through	Right	Total
1 (SB)	Scooter	7.0	199.9	23.4	230.3
	Passenger Car	91.5	520.6	530.8	1142.9
	Total	**98.5**	**720.5**	**554.2**	**1373.2**
2 (WB)	Scooter	90.5	393.5	18.4	502.5
	Passenger Car	330.3	533.2	38.9	902.4
	Total	**420.8**	**926.7**	**57.3**	**1404.9**
3 (NB)	Scooter	92.5	359.1	115.2	566.8
	Passenger Car	538.8	484.4	365.6	1517.8
	Total	**631.3**	**843.5**	**480.8**	**2084.6**
4 (EB)	Scooter	22.0	280.6	48.6	351.2
	Passenger Car	153.9	729.0	236.5	1119.4
	Total	**175.9**	**1009.6**	**285.1**	**1470.6**

Table 2 Observed average numbers of vehicles in queue

Approach	movement / v-type	Left	Through	Right
1 (SB)	Scooter	-	0.102	0.686
	Passenger Car	0.636	7.326	3.203
2 (WB)	Scooter	-	1.678	4.136
	Passenger Car	2.466	4.008	0.703
3 (NB)	Scooter	-	0.0254	2.381
	Passenger Car	16.076	11.678	0.737
4 (EB)	Scooter	-	0.458	1.788
	Passenger Car	1.65	7.076	2.424

3.2 Result and Discussion

The optimized parameters were evaluated by calculating the absolute errors between the observations and model simulation results. The pareto front of each approach is shown in Figure 3-6, respectively. It is obvious that approach 1 does not have tradeoff between two objectives. We may notice that the total demand volume (in Table 2) on the approach is lighter than other approaches and it also has the lowest total scooter volume among all movements. By observing the formulation of lane choice model and discharging model mentioned before, we may notice that different compositions of the mixed traffic, i.e., the number of vehicles in the queue, have great influence on the mixed-traffic saturation flow rate. Since all the scooters that cannot go to the waiting zone will merge into the queue with passenger cars, the parameter has a great deal to do with the mixed-traffic composition. In other words, the approach with higher volume of scooter demand has greater chance of appearance of tradeoff.

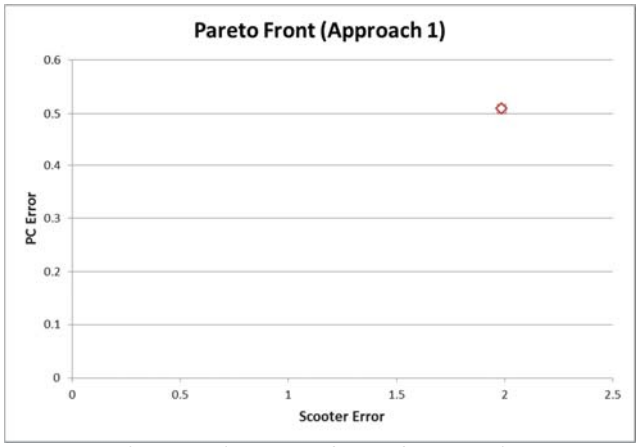


Figure 3 The Pareto front of Approach 1

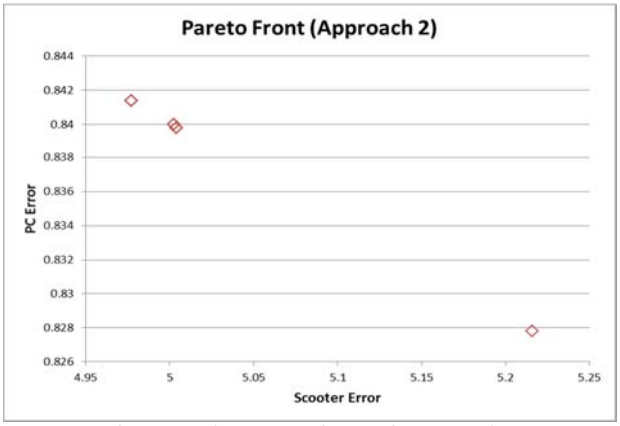


Figure 4 The Pareto front of Approach 2

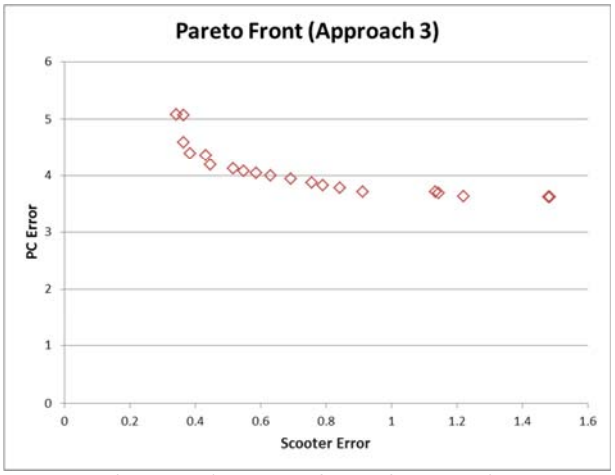


Figure 5 The Pareto front of Approach 3

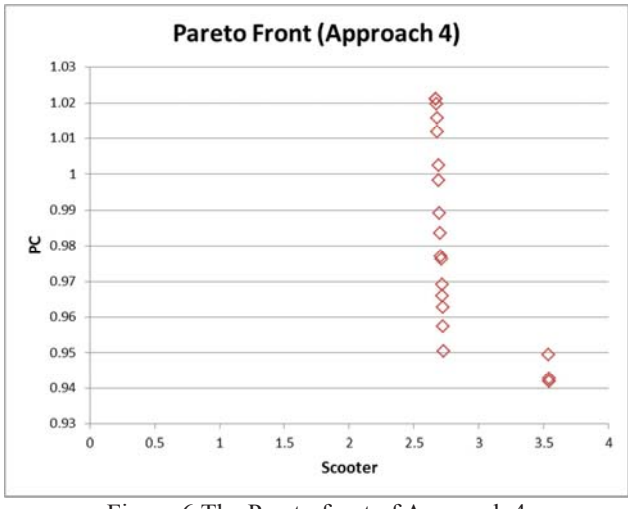


Figure 6 The Pareto front of Approach 4

4 Conclusion

In this study, a framework for parameter optimization of the mixed-traffic model with multi-objectives has been presented. In the framework, a NSGA-II algorithm is coupled with the mixed-traffic flow model to find the non-dominated front in the objective space. An application example has been presented that illustrates the use of the proposed calibration framework. It can be concluded that the proposed calibration framework has potential use in parameter optimization of the mixed-traffic flow model. From the results, the phenomenon of tradeoff between scooter and passenger car errors become obvious when scooter demand rises. Such phenomenon quite coincides with the chaotic nature of mixed-traffic flow in real world.

Acknowledgment

This research was partially supported by the Ministry of Science and Technology through contract number 103-2221-E-009-158 and 103-2622-E-009-012; Ministry of Transportation and Communications R.O.C. under contract number MOTC-STAO-103-02 and MOTC-STAO-103-03; China Engineering Consultants, Inc., under contract number CECI-01921. We also acknowledge the reviewers' valuable suggestion for the further research topics.

5 References

[1] Balakrishna, R., Antoniou, C., Ben-Akiva, M., Koutsopoulos, H. N., & Wen, Y. 2007. Calibration of microscopic traffic simulation models: Methods and application. Transportation Research Record 1999, 198-207.

[2] Chiou, Y. C., Zhou, L., and Song, H., 2010, Development and Calibration of the Anisotropic Mesoscopic Simulation Model for Uninterrupted Flow Facilities, Transportation Research-B, 44 (1), pp. 152-174.

[3] Chiou, Y. C., Huang, Y. F., 2013, Stepwise genetic fuzzy logic signal control under mixed traffic conditions, JOURNAL OF ADVANCED TRANSPORTATION 47:43-60

[4] Deb, K., Pratap, A.,Agarwal, S., Meyarivan, T., 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6 (2), 182-197.

[5] Efstratiadis, A., Koutsoyiannis, D., 2010. One decade of multi-objective calibration approaches in hydrological modelling: a review. Hydrological Sciences Journal 55 (1), 58-78.

[6] Hourdakis, J., P.G. Michalopoulos, and J. Kottommannil, 2003. A Practical procedure for calibrating microscopic traffic simulation models. Transportation Research Board Annual Meeting 1852, 130-139.

[7] Hsu, T.-P., Mohd Sadullah, A. F., Nguyen , X. D., 2003. A comparative study on motorcycle traffic development of Taiwan, Malaysia and Vietnam. Journal of the Eastern Asia Society for Transportation Studies 5, 381–395.

[8] Lan, C.-L., Chang, G.-L., 2014. A Traffic Signal Optimization Model for Intersections Experiencing Heavy Scooter–Vehicle Mixed Traffic Flows. Intelligent Transportation Systems, IEEE Transactions on DOI: 10.1109/TITS.2014.2376292.

[9] Lan, C.-L., Chang, G.-L., 2015. An empirical study of the scooter-vehicle mixed traffic propagation on urban arterials. In: 94th TRB Annual Meeting, Washington, DC.

[10] Lan, L.W., Chang, C.-W., 2005. Inhomogeneous cellular automata modeling for mixed traffic with cars and motorcycles. Journal of Advanced Transportation 39 (3), 323-349.

[11] Meng, J.-P., Dai, S.-Q., Dong, L.-Y., Zhang, J.-F., 2007. Cellular automaton model for mixed traffic flow with motorcycles. Physica A: Statistical Mechanics and its Applications 380 (1), 470-480.

[12] Sun, D., Benekohal, R.F., Waller, S.T., 2003. Multiobjective traffic signal timing optimization using non-dominated sorting genetic algorithm. IEEE Intelligent Vehicles Symposium. Proceedings, 198 - 203

.

MATHEMATICAL MODEL OF THE MANUFACTURING INDUSTRY OF THE REPUBLIC OF KAZAKHSTAN

The 2015 International Conference on Modeling, Simulation and Visualization Methods,
Las Vegas, Nevada, USA,
July 27-30, 2015

Seilkhan Boranbayev
L.N. Gumilyov Eurasian National University
5 Munaitpasov Street
Astana, 010008, Kazakhstan
sboranba@yandex.kz

Askar Nurbekov
L.N. Gumilyov Eurasian National University
5 Munaitpasov Street
Astana, 010008, Kazakhstan
nurbekoff@gmail.com

Abstract— *This paper reviews the construction of production functions for simulating the operation of the manufacturing industry of the Republic of Kazakhstan. Several production function models were developed. Economic indicators of manufacturing industry of Kazakhstan were calculated on the considered production functions as well as the most appropriate for the industry were determined.*

Keywords— model, method, modeling, forecasting, production function, capital.

1. Introduction

In paper (1) for the forecast the level of gross domestic product for the manufacturing industry of the Republic of Kazakhstan was constructed and studied 7 models of production functions

Namely: linear; Cobb-Douglas with $\alpha + \beta = 1$; Cobb-Douglas with $\alpha + \beta \neq 1$; Cobb-Douglas, taking into account scientific and technical progress when $\alpha + \beta = 1$; Cobb-Douglas, taking into account scientific and technical progress when $\alpha + \beta \neq 1$; Quadratic (type 1); Quadratic (type 2).

For mathematical modeling of operation of the manufacturing industry of the Republic of Kazakhstan using data on gross domestic product this industry for 1998-2013 years. relatively labor (L) and capital (K). These are listed in Table 1.

Table 1. Economic indices of the manufacturing industry of the Republic of Kazakhstan for 1998-2013 years. [2,3]

Years	K – capital costs (million. tenge)	L – the cost of workers' wages (million. tenge.)	Y – gross domestic product sector (million. tenge.)
1998	40 618,00	102 893,40	208 336,60
1999	52 907,28	130 240,20	284 152,00
2000	74 794,93	149 259,00	428 932,70
2001	102 421,81	167 483,10	534 563,00
2002	102 550,03	172 655,70	547 414,10
2003	119 870,48	213 417,00	655 719,00
2004	191 366,17	272 891,30	781 558,70
2005	258 886,78	325 058,20	914 013,20
2006	293 475,13	423 004,30	1 188 108,00
2007	316 339,43	551 380,20	1 476 647,60
2008	370 062,97	669 651,40	1 890 053,00
2009	396 261,47	643 251,10	1 849 097,50
2010	404 925,35	821 158,50	2 469 804,10
2011	455 466,43	989 957,40	3 131 187,00
2012	595 214,22	1 066 127,50	3 436 730,50
2013	636 886,41	1 130 987,00	3 651 704,60

Select optimal model of from the constructed models of production functions. The choice of the optimal model carried out in two stages.

2. Construction of optimal mathematical model

2.1. Regression analysis of the data (the first stage).

For each of the constructed production functions was made regression analysis. The regression analysis of the data given in Table 2.

Table 2. Results of regression analysis of data

Model of production function	Coefficient of determination	Standard error	Sums of deviations squares
Linear	0,9927608	101 796,940	190 638 377 603,900
Cobb-Douglas with α+β=1	0,9928107	101 445,767	194 254 528 383,208
Cobb-Douglas withα+β≠1	0,9959064	76 549,895	87 250 448 602,112
Cobb-Douglas, taking into account scientific and technical progress when α+β=1	0,9928614	101 087,447	194 356 305 899,109
Cobb-Douglas, taking into account scientific and technical progress when α+β≠1	0,9729569	196 752,076	1 010 933 878 402,360
Quadratic (type1)	0,9711106	203 357,327	2 740 703 153 148,110
Quadratic (type 2)	0,9721592	199 632,800	2 778 133 533 860,310

Criterion for selecting is the following: the largest value of the coefficient of determination, the smallest standard error and the smallest sum of squared deviations.

As can be seen from Table 2, the production function is Cobb-Douglas with $\alpha + \beta \neq 1$ the most fit for the specified selection criteria. In addition, the three models of the production function also satisfy the selection criteria:

• Linear
• Cobb-Douglas with $\alpha + \beta = 1$
• Cobb-Douglas, taking into account scientific and technical progress with $\alpha + \beta = 1$

Exactly these 4 models will be considered at the second stage of the selection of the optimal model.

2.2 Building an additional models (second stage)

The second stage of choosing the optimal model is to build additional models based on the models selected in the first stage.

Algorithm for selecting the optimal model on supplementary models:

•An the initial data come from economic indicators for the manufacturing industry of Kazakhstan 1998-2010 years.
• On the basis of this data is constructed corresponding additional model
• additional models based on known economic indicators K and L for the 1998-2010 years. evaluated forecast data for the 2011-2013 F.
• Compare the actual values of GDP (Y) and the values obtained additional model (F) for 2011-2013.

An additional linear model of the production function.

Will use the linear production function.

$$F = a_0 + a_1 K + a_2 L ,$$

where K - the cost of capital; L - salary costs.

Function inconsistencies is:

$$\sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} \left[Y_i - (a_0 + a_1 K_i + a_2 L_i) \right]^2 \to \min_{a_0, a_1, a_2}$$

Carrying out the computation using data of Table 1. As a result we obtain that the function is reaches a minimum at inconsistencies when a0 = -0.000009; a1 = -0.374; a2 = 3.087.

With reference to our data model more linear production function has the form:

$$F = -0{,}000009 - 0{,}374K + 3{,}087L$$

Table 3. Economic indices of the manufacturing industry of Kazakhstan for 1998-2010. with additional calculations based on linear production function

Years	K	L	Y	F	$(Y-F)^2$
1998	40 618,00	102 893,40	208 336,60	302 422,628	8 852 180 695,121
1999	52 907,28	130 240,20	284 152,00	382 241,246	9 621 500 227,979
2000	74 794,93	149 259,00	428 932,70	432 763,575	14 675 604,602
2001	102 421,81	167 483,10	534 563,00	478 686,598	3 122 172 325,009
2002	102 550,03	172 655,70	547 414,10	494 605,461	2 788 752 317,633
2003	119 870,48	213 417,00	655 719,00	613 950,607	1 744 598 640,100

2004	191 366,17	272 891,30	781 558,70	770 799,990	115 749 846,853
2005	258 886,78	325 058,20	914 013,20	906 579,353	55 262 083,156
2006	293 475,13	423 004,30	1 188 108,00	1 195 985,448	62 054 186,578
2007	316 339,43	551 380,20	1 476 647,60	1 583 706,720	11 461 655 153,864
2008	370 062,97	669 651,40	1 890 053,00	1 928 696,739	1 493 338 569,457
2009	396 261,47	643 251,10	1 849 097,50	1 837 407,041	136 666 822,441
2010	404 925,35	821 158,50	2 469 804,10	2 383 332,782	7 477 288 861,870

On the basis of K and L for years 1998-2010. compute the supplementary model predicted values F (Table 4).

Table 4. Economic indices of for the manufacturing industry of Kazakhstan for 2011-2013. with additional calculations based on linear production function

Year	K	L	Y	F_{add}	$F_{basic.}$	$\|Y-F_{.add}\|$	$\|Y-F_{basic.}\|$
2011	455 466,43	989 957,4	3 131 187,0	2 882,113	8853 115 434,875	245 704,887	15 752,125
2012	595 214,22	1 066 127,5	3 436 730,5	3 344,909	0683 265 372,922	368 385,591	171 357,578
2013	636 886,41	1 130 987,0	3 651 704,6	3 970,039	2523 459 344,262	398 734,561	192 360,338

An additional model Cobb-Douglas with α + β = 1.

We construct a production function of the Cobb-Douglas form:

$$F = AK^{\alpha}L^{\beta},$$

Where α + β = 1, K - cost of capital; L - salary costs.

Function inconsistencies is:

$$\sum_{i=1}^{n}\varepsilon_i^2 = \sum_{i=1}^{n}\left[Y_i - (\mathrm{a}_0 K_i^{\,a_1} L_i^{(1-a_1)})\right]^2 \to \min_{a_0,a_1}$$

Carrying out the computation using data of Table 1. As a result we obtain that the function is reaches a minimum at inconsistencies a0 = 2.756; a1 = -0.072.

With reference to our data model more linear production function has the form:

$$F = 2{,}756K^{-0{,}072}L^{1{,}072}$$

Table 5. Economic indices of for the manufacturing industry of Kazakhstan 1998-2010. with additional calculations based on a Cobb-Douglas with α + β = 1

Years	K	L	Y	F	$(Y-F)^2$
1998	40 618,00	102 893,40	208 336,60	303 235,738	9 005 846 471,126
1999	52 907,28	130 240,20	284 152,00	383 035,100	9 777 867 556,019
2000	74 794,93	149 259,00	428 932,70	432 357,733	11 730 849,702
2001	102 421,81	167 483,10	534 563,00	478 212,158	3 175 417 386,146
2002	102 550,03	172 655,70	547 414,10	494 022,031	2 850 713 046,856
2003	119 870,48	213 417,00	655 719,00	613 125,263	1 814 226 427,390
2004	191 366,17	272 891,30	781 558,70	771 508,793	101 000 637,589
2005	258 886,78	325 058,20	914 013,20	910 575,884	11 815 139,734
2006	293 475,13	423 004,30	1 188 108,00	1 196 828,944	76 054 868,487
2007	316 339,43	551 380,20	1 476 647,60	1 581 629,847	11 021 272 088,434
2008	370 062,97	669 651,40	1 890 053,00	1 926 101,397	1 299 486 916,572
2009	396 261,47	643 251,10	1 849 097,50	1 835 693,889	179 656 790,545
2010	404 925,35	821 158,50	2 469 804,10	2 381 412,360	7 813 099 722,069

On the basis of K and L for years 1998-2010. compute the supplementary model predicted values F (Table 6).

Table 6. Economic indices of the manufacturing industry of Kazakhstan for 2011-2013. with additional calculations based on a Cobb-Douglas with α + β = 1

Years	K	L	Y	$F_{add.}$	$F_{basic.}$	$\|Y-F_{add.}\|$	$\|Y-F_{basic}\|$
2011	455 466,43	989 957,4	3 131 187,0	2 885 365,842	3 112 717,883	245 821,158	18 469,117
2012	595 214,22	1 066 127,5	3 436 730,5	3 064 211,235	3 263 101,961	372 519,265	173 628,539
2013	636 886,41	1 130 987,0	3 651 704,6	3 248 603,857	3 457 468,440	403 100,743	194 236,16

An additional model Cobb-Douglas with α + β ≠ 1.

We construct a production function of the Cobb-Douglas form:

$$F = AK^{\alpha}L^{\beta},$$

Where α + β ≠ 1, K - cost of capital; L - salary costs.

Function inconsistencies is:

$$\sum_{i=1}^{n}\varepsilon_i^2 = \sum_{i=1}^{n}\left[Y_i - (\mathrm{a}_0 K^{a_1} L^{a_2})\right]^2 \to \min_{a_0,a_1,a_2}$$

Carrying out the computation using data of Table 1. As a result we obtain that the function is reaches a minimum at inconsistencies a0 = 1.923; a1 = -0.028; a2 = 1.057.

With reference to our data model more linear production function has the form:

$$F = 1{,}923K^{-0{,}028}L^{1{,}057}$$

Table 7. Economic indices of the manufacturing industry of Kazakhstan for 1998-2010. with additional calculations based on a Cobb-Douglas with $\alpha + \beta \neq 1$

Years	K	L	Y	F	$(Y-F)^2$
1998	40 618,00	102 893,40	208 336,60	283 976,384	5 721 376 885,953
1999	52 907,28	130 240,20	284 152,00	361 631,670	6 003 099 255,927
2000	74 794,93	149 259,00	428 932,70	413 645,831	233 688 360,711
2001	102 421,81	167 483,10	534 563,00	463 116,560	5 104 593 804,662
2002	102 550,03	172 655,70	547 414,10	478 232,474	4 786 097 365,138
2003	119 870,48	213 417,00	655 719,00	595 720,646	3 599 802 430,309
2004	191 366,17	272 891,30	781 558,70	762 440,689	365 498 362,247
2005	258 886,78	325 058,20	914 013,20	909 572,345	19 721 196,701
2006	293 475,13	423 004,30	1 188 108,00	1 197 358,976	85 580 554,741
2007	316 339,43	551 380,20	1 476 647,60	1 581 205,943	10 932 447 096,865
2008	370 062,97	669 651,40	1 890 053,00	1 933 281,785	1 868 727 848,512
2009	396 261,47	643 251,10	1 849 097,50	1 849 258,299	25 856,191
2010	404 925,35	821 158,50	2 469 804,10	2 392 402,203	5 991 053 678,765

On the basis of K and L for years 2011-2010. compute the supplementary model predicted values F (Table 8).

Table 8. Economic indices of the manufacturing industry of Kazakhstan for 2011-2013. with additional calculations based on a Cobb-Douglas with $\alpha + \beta \neq 1$

Years	K	L	Y	$F_{add.}$	$F_{basic.}$	$\|Y-F_{add}\|$	$\|Y-F_{basic.}\|$
2011	455 466,43	989 957,4	3 131 187,0	2 905 541,562	3 102 812,777	225 645,438	28 374,223
2012	595 214,22	1 066 127,5	3 436 730,5	3 118 900,463	3 375 271,455	317 830,037	61 459,045
2013	636 886,41	1 130 987,0	3 651 704,6	3 313 528,523	3 607 262,340	338 176,077	44 442,26

An additional model Cobb-Douglas, taking into account scientific and technical progress with α + β = 1.

We construct a production function of the Cobb-Douglas, taking into account the type of STP:

$$F = Ae^{p_0 t}K^{\alpha}L^{(1-\alpha)},$$

Where $\alpha + \beta = 1$, K - cost of capital; L - expenditure on wages, $e^{p_0 t}$ - a special factor of technological progress, p_0 - neutral setting NTP ($p_0 > 0$).

Function inconsistencies is:

$$\sum_{i=1}^{n}\varepsilon_i^2 = \sum_{i=1}^{n}\left[Y_i - (a_0 e^{p_0 t}K^{a_1}L^{(1-a_1)})\right]^2 \to \min_{a_0, a_1, p_0}$$

Carrying out the computation using data of Table 1. As a result we obtain that the function is of discrepancies reaches a minimum at p0 = 0,0014; a0 = 2.752; a1 = -0.073.

With reference to our data model more linear production function has the form:

$$F = 2{,}752e^{0{,}0014t}K^{-0{,}073}L^{1{,}073}$$

Table 9: Economic indices of for the manufacturing industry of Kazakhstan 1998-2010. with additional calculations based on a Cobb-Douglas, taking into account scientific and technical progress with $\alpha + \beta = 1$

Years	K	L	Y	F	$(Y-F)^2$
1998	40 618,00	102 893,40	208 336,60	303 484,036	9 053 034 597,986
1999	52 907,28	130 240,20	284 152,00	383 342,435	9 838 742 464,790
2000	74 794,93	149 259,00	428 932,70	432 652,493	13 836 859,582
2001	102 421,81	167 483,10	534 563,00	478 483,461	3 144 914 669,093
2002	102 550,03	172 655,70	547 414,10	494 310,582	2 819 983 676,471
2003	119 870,48	213 417,00	655 719,00	613 503,067	1 782 184 973,543
2004	191 366,17	272 891,30	781 558,70	771 885,814	93 564 726,972
2005	258 886,78	325 058,20	914 013,20	910 954,291	9 356 924,460
2006	293 475,13	423 004,30	1 188 108,00	1 197 421,171	86 735 153,451
2007	316 339,43	551 380,20	1 476 647,60	1 582 585,148	11 222 764 075,593
2008	370 062,97	669 651,40	1 890 053,00	1 927 306,230	1 387 803 153,460
2009	396 261,47	643 251,10	1 849 097,50	1 836 727,608	153 014 240,135
2010	404 925,35	821 158,50	2 469 804,10	2 383 057,893	7 524 904 489,558

On the basis of K and L for years 1998-2010. compute the supplementary model predicted values F (Table 10).

Table 10. Economic indices of the manufacturing industry of Kazakhstan for 2011-2013. with additional calculations based on a Cobb-Douglas, taking into account scientific and technical progress with $\alpha + \beta = 1$

Years	K	L	Y	$F_{add.}$	$F_{basic.}$	$\|Y-F_{add.}\|$	$\|Y-F_{basic}\|$
2011	455 466,43	989 957,4	3 131 187,0	2 887 474,541	3 108 046,491	243 712,459	23 140,509
2012	595 214,22	1 066 127,5	3 436 730,5	3 066 110,001	3 264 630,365	370 620,499	172 100,135
2013	636 886,41	1 130 987,0	3 651 704,6	3 250 600,810	3 459 391,254	401 103,79	192 313,346

2.3 Choice of the optimal model

Results of building additional models we reduce to a single table 11.

Table 11. Indicators built by additional models

Production function model	2011 г.	2012 г.	2013 г.
Linear	2 885 482,113	3 068 344,909	3 252 970,039
Cobb-Douglas with $\alpha + \beta = 1$	2 885 365,842	3 064 211,235	3 248 603,857
Cobb-Douglas with $\alpha + \beta \neq 1$	2 905 541,562	3 118 900,463	3 313 528,523
Cobb-Douglas, taking into account scientific and technical progress with $\alpha + \beta = 1$	2 887 474,541	3 066 110,001	3 250 600,810
The real value	3 131 187	3 436 730,5	3 651 704,6

3. Conclusion

As can be seen from Table 10, the data obtained from the model of the production function of the Cobb-Douglas with $\alpha + \beta \neq 1$ most similar to the real data. Therefore, for a given industry as a best we choose this model.
Select the optimal model of the production function of the Cobb-Douglas with $\alpha + \beta \neq 1$ can be used to predict the values of gross domestic product based on known or expected levels of capital and wage costs.

REFERENCES

1. Boranbayev S.N., Nurbekov A.B., Orakbayev E.M., Tlepov D.A. Mathematical modeling of the functioning of the manufacturing industry of the Republic of Kazakhstan // Herald L.N. Gumilyov Eurasian national University. -2014, №6, p. 19-29.
2. http://www.stat.gov.kz/
3. http://taldau.stat.kz/

SESSION

SIMULATION, NUMERICAL METHODS, AND APPLICATIONS

Chair(s)

TBA

Towards an Agent-Based Simulation of Predators Developing a Search Image

Chris Scogings
Computer Science
Massey University - Albany Campus
North Shore 102-904, Auckland
New Zealand
email: c.scogings@massey.ac.nz

ABSTRACT

Predators build a "search image" to rapidly detect common prey and exclude other possibilities. Prey respond with "apostatic selection" and produce mutations with previously unknown color variations. This process has been observed since the 1960s but is difficult to validate with real populations. The use of agent-based simulations presents the opportunity to explore specific changes to predators and/or prey in a controlled environment using artificial animals. This article investigates the effects on predator and prey populations of specific changes in behaviour and finds that search images may actually be detrimental to predator success and that prey color mutation may benefit the overall predator population.

KEY WORDS
agent-based model; animat; search image; apostatic selection; intelligent agents.

1 Introduction

Predators are said to develop a "search image" that enables them to rapidly detect familar prey and all but exclude other prey that may be equally suitable but more difficult to detect [1, 2].

Thus common forms of a species are preyed on more than the rarer types and it is believed that this tends to assist in the survival of new (rare) mutations while simultaneously ensuring that the most common types are frequently culled and kept in check [3]. This is known as "apostatic selection". It is useful to keep in mind that apostatic selection refers to prey (and their evolution) whereas search image refers to predators (and their effectiveness in detecting prey).

"Because it is harder to search simultaneously for two or more cryptic prey types than to search for only one, visual predators should tend to focus on the most abundant forms and effectively overlook the others. The result should be apostatic selection but validating this elegant hypothesis has been difficult." [4]

This problem is common across the biological sciences where it is often difficult (or impossible) to construct repeatable experiments to validate theory – particularly in the area of evolutionary biology. There have been some interesting experiments that combine real subjects with virtual environments [5, 6] including blue jays searching for digitial moths on a variety of backgrounds [7]. The experiments cited above, and the experiments described in this article, focus on prey where the difference between types is one of color.

One approach is to develop artificial simulations and then to use these as platforms for various experiments. This has not yet been fully achieved for a number of reasons including: considerable computer programming skill is needed; a significant number of precise parameters is required (of which some are unknown and others are only partly known); it is difficult to measure the accuracy of the simulation. In general, "off-the-shelf" simulation packages are not sufficient and the software has to be developed from scratch.

Agent-based models offer a possible way forward. The classic agent-based "artificial life" models [8–12] were relatively simple and were based around the concept of an artificial animal (or "animat" [13]). They did not attempt in any way to simulate real biology. There has been some work around the use of agent-based models to simulate aspects of real animal behaviour including sentinels [14], camouflage [15], termites [16], and hunting in packs [17].

The work described here uses a well-established predator-prey model [18] as a first step towards achieving a more accurate simulation of the formation of search images by predators. The model has been shown [19] to reproduce the typical population cycles of equation-based models such as the Lotka-Volterra equations [20]. The model has been continously refined over several years [21, 22] and has recently been advanced to a new level of complexity with the inclusion of gestation [23].

Figure 1 shows a typical situation at step 400 of the predator-

prey model. Over 100,000 agents are visible and successful predators can be seen clustering along the edges of large groups of prey.

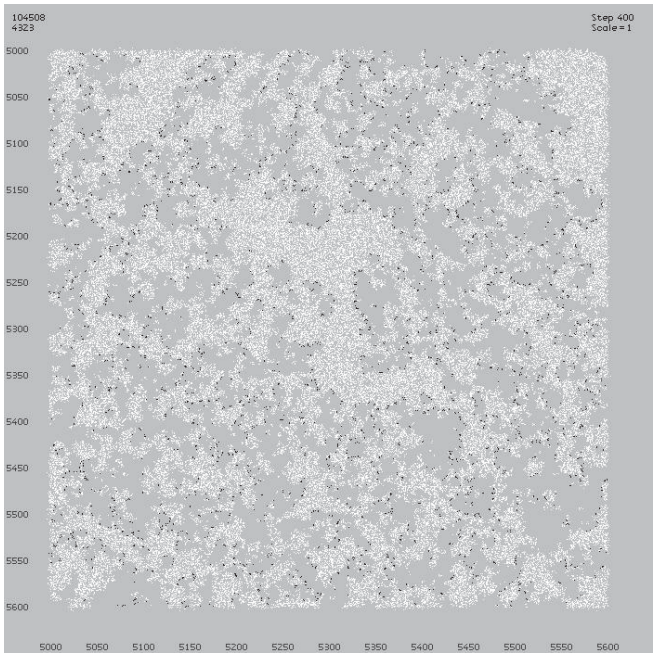


Figure 1: The situation at step 400 of a typical run showing animats on a square grassed area. There are 4,323 predators (black) and 104,508 prey (white) in this display.

In this paper we explore what is required in an agent-based model to successfully simulate the concept of predators forming a "search image" in order to focus on one type of prey. The predator-prey model is outlined in Section 2. The concept of a predator "detection rate" is introduced in Section 3. Section 4 investigates the effects of individual predators increasing the detection rate over time. Section 5 models the prey response in which prey mutate and change over generations. Some conclusions and ideas for future work are presented in Section 6.

2 The Predator-Prey Animat Model

The model includes two "species" of agents – the predators and the prey. Predators move, breed and catch prey. Prey move, breed and graze (consume "grass"). For these experiments, a large square area of grass is placed on the map. Grass is continously present and does not change its location. The square grassed area thus limits the location of the prey, which in turn limits the location of the predators. Previous work [24] has shown that these limitations do not affect the clustering and movement of the agents.

Every agent maintains its current state including: age, health, and location (x-y coordinates on the 2-dimensional map). When a new agent is produced, its health is set to the health of its parent. From then on, the health is reduced at each time step and if it reaches zero the agent dies. The only way to increase the health is by eating. Details of agent health and eating behaviours are discussed in [25].

Every agent also includes a short set of simple rules. In every time-step of the simulation, each agent updates its state variables by executing one of the rules. The order of the rules is important – each time-step, every animat attempts to execute the first rule in its rule set. However, most rules have conditions and often cannot be executed. If the conditions for the first rule can not be satisfied, the agent attempts to execute the next rule in the set and so on. The rules are listed below:

Rules for predators:

1. if health $>$ 50% then move towards nearest mate
2. if health $<$ 50% then move towards nearest prey
3. if health $<$ 50% and prey is adjacent then hunt
4. if health $>$ 50% and mate is adjacent then breed
5. flip a coin: heads = move randomly; tails = do nothing

Rules for prey:

1. if health $<$ 50% then eat "grass"
2. if health $>$ 50% and mate is adjacent then breed
3. if health $>$ 50% then move towards nearest mate
4. flip a coin: heads = move randomly; tails = do nothing

Each rule has several conditions that must be satisfied before that rule can be executed. In addition, there is still a possibility that nothing may be achieved even if the rule is executed. Breeding only has a certain chance of success. This is a simple alternative to factoring in a range of difficult parameters including nutrition, adequate shelter and so on. Changing the chances of a successful birth can dramatically alter the number of agents. For these experiments the chance of a successful birth was set to 30% for predators and 90% for prey.

Experimentation with various rule sets [26] has led to the rules that are described above. Further details on the sequencing and updating processes of each time step in the simulation can be found in [27]. Predators may not always succeed when they hunt prey as the chance of success is governed by the "detection rate" which is introduced in section 3 below.

3 Fixed Detection Rates

Each predator is assigned a "detection rate" which is an integer value in the range 50 to 100. The detection rate is the percentage chance that the predator will detect (and then catch) a

prey animat. For example, a predator with a detection rate of 62 has a 62% of successfully detecting, catching and eating a prey animat. The detection rate is checked each time the predator wants to catch prey. Thus a predator may successfully catch one prey animat but may not succeed in catching another. Detection is checked when the predator is adjacent to prey and uses the "eat" rule. If the predator does not detect the prey it continues checking for adjacent prey animats. If a predator is adjacent to several prey, it will attempt to detect each one in turn until one prey animat is successfully detected (and consumed) or the predator fails to detect any of the adjacent prey. If the predator fails to detect any adjacent prey, the predator does not eat and thus does not receive the increased health from eating.

In this experiment, all predators in one simulation are assigned the same fixed detection rate. The detection rate does not change during each simulation. Several simulations were run with the fixed detection rates 60, 70, 80 and 90. Simulations were not run with detection rates of 50 or less because experiments showed that predators with detection rates of 50 or less could not form a viable population.

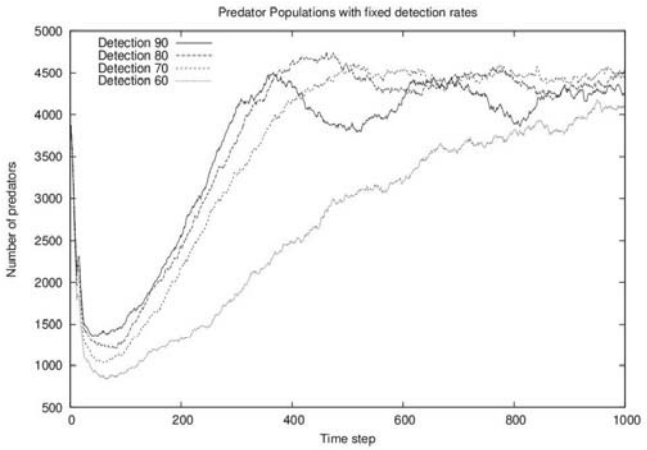


Figure 2: Plot showing predator populations for groups with different detection rates. The detection rate for each group is fixed. Predators with higher detection rates are able to catch prey more effectively. However, the highest detection rate leads to a slightly lower population than other groups.

Figure 2 shows predator populations from simulations where each simulation has a different fixed detection rate. As the detection rate increases, individual predators are able to catch prey more successfully and hence the predator population increases. However, with a detection rate of 90% the predators become too efficient and reduce the prey population to the point where predators cannot easily find prey and the predator population drops until the prey population starts to rise again. This leads to the "boom-bust" effect typical of theoretical predator-prey models [28].

Figure 3 shows the corresponding prey populations from the same simulations. The prey populations can be seen to be lower for higher fixed detection rates. The prey populations for detection rates of 80 and 90 are similar (and overlapping) due to the "boom-bust" effects discussed above.

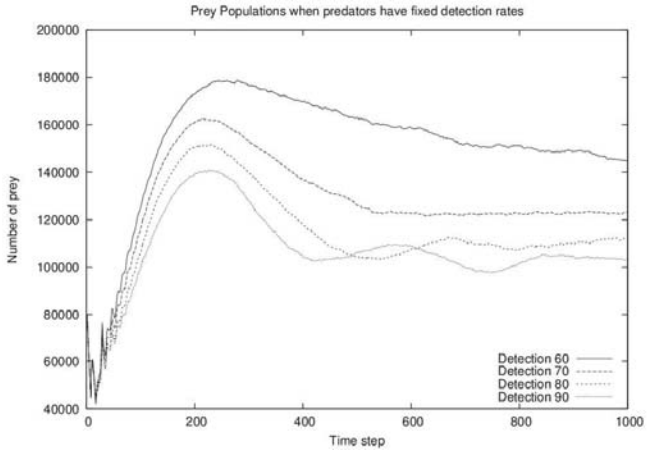


Figure 3: Plot showing prey populations when the predators have different detection rates. The detection rate for each group is fixed. Predators with higher detection rates are able to catch prey more effectively and the prey population is decreased accordingly.

Figure 4 shows the situation at step 400 of a run where predators have a fixed detection rate of 60. This is a lower detection rate and predators are not as effective at catching prey. Thus the predator population has decreased while the prey population has increased. The clusters of prey are significantly more dense than those shown in Figure 1 in which predators have a fixed detection rate of 90.

4 Variable Detection Rates

Each simulation in section 3 assigns the same fixed detection rate to all predators. Predators with detection rates of 50 or less could not create a viable population and predators with rates of 90 or above actually proved so successful at catching prey that they reduced the prey population to the point where the predator poulation was adversely affected. For detection rates from 60 to 80, the predator population increased (causing the corresponding prey population to decrease) as the detection rate increased. This section investigates what happens if individual predators are able to increase their detection rates during their lifetimes. This is to simulate real predators forming "search images" [1, 2].

In the simulations described in this section all predators are initially assigned a fixed starting detection rate. After that, each individual predator can increase its detection rate as it catches prey. Each time a predator successfully detects (and thus catches and consumes) prey, the detection rate for that

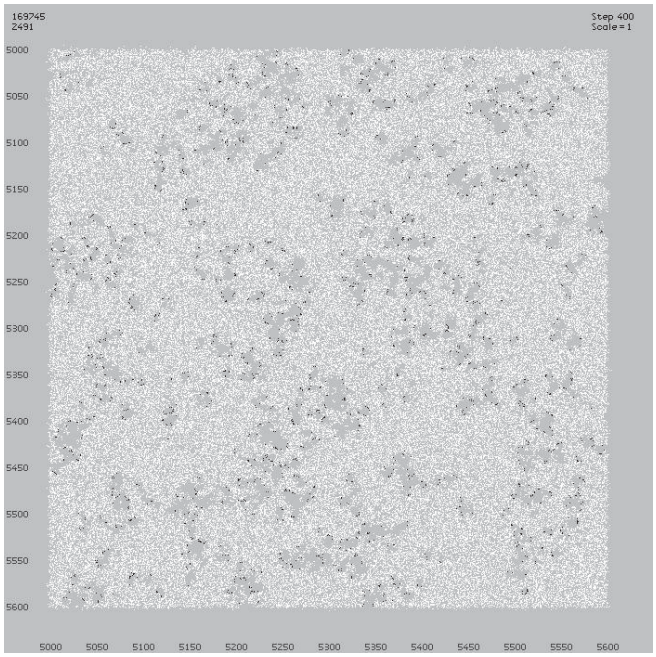


Figure 4: The situation at step 400 of a run in which all predators have a fixed detection rate of 60. This screen shot contains 2,491 predators (black) and 169,745 prey (white) animats. The clusters differ from those seen in Figure 1 where predators have a fixed detection rate of 90.

particular predator is increased by 5. For example, a predator that starts with a detection rate of 60 would increase the rate to 65 after catching one prey animat and then again increase the rate to 70 after catching the second prey animat and so on.

During these experiments, the prey remain unchanged. There are not different prey "species" or even prey with different colors or distinguishing characteristics. They are all exactly the same and no mutation effects are possible.

Figure 5 shows the predator populations for predators with changing detection rates. Each simulation has a different "starting rate" which is assigned to new predators. Each predator is then able to increase its own indivdual rate if it successfully detects and catches prey. When detection rates were fixed (as described in section 3) a fixed detection rate of 50 was not viable. In the simulations described in this section, a starting rate of 50 is viable because each predator is capable of increasing its own detection rate. The predator population with a starting rate of 50 is lower than the other populations but it is viable, i.e. does not become extinct during the simulation.

Once again, a detection rate of 90 leads to predators which are too successful, reducing the prey population and thus causing a corresponding drop in the predator population. In fact, in these simulations where each predator can increase its own detection rate considerably, the starting rates of 70, 80 and 90 all show evidence of the "boom-bust" effects that occur when

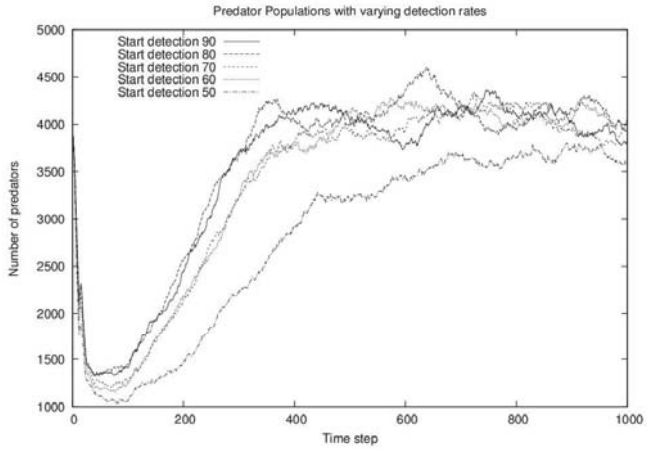


Figure 5: Plot showing predator populations for groups with changing detection rates. Each group has a different "start rate" for new predators. Higher detection start rates enable predators to catch prey more effectively. Once again, the highest detection rate leads to a slightly lower population than other groups.

predator populations become too large to be supported by the current prey population.

Figure 6 shows the corresponding prey populations for the same simulations. Various dips and overlaps in population numbers show instances when the predator detection rate became too high and the prey population was reduced by overpredation. The group in which predators have a starting rate of 50 is now viable because individual predators can increase their own detection rates above 50 – thus the average detection rate is above 50. This group shows the highest population of prey because these predators are less efficient at detecting prey.

5 The Prey Response

Section 3 analysed predators with a fixed detection rate and Section 4 outlined the results for predators with changing detection rates. Both sections included prey that was identical and unchanging.

This section investigates the effects of natural selection on the prey. Predators remain as in Section 4, that is they are able to individually increase their detection rates on successful detection of prey. However, the prey are now allowed to evolve and change. This simulates what has been observed in real prey populations [4, 6].

Prey are now assigned a "color value". This is an abstract notion of color and is simply an integer value from 0 to 18. The maximum number of colors could be set to any value and 18 was selected because it is difficult to distinguish more than 18 colors in a graphical display. New prey animats are

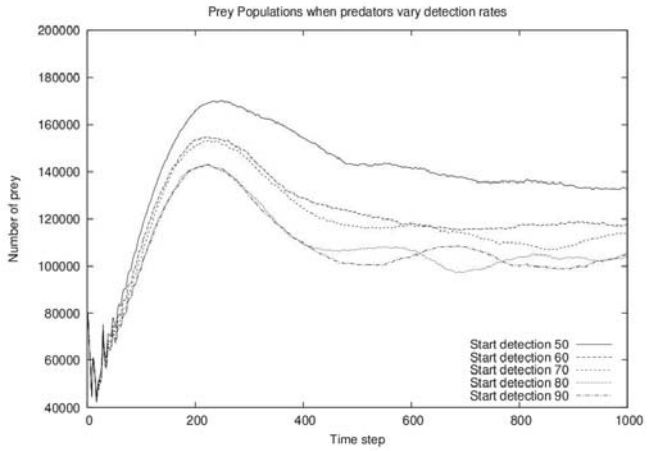


Figure 6: Plot showing prey populations when the predators have changing detection rates. Each group has a different "start rate" for new predators. Higher detection start rates enable predators to catch prey more effectively and the prey population is decreased accordingly. A start rate of 50 is now viable.

assigned a color value when they first appear according to the following algorithm:

```
r = random number from 1 to 100
if (r <= 2) then:
  d = random number from 1 to 2
  color = d + color of parent
else:
  color = color of parent
```

In other words, prey are usually the same color as the parent but have a 2% chance of being a different color. The new color value can differ from the parent color value by 1 or 2. This value remains unchanged for the lifetime of the prey animat. Both the chance of mutation (currently 2%) and the maximum difference from the parent color (currently set to 2) could be changed to other values.

The predator detection system is also changed to align with the color values of the prey. Each predator now maintains a "focus color". This is the color value of the prey that the predator is focussed on. The predator detection rate is increased by 5 only if the predator successfully detects prey such that the prey color value matches the predator focus value. If the predator detects prey with a color value that does not match the focus value – the predator detection rate is "reset" to the assigned starting detection value.

For example, assume that predator X currently has a focus color of 5 and a starting detection rate of 60. Then the following sequence of events could occur:

1. Predator X detects prey A with a color value of 12. Chance of successful detection is 60%.
2. Predator X focus color is set to 12 and detection rate increases to 65.
3. Predator X detects prey B also with a color value of 12. Chance of successful detection is 65%.
4. Predator X focus color remains unchanged and detection rate increases to 70.
5. Predator X detects prey C with a color value of 14. Chance of successful detection is 60% (the start rate).
6. Predator X focus color is set to 14 and detection rate is reset to the starting detection value of 60.

In summary, predators focus on prey of a particular color. As long as the predator detects only prey of that color, it will increase its detection rate. If the predator detects prey of a different color, the detection rate is reset to the starting rate and the predator switches its focus to prey of the new color.

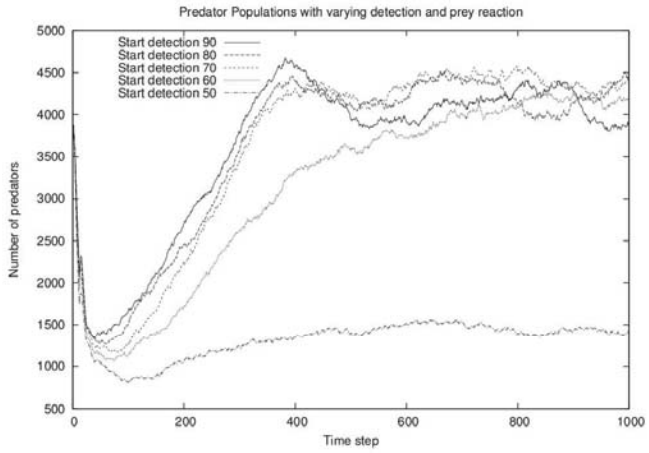


Figure 7: Plot showing predator populations when predators can increase their detection rates and prey can respond by mutating (change color) in order to reduce the chance of detection. Most groups show a drop due to initial over-predation when detection rates become too high.

Figure 7 shows the predator populations during these simulations. Each simulation has a different "starting rate" which is assigned to new predators. Each predator is then able to increase its own indivdual rate if it successfully detects and catches prey. Prey have a 2% chance of mutating to a different color. Note that prey do not change color during their lifetime – the effect described here happens only once ("at birth") or not at all.

Figure 8 shows the corresponding prey populations for the same experiments. Comparison with Figure 6 shows that the prey population benefits significantly from mutating and changing, i.e. from the process of apostatic selection.

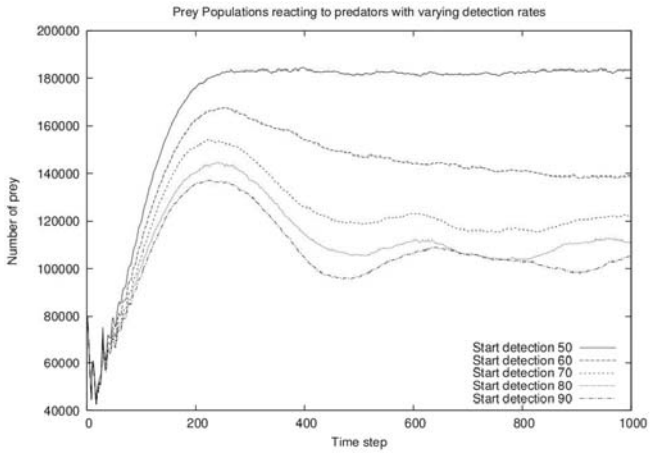


Figure 8: Plot showing prey populations when predators can increase their detection rates and prey can respond by mutating (change color) in order to reduce the chance of detection. Prey populations benefit from mutating – compare with Figure 6.

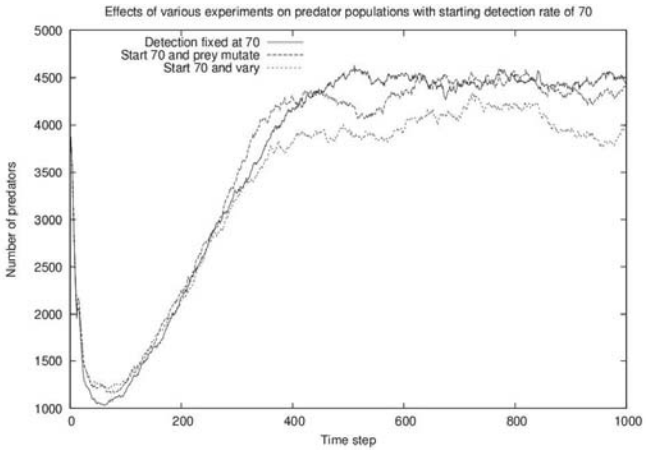


Figure 9: Predator populations all with a starting detection rate of 70. This plot compares the results of the three experiments: fixed detection rates, varying detection rates, and the prey response.

6 Conclusions and Future Work

This paper describes the results of several experiments that were based on a simple agent-based model that simulates the formation of "search images" by predators. The use of a virtual simulation enables a series of controlled experiments that would be impossible to duplicate with real predators. These include investigating the effects on predator and prey populations of: fixed detection rates; detection rates that vary for each individual; and the prey response to the predator search image.

The model has proven to be robust and able to support in excess of 150,000 agents for over 1,000 time-steps. The rule-based system effectively simulates the typical boom-bust oscillations of mathematical models of predator-prey systems.

The work described here is the first step towards a more accurate simulation of the effects of predators developing a search image and prey responding with apostatic selection. Future work includes fine tuning the values of numerous parameters including birth rates, death rates, chance of mutation, color difference between parents and offspring, etc.

Figure 9 shows the effects of the various experiments on the predator populations. In this example, populations are compared for a starting detection rate of 70. Similar comparisons could be made for any other starting detection rate. The most successful population is that with the fixed detection rate. However, this is also the least realistic scenario as it would be unusual for predators not to become skilled at detecting prey. The least successful population is when predators can increase the detection rate indefinitely with no corresponding prey mutation. While it can be assumed that this may be the best scenario for individual predators, it is the worst outcome for the predator population – probably due to over-predation and a reduce in the prey population. The best outcome is when the prey respond by mutating and changing color. This ensures that predator detection rates do not rise too high, thus ensuring a plentiful supply of prey, even if it means that predators have to re-learn a new search image when the most common prey changes color. It is interesting that the best result for prey (color mutation) is also the best (realistic) result for predators. What is best for the individual is not always best for the population as a whole.

As is common with agent-based systems, the results that emerge are not always what is expected. It is assumed that a high detection rate for predators is beneficial to them. While this may be true for individuals, an extremely high detection rate (95% or higher) is actually detrimental to the overall predator population because if predators are too successful they decimate the prey population. Ironically, the color mutation of prey, while obviously beneficial to individual prey, serves to reduce the average predator detection rate, thus assisting predators to overcome the problems of over-predation. It is anticipated that more accurate models will produce further interesting results in future.

References

[1] Tinbergen, L.: The natural control of insects on pinewoods. factors influencing the intensity of predation by songbirds. Arch. Neerl. Zool. **13** (1960) 265–343

[2] van Leeuwen, E., Jansen, V.: Evolutionary consequences of a search image. Theoretical Pop. Biology **77** (2010) 49–55

[3] Paulson, D.R.: Predator polymorphism and apostatic selection. Evolution **27** (1973) 269–277

[4] Bond, A.B.: The evolution of color polymorphism: Crypticity, searching images, and apostatic selection. Annu. Rev. Ecol. Evol. Syst. **38** (2007) 489–514

[5] Dukas, R., Kamil, A.: Limited attention: The constraint underlying search image. Behavioral Ecology **12** (2001) 192–199

[6] Dimitrova, M., Merilaita, S.: Prey concealment: visual background complexity and prey contrast distribution. Behavioral Ecology **21** (2010) 176–181

[7] Bond, A.B., Kamil, A.C.: Spatial heterogeneity, predator cognition, and the evolution of color polymorphism in virtual prey. Proceedings of the National Academy of Sciences of the USA **103** (2006) 3214–3219

[8] Adami, C.: On modeling life. In Brooks, R., Maes, P., eds.: Proc. Artificial Life IV, MIT Press (1994) 269–274

[9] Ray, T.: An approach to the synthesis of life. Artificial Life II, Santa Fe Institute Studies in the Sciences of Complexity **xi** (1991) 371–408

[10] Holland, J.H.: Echoing emergence: Objectives, rough definitions, and speculations for echo-class models. In Cowan, G.A., Pines, D., Meltzer, D., eds.: Complexity: Metaphors, Models and Reality. Addison-Wesley, Reading, MA (1994) 309–342

[11] Tyrrell, T., Mayhew, J.E.W.: Computer simulation of an animal environment. In Meyer, J.A., Wilson, S.W., eds.: From Animals to Animats, Proceedings of the First Int. Conf. on Simulation of Adaptive Behavior. (1991) 263–272

[12] Yaeger, L.: Computational genetice, physiology. metabolism, neural systems, learning, vision and behavior or polyworld: Life in a new context. In Langton, C., ed.: Proc Artificial Life III Conference. (1994)

[13] Wilson, S.W.: The animat path to AI. In Meyer, J.A., Wilson, S., eds.: From Animals to Animats 1: Proceedings of The First Int. Conf. on Simulation of Adaptive Behavior, Cambridge, MA: MIT Press/Bradford Books (1991) 15–21

[14] Scogings, C.J., Hawick, K.A.: An investigation into the effects of sentinels on animat collectives. In: Proc. Nineteenth IASTED Int. Conf. on Applied Simulation and Modelling (ASM 2011), Crete, Greece, IASTED (2011) 221–226

[15] Scogings, C.J., Hawick, K.A.: Modelling predator camouflage behaviour and tradeoffs in an agent-based animat model. In: Proc. IASTED Int. Conf. on Modelling and Simulation (MS'2013). Number 802-032, Banff, Canada, WorldComp (2013)

[16] Scogings, C.J., Hawick, K.A.: Simulating intelligent emergent behaviour amongst termites to repair breaches in nest walls. In: Proc. Int. Conf. on Computational Intelligence (CI2010), Maui, Hawaii, IASTED (2010) 125–132

[17] Scogings, C.J., Hawick, K.A.: Pack-hunting multi-agent animats. In: Proc. IASTED Int. Symp. on Artificial Intelligence and Applications, Innsbruck, Austria, IASTED (2009) 220–224

[18] Scogings, C.J., Hawick, K.A.: Global constraints and diffusion in a localised animat agent model. In: Proc. IASTED Int. Conf. on Applied Simulation and Modelling, Corfu, Greece, IASTED (2008) 14–19

[19] Hawick, K.A., Scogings, C.J., James, H.A.: Defensive spiral emergence in a predator-prey model. Complexity International **12** (2008) 1–10

[20] Lotka, A.J.: Elements of Physical Biology. Williams & Williams, Baltimore (1925)

[21] Scogings, C.J., Hawick, K.A., James, H.A.: Tools and techniques for optimisation of microscopic artificial life simulation models. In Nyongesa, H., ed.: Proceedings of the Sixth IASTED Int. Conf. on Modelling, Simulation, and Optimization, Gabarone, Botswana, IASTED (2006) 90–95

[22] Scogings, C.J., Hawick, K.A., James, H.A.: Tuning growth stability in an animat agent model. In: Proceedings of the 16th IASTED Int. Conf. in Applied Simulation and Modelling (ASM 2007), Palma de Mallorca, Spain, IASTED (2007) 312–317

[23] Scogings, C.J., Hawick, K.A.: Introducing a gestation period of time-delayed benefit into an animat-based artificial life model. In: Proc. 12th IASTED Int. Conf. on Artificial Intelligence and Applications (AIA'13), Innsbruck, Austria, IASTED (2013) 43–50

[24] Scogings, C.J., Hawick, K.A., James, H.A.: Boundary conditions and locality in an agent-based predator-prey model. In: Proc. 17th IASTED Int. Conf on Applied Simulation and Modelling, Corfu, Greece, IASTED (2008) 1–6

[25] Scogings, C.J., Hawick, K.A.: Altruism amongst spatial predator-prey animats. In Bullock, S., Noble, J., Watson, R., Bedau, M., eds.: Proc. 11th Int. Conf. on the Simulation and Synthesis of Living Systems (ALife XI), Winchester, UK, MIT Press (2008) 537–544

[26] Hawick, K.A., Scogings, C.J., James, H.A.: Spatial emergence of genotypical tribes in an animat simulation model. In Henderson, S.G., Biller, B., Hsieh, M.H., Tew, J.D., Barton, R.R., eds.: Proc. 2007 Winter Simulation Conference 2007 (WSC2007), Washington DC, USA (2007) 1216–1222

[27] Scogings, C.J., Hawick, K.A.: Multiphase updating - a practical approach to simulating animat agents. In: Proc. Int. Conf. on Artificial Intelligence (ICAI'12), Las Vegas, USA, WorldComp (2012) 1–7

[28] Volterra, V.: Variazioni e fluttuazioni del numero d'individui in specie animali conviventi. Mem. R. Accad. Naz. dei Lincei, Ser VI **2** (1926)

Computational Modelling and Simulation of Core-Periphery Terrorist Network

A.Kiruthiga[1] and Dr S.Bose[2]
Department of Computer Science and Engineering, College of Engineering Guindy,
Anna University, Chennai- 600 025,Tamilnadu,India.
[1]E-mail Address: kiruthiga312@gmail.com
[2]E-mail Address: sbs@cs.annauniv.edu

Abstract - *Terrorism and terrorist organizations are becoming more vulnerable nowadays. Understanding the terrorism and the organizational structures of terrorist networks are most important to devise successful defensive strategies against the terrorist activities. Terrorist organizational network is a real world complex system which is the association of numerous entities and has significant structural features . Core-Periphery terrorist network structure is one type of terrorist organizational structure which favours when the network members are very few to achieve the task. This paper presents Core-Periphery Terrorist Network (CoPTeN) model to synthetically simulate the core-periphery terrorist network structure based mainly on the preferential growth feature of complex networks. The structural characteristics of this model were analysed and discussed elaborately. As this model simulates the Core-Periphery terrorist network structures with imprecise inputs, it can help the anti-terrorism agencies by giving more insight into the structure of terrorist network to develop effective dismantling strategies against terrorist networks.*

Keywords: simulation, terrorist network, organizational structures, computational modelling, core-periphery.

1 Introduction

Networks are widely used to describe most natural phenomena and this can be defined as a collection of vertices and edges between vertices. These edges can be directed or undirected, and weighted or un-weighted. Entities in a social context and the relationship between them can be represented as a social network. Graphs are used to visually represent the social network. Complex networks are conceptualized as a set of entities intermingled in certain fashion and with significant characteristics. To construct these complex networks in order to analyse them, theory of networks is used predominantly. Violent acts which brings fear in to the people is called terrorism; nowadays terrorism becoming a serious threat to the human life. Terrorist networks are social networks with lot of secrecy about the connectivity and membership in the network.

Core-Periphery Terrorist Network (CoPTeN) structure is an existing configuration of terrorist organizational forms. Many economic and social networks also exhibit this core-periphery structure. Core-Periphery network structure is defined as alliance of two set of vertices namely core and peripheral, in which core is a small sized and heavily connected network and peripheral is big sized and loosely connected network[1]. Previously no terrorist network structures were taken into account to analyse and get an insight into it, in order to frame effective dismantling strategies. But terrorist organizations use the various network structures for better security and efficiency in operations.

In the following discussion, various works in simulation and analysis of core-periphery terrorist network structure are presented. Basic Core-Periphery network model was developed based on the characteristic definition of core and periphery nodes. Core network is formed by highly interconnected actors and periphery is formed by loosely interconnected actors [2]. Basic model of Core-Periphery [2] was enhanced by employing information measures which exhibits great degree of coherence and accuracy in developing core-periphery network[3].Core-Periphery model with randomness was developed in which probability values differ for connectivity between core nodes, peripheral nodes and between core-periphery nodes[4].

Measure to analyse the dichotomy of Core-Periphery structure was proposed and the findings suggested that the Core-Periphery structure of social networks was weak when compared to geographical networks[5].Studies in Core-Periphery structure revealed that this complex network lies in between several extreme properties like random/condensed structures and hierarchy/anti hierarchy structures. These high complex properties ensures the robustness of Core-Periphery structure[6].The basic assumptions reflected for this research work were; the actors in the network are dependent on each other and not autonomous when making decisions; the interactions that the edges denote are pathways for the flow of resources; the structure of the network affords opportunities or restrictions on the actions of the actors; the network represents lasting ties among the actors [5].

Based on the above literature investigation, the Core-Periphery structure exists in different complex network types like economic, transportation, geographical and social networks differs in their characteristics was observed. The

Core-Periphery structure involved in terrorist organizational structure was analysed in real time networks to find the core and periphery nodes respectively. In literature no model has been proposed to simulate Core-Periphery structure exists in terrorist networks and analysed its structural features. The objectives of this research work are to model Core-Periphery structured terrorist network and analyse its structural characteristics.

The remaining structure of the paper is arranged to explain the proposed framework of CoPTeN model. The following section presents the principal model with the use of the algorithm. The later section presents the detailed results of the proposed CoPTeN model and discusses the simulated Core-Periphery terrorist network results. Finally, this paper concludes by mentioning the future enhancements that could improvise the CoPTeN model.

2 CoPTeN Model

Terrorist networks organize themselves into a suitable organizational structure based on the secrecy/efficiency trade-off. Understanding of the dynamics of terrorist organizational structure helps to dismantle them effectively. When the secrecy is given higher priority the Core-Periphery structure is adopted by the terrorist organizations. Hezbollah terrorist group has adopted this core-periphery structure for its operations. Core-Periphery structure can also act as a sub network of a terrorist group in which hybrid more than one network structure is in practice. General patterns of core-periphery network structure witnessed in terrorist networks are listed below.

- Security is given importance more than efficiency [7]
- Core members are densely connected to one another and peripheral members are connected to the core. No connectivity exists in between the peripheral nodes [8]
- Generally the core-periphery structure evolves from multiple hub-spoke structure, which in turn evolved from scattered clusters.[9]
- Insulator members exists in between core and peripheral members, which distant the core member from rest of the network [10]

The following algorithm describes the simulation of Core-Periphery terrorist network simulation based on the general patterns listed above. In that the inputs taken are the number of nodes in the network (N), number of links to be imposed in the simulated network (L) and the node density of the core part of Core-Periphery network to be simulated(Cr).

This model basically constructs a random seed network with specific percentage of hub nodes out of core network nodes, then gives this seed to preferential growth procedure, in which the core network is constructed with preferential attachment and growth as the basic evolution stimulants. The generated core network is then analysed to find the potential leaders and insulators.These leaders are the nodes with maximum degree connectivity and the insulators are direct neighbors to these leaders and who can make the core leaders distant from the peripheral nodes and keep the core leaders secure. The procedure create_vertex(x) creates a vertex instance with 'x' as the index.

Algorithm : CoPTeN Model
Input : Network Size (N), Links(L) and Coreness Ratio (Cr)
Output : Core-Periphery Network Instance

Symbol – Definition
Hs - Hub strength
H - Hub Tuner
Es - Evolution steps
Lc -Target number of core links
Lp - Target number of peripheral links
Cn - Core nodes count
Pn - Peripheral nodes count
LpE - Links per Evolution
v - Instance of vertex
vc -Vertex count
p - Attachment probability

```
Begin
Let G(V,E) be an undirected graph where,
V(G) is a set of vertices & E(G) is a set of edges
H(G) is the set of hub vertices & I(G) is the set of insulator vertices
V(G)= {ø};E(G)= {ø};H(G)= {ø};I(G)= {ø}
H(G) ⊂ V(G) and I(G) ⊂ V(G) ∋ (I(G)∩ H(G))= {ø}
Cn=N*Cr & Pn=N-Cn
Lc=L-Hs+1 & Lp=L-Lc
Hs=Cn*(H)*(0.01) & Es=Cn-Hs
LpE=Lc/Cn
G=RandomGraph(Hs)
Vc=Hs
Repeat
  G=EvolvePreferentialGrowth(G,LpE,vc)
  vc ←vc+1
  Es←Es-1
Until Es>0
For counter 1 to Hs
  Choose v  (v∈V(G)&degree(v)=maxDegree(G)&v∉ H(G))
    H(G) ←H(G)U{v}
    N(v) ←N(v)U{u} if (v,u) ∈E(G)
  Choose u ∋ (u∈N(v) & degree(u)=maxDegree(N(v))
    I(G) ←I(G)U{u}
End for
Repeat
```

```
  u = create_vertex(pn)
  V(G) ←V(G)U{u}
  v = Random(I(G))
  E(G) ←E(G)U{(v,u)}
  Pn←Pn-1
  Until Pn>0
End
```

Procedure RandomGraph(n)

```
Begin
  For i=1 to n do
    v ←create_vertex(i)
    V(G) ←V(G)U{v}
  End for
  repeat
    Choose random vertex pair (v,u)
    Generate random number θ ∈ [0,1]
    If θ<p then
      E(G) ←E(G)U{(I,j)}
      n ← n-1
  Until n>1
End
```

Procedure EvolvePreferentialGrowth(G,LpE,vc)

```
Begin
  v ←create_vertex(vc)
  V(G) ←V(G)U{v}
  For edgecount=1 to LpE
    Repeat
      Choose vertex pair (v,u) ∋ (v≠u) & (v,u) ∉ E(G)
      Ψ= ( degree(u)+1) /((|E(G)|+|V(G))-1)
    Until (Ψ>p)
    E(G) ←E(G)U{(u,v)}
  End for
End
```

---

3 Results and Discussions

Simulation experiments were performed using the above algorithm for a variety of parameter constellations. Experimenting in any combination with N in the range of [30,1000], link density in the range of [0,1] inturn L in the range of thousands of links, Cr in the range of [0,1].Results of these simulation experiments are adjacency matrices of the constructed network instances. The sample simulated Core-Periphery network with N as 100 nodes,L as 990 and Cr as 0.6 is shown in Figure 1. The red, blue and black coloured nodes are leaders, insulators and member nodes respectively in Figure 1.

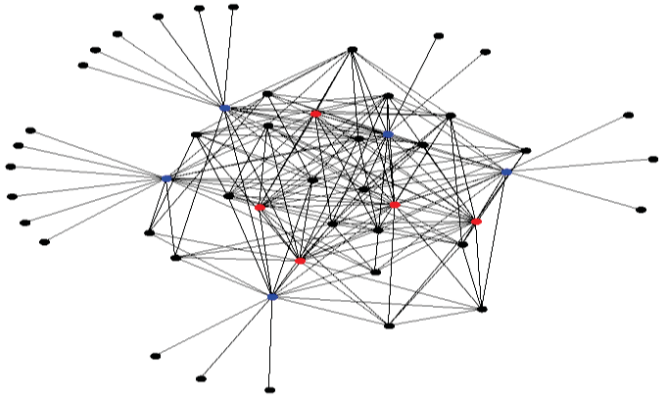


Figure 1. Sample Simulated Core-Periphery Network

3.1 Structural Characteristics of COPTEN Model

Dynamics of complex networks are generally analysed interms of the following structural measures. (a) Scale free degree distribution, (b) Small world and (c) Community Structure .

The following section exhibits the structural measures of CoPTeN model derived through extensive simulation experiments.

Scale Free Degree Distribution:

One of the common feature that exists among the real world complex networks is the presence of large percentage of nodes with very few links(degree) and very few nodes which has links to many other nodes in the network. This distribution follows power law degree distribution.Power law degree distribution of the undirected network is described as follows,

$$Pdeg(k) \propto k^{-\gamma} \qquad (1)$$

Where $\gamma>0$ is the power law exponent which is some constant and $\propto$ denotes asymptotically proportional to as $x \rightarrow \infty$.

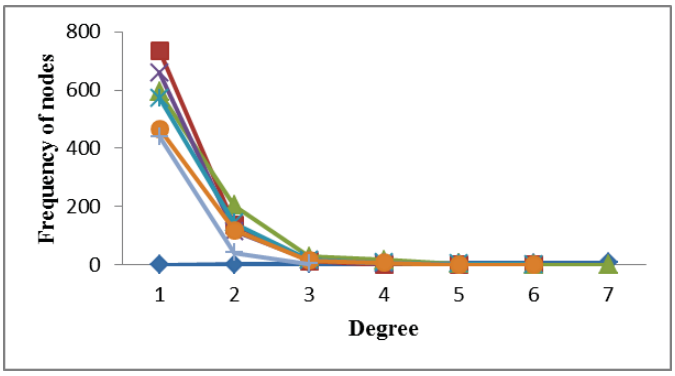


Figure 2. Sample Power Law Degree Distribution of CoPTeN Model

The Figure 2, clearly shows that the CoPTeN model follows power law degree distribution. CoPTeN model evole the network from a randomly generated seed network with

premeditated number of hubs to induce the leadership in the evolving network. This process is followed with a preferential growth mechanism, which will make these hubs as attractive nodes to get linked with the new nodes whichever coming into the network. This sequence of processes made the degree distribution to follow the power law inturn scale free degree distribution.The power law exponent is one way of endorsing the power law degree distribution of a complex network. Power law exponent of the CoPTeN model is given for various network sizes in Figure 3. Power law exponent of this proposed CoPTeN model falls within the maximum value of 3 frequently. It clearly shows that the CoPTeN model exhibits power law degree distribution.

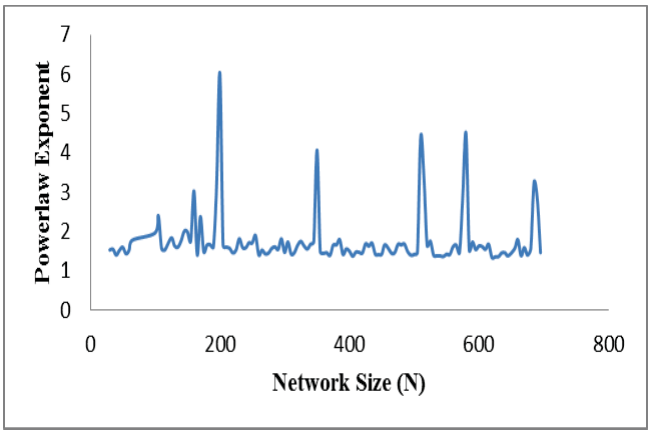


Figure 3. Sample Power Law Exponent of CoPTeN Model

Small World Property:

Diameter and Average path length are the measures to capture the small world property in complex networks. These terms describe the degrees of separation in the network. The longest path among the shortest paths from each node to every other node in the network is called as the diameter of the network.

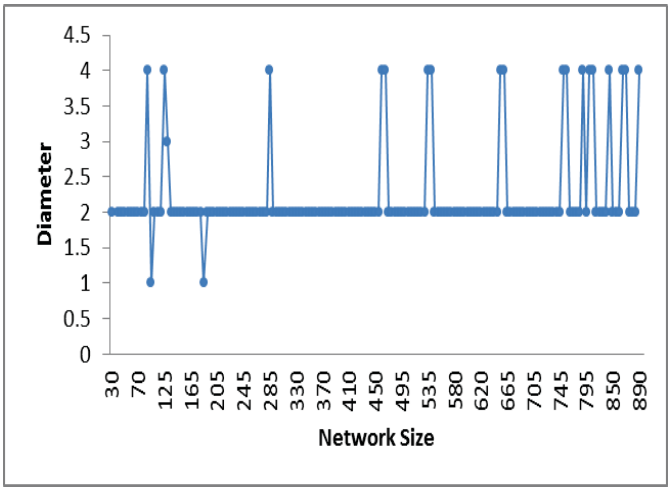


Figure 4. Sample Diameter of CoPTeN Model

The Figure 4, exhibits the sample diameter of the CoPTeN model and it is obvious that the diameter majorly presented as 2 and based on the random graph generated, in some cases it is going to the peak value of 4. Seed random graph has the ability to change the overall network structure, in case if the random seed is set as chain or path the diameter gets increased in the network. Average path length is the key measure for small world property. For distinct node the average path length tells how adjacent the individual to all of the others in a network, this individual can reach others quickly without going through too many intermediaries.

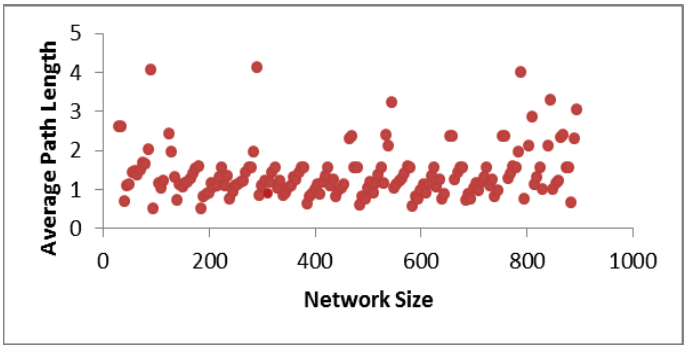


Figure 5. Sample Average Path Length of CoPTeN Model

The Figure 5 , shows the average path length of the CoPTeN model for various network sizes. Small average path length is an indicator of the small world property.CoPTeN model allows any node in the network, even the peripheral nodes to reach their leader in single hop, consequently the average path length becomes small and exhibits small world property. The following equation(2), shows the relationship between the average path length and the network parameters respectively network size and coresize. Coresize is the number of nodes in the core part of the core-periphery network generated through the CoPTeN model.

$$\text{Average Path Length} = \log_{coresize} networksize + 1 \qquad (2)$$

Community Structure:

A community is usually considered to be a set of nodes where each node is closer to the other nodes within community than to nodes outside it. Widely used measure for community structure is clustering coefficient; it measures the clumpiness of a graph and has relatively high values in many graphs.

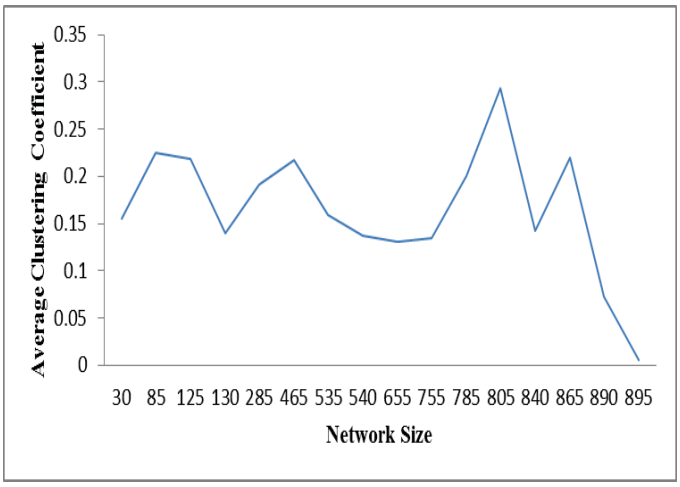


Figure 6. Sample Average Clustering Coefficient of CoPTeN Model

The Figure 6, shows the average clustering coefficient of the proposed model, it shows that the CoPTeN model clusters well the nodes in the network as there is an increase in the network size and number of links, the clustering coefficient increases. This demonstrates the existence of community effects in the CoPTeN model.

3.2 Simulations Results

The characteristics of any simulation model in complex systems are best understood when it is compared with the pioneer benchmark network evolution models Erdos-Renyi(E-R) model and Barabasi-Albert(B-A) model[11,12]. The following Table I, compares the basic network measures of the proposed CoPTeN model with Erdos-Renyi and Barabasi-Albert model.

Table I. Simulation Results of Network Models

Model	Network Size (N)	Links Count (L)	Diameter	Average Path Length	Average Betweeness Centrality	Average Eigenvector Centrality	Average Closeness Centrality
CoPTeN Model	125	1716	4	2.40	089.936	0.0034	0.0080
	285	10760	4	1.95	136.144	0.0018	0.0035
	355	15673	4	2.05	219.187	0.0012	0.0028
	465	26502	4	2.31	304.763	0.0009	0.0022
	535	23674	4	2.40	375.362	0.0008	0.0019
	655	29032	4	2.36	447.302	0.0007	0.0015
	755	38570	4	2.36	514.885	0.0006	0.0013
Erdos-Renyi Model	125	1716	3	1.76	047.512	0.0045	0.0070
	285	10760	2	1.73	103.502	0.0020	0.0024
	355	15673	2	1.74	138.501	0.0019	0.0020
	465	26502	2	1.75	174.279	0.0013	0.0018
	535	23674	2	1.83	221.895	0.0010	0.0013
	655	29032	3	1.86	281.792	0.0008	0.0010
	755	38570	2	1.86	325.052	0.0007	0.0010
Barabasi-Albert Model	125	1716	3	1.81	050.336	0.0044	0.0180
	285	10760	3	1.75	106.905	0.0020	0.0095
	355	15673	3	1.76	140.021	0.0016	0.0058
	465	26502	2	1.77	178.912	0.0012	0.0042
	535	23674	3	1.84	224.286	0.0010	0.0038
	655	29032	3	1.87	284.160	0.0007	0.0025
	755	38570	3	1.87	327.254	0.0007	0.0021

From the above Table I, it is clear that the diameter and average path length of the CoPTeN model is high when compared to the E-R and B-A models of comparable sizes. The reason behind this increased diameter and average path length is the deployment of insulator node inbetween the peripheral and core nodes, to make the core leaders distant. Average betweeness centrality of CoPTeN model is high when compared to E-R model and B-A model of comparable sizes, it shows the dominance of hubs i.e. core leaders and insulators in the CoPTeN model, as the betweeness centrality is a measure of the extent to which the nodes being a point of communication in the network.

E-R & B-A model possess the high eigen vector centrality when compared to the CoPTeN model as shown in the above table. As there are few hubs which are the base for making well connected communities in the B-A model, the eigen vector centrality is high when compared to the CoPTeN model. Where as in CoPTeN model to make the core leaders distant from the peripheral nodes, insulators are used, hence the eigen vector centrality is low when compared to more reachable B-A and E-R model.Average closeness centrality is the invese of distance measure, which inturn depicts how close a node is from all other nodes in the network. B-A model generates scale free graph with notable number of hubs, hence

it can reach every node from every other node, the closeness centrality of this model is high when compared to the CoPTeN model. The E-R model posses low closeness centrality when compared to the CoPTeN model as there are no significant hubs (leaders) in it.

4 Conclusion

The model presented in this paper simulates the Core-Periphery organizational structure of terrorist organizations based on the building block of Core-Periphery terrorist network structure found from broad literature and based on the properties of complex networks. The structural properties and measures of the proposed model is derived on top of extensive simulation experiments and the results confirmed the proposed model synthetically generates the core-periphery structure of terrorist networks with significant features. The proposed model simulates the Core-Periphery network based on the random seed graph solely nonetheless the role of the seed in altering the structural properties of the simulated network. The significance of random seed graph in this proposed model can be analysed further in detail.

5 References

[1] Hojman, D. A., & Szeidl, A. (2008). Core and periphery in networks. *Journal of Economic Theory*, *139*(1), 295-309.

[2] Borgatti, S. P., & Everett, M. G. (2000). Models of core/periphery structures.*Social networks*, *21*(4), 375-395.

[3] Muñiz, A. S. G., & Carvajal, C. R. (2006). Core/periphery structure models: An alternative methodological proposal. *Social networks*, *28*(4), 442-448.

[4] Rombach, M. P., Porter, M. A., Fowler, J. H., & Mucha, P. J. (2014). Core-periphery structure in networks. *SIAM Journal on Applied mathematics*, *74*(1), 167-190.

[5] Holme, P. (2005). Core-periphery organization of complex networks. *Physical Review E*, *72*(4), 046111.

[6] Csermely, P., London, A., Wu, L. Y., & Uzzi, B. (2013). Structure and dynamics of core/periphery networks. *Journal of Complex Networks*, *1*(2), 93-123.

[7] Morselli, C., Giguère, C., & Petit, K. (2007). The efficiency/security trade-off in criminal networks. *Social Networks*, *29*(1), 143-153.

[8] S. P. Borgatti, M. G. Everett. (2000), Soc. Networks 21, 375.

[9] Ronfeldt, D. (2005).Al Qaeda and its affiliates: A global tribe waging segmental warfare?.First Monday.

[10] Phil Williams, "Drug Trafficking Networks," Current History (April 1998) pp. 154–159.

[11] Erdos, P., & Rényi, A. (1961). On the evolution of random graphs. Bull. Inst. Internat. Statist, 38(4), 343-347.

[12] Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks.*science*, *286*(5439), 509-512.

Analysis of Simulation Modeling Systems Illustrated with the Problem of Model Design for Technological Logistics

K. Aksyonov[1], **E. Bykov**[1], **O. Aksyonova**[1], **N. Goncharova**[1], **and A. Nevolina**[1]
[1]Department of Information Technology, Ural Federal University, Ekaterinburg, Russian Federation

Abstract - *The paper presents definition of the models for technological logistics in various modeling systems: Plant Simulation, Simio, AnyLogic, as well as the software, which is currently run at an enterprise that operates metallurgical production with the purpose of quality assurance. Performance analysis of simulation systems identified the advantage of current software from the point of view of RAM and CPU load, and Simio with Plant Simulation from the experiment duration point. Main disadvantages of the systems under analysis include incomplete correspondence to the problem domain of the resource conversion processes, orientation towards users with programming skills and the corporate information system integration interfaces.*

Keywords: Simulation modeling systems; technological logistics processes; metallurgical production; agent-based modeling

1 Introduction

Current ideas of enterprise activity rely on process-based approach. According to it, activity consists of a structured network of processes that consume resources and produce the output. Quality forecasting for the manufactured products, as well as identification of quality degradation incidents, are among the most pressing tasks for metallurgical industry. Simulation modeling is widely used at present for forecasting and optimization of technological, industrial and logistical processes at enterprises of various spheres [2-4]. For this paper we discuss a sample model of metallurgical production for modeling of technological logistics that covers convertor production, hot- and cold-rolled mill products. We are comparing simulation modeling tools from the point of view of this process representation and convenience of model design for the technologist without special programming skills.

Metallurgical manufacture model will be used for analysis of three processes: 1. Technology of production unit processing, 2. Logistical process of production unit load into cars, 3. Management process of man-loading during product load. The model requires definition of the roles of crane and sling operators and their link with the operation of "Loading product unit into the car". Operations need to have corresponding utilized tools set, including various devices, a crane, a train set. The model would show technologist's activity, whose goal is to 1. Direct different order types to corresponding routes, 2. Direct condemned product units for additional treatment. The model will simulate production of pre-defined number of product units (varying from 10 to 10.000).

2 Choosing analyzed modeling tools

An enterprise uses its own quality assurance software, which includes the enterprise process definition module for design of simulation models of processes under research, and process optimization module for experimenting with the models and searching for management decisions. These two modules are based on multi-agent modeling and the concept of big data. Authors compare the metallurgical manufacturing process model definition with the enterprise software and the popular simulation tools Plant Simulation [6], Simio [5], and AnyLogic [3]. Model definition in AnyLogic can be found in [2] and is not presented in this work.

3 Model definition with Plant Simulation system

The following objects were used for model design.

Source/Drain objects were used for generation and destruction of model entities (transacts, product units). Plant Simulation provides several types of distributions when defining entity generation: uniform, exponential, etc.

SingleProc objects are used for definition of resource production and entity seizure by an object for the specified time (e.g. loading of the car requires processing of one entity "Empty car" and production of resource "Product units, loaded into cars"). ParallelProc objects are used for definition of multiple entity seizure by a single object (e.g. for all technology operations). Work with entity attributes is done in program code on object-oriented language SimTalk in object processing methods of SingleProc and ParallelProc. WorkerPool object is used for definition of labor resources in form of a table. Broker object was used to spread the workers among the operations in the code. FlowControl object is used for probabilistic entities distribution between SingleProc and ParallelProc objects. Also this object may be used for process

synchronization, to make it possible to dispatch entities only when all receiving channels are available. PS system uses object-oriented approach for model design, with aid of SimTalk language.

4 Model definition with Simio

A model within Simio is defined in form of the objects (models of objects). For the metallurgical production model we used standard Server objects for processes, which were linked by uni- and bi- directional relations, including Paths, Timed paths and Connectors, that define process chronology and entity flow direction. Entities are generated by the Source object and destroyed by the Sink object. Simio is a simulation system, oriented to objects, not processes. Objects within the model may represent product units, aggregates, workers, transportation vehicles, etc. Objects are built graphically based on object-oriented approach. Design of metallurgical manufacture model was based on equivalency principle, according to which models and objects are not distinguished and may be used on various complexity levels of process definition.

Each block of a Simio model has its inputs and outputs. Probabilistic type of object flow direction is set with path weights (Figure 1). We used the Combiner object to simulate loading of cars. Every 24 hours it generates a 14-car train set. Loop between car generator, warehouse and car sink resulted in introduction of an agent, distributing entities between these operations.

Part Generation | Vehicle Timing

	Generated Part	Part Weight	Initial Destination	Destination After Etching
▸ 1	TypeA	15	Input@Melting1	Input@FiveKletStan
2	TypeB	10	Input@Melting2	Input@FourKletStan
3	TypeC	75	Input@Melting3	Input@FiveKletStan

Figure 1. Setting path weights in Simio

5 Metallurgical model definition with current enterprise software

Models are defined in the framework of multi-agent resource conversion processes, with visual notation and production rules "If-Then" [1]. Graphical notation is applied to set the nodes (operations and agents) and relations between them. Production rules – to generate agents knowledge base. Knowledge base contains definition of the situations that the agent may diagnose, as well as agent reaction to the situation.

Model may be based on resources or on entities. Two models were designed to analyze the processes: resource-based and entity-based. Resource-based model assumes that each node is defined in form of setting resource consumption of each input and resource production on the output. Here inputs and corresponding outputs are separated by the specified delay. Resources in this model represent the queues for each node that increase on the output of the previous node and decrease on the input of the next one.

Entity-based model assumes that node behavior is defined in form of entity seizure on the input for the specified time. The entity is then transferred to the next node on the output. Entities in this model represent product units. Each entity has the predefined set of system attributes and a set of user attributes, different for each problem domain. For the current problem three attributes were introduced: entity type, unfinished indicator, waste ratio. Agents were used to define behavior of "Technologists".

6 Comparison of current enterprise software with simulation tools

We assume that models are equivalent and produce averagely the same output. As an effectiveness criterion we use the duration of experiment on the same hardware with animation set to off (Figure 2).

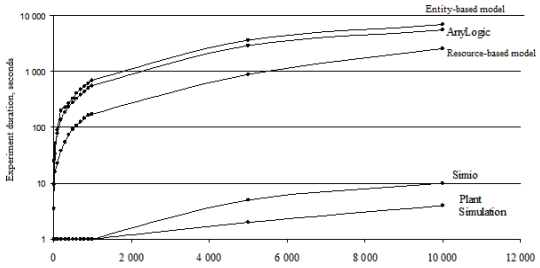


Figure 3. Experiment duration and number of processed product units

The analyzed simulation modeling tools may be separated into fast tools (Simio, PS) and slow tools (AnyLogic and enterprise process optimization module). Optimization module speed is related to detailed journaling of log tables and statistics on model variables and entity instances. No other simulation tool provides this statistics. Resource-based model works faster in the same simulation system. This fact is related to computational resources being spent on queues processing in the process optimization module. The slowest experiment duration was 2 hours and 13 minutes, which may be applicable in case of non-real-time decision making.

After analyzing simulation results we may conclude:

1. All models are adequate to logistical processes of a metallurgical enterprise,
2. Simulation speed is applicable for all simulation systems for various production volumes,
3. Simio and PS have an advantage in simulation speed for simulation of logistical processes of an enterprise,
4. CPU and RAM load are applicable for a short (under 10 minutes) simulation experiment without animation in all systems,
5. CPU and RAM load are applicable for a long non-animation experiment (over 1 hour) for all systems,

except PS (due to hang up) and Simio (due to high RAM load),

6. Advantage of the enterprise optimization module from the RAM load point of view for "short" and "long" non-animated experiments.

7 Running experiments

Work presents analysis of functionality of enterprise modeling and optimization modules and PS, AnyLogic, Simio simulation systems. The implemented model represented operation of metallurgical manufacture. Experiment results are presented in Table 1.

All analyzed systems do not provide comprehensive functionality for modeling of technological logistics. Among the drawbacks of most systems we may name the following:

1. Missing functionality:

№	Parameter	P S	Simio	AnyLogic	Current
1	**Resource conversion processes definition language**				
1.1	- Definition of resources, tools, convertors	☑ / ☑ / ☑	☑ / ☑ / ☑	☑ / ☑ / ☑	☑ / ☑ / ☑
1.1.1	- vector of inputs / outputs / tools	☑ / ☑ / ☐	☑ / ☑ / ☑	☑ / ☑ / ☑	☑ / ☑ / ☑
1.1.2	- amount of tools: constant / function / random	☑ / ☐ / ☐	☑ / ☑ / ☑	☑ / ☑ / ☑	☑ / ☑ / ☑
2	**Support for queuing systems**				
2.1	- Generating and deleting transacts / Processing transact attributes	☑ / ☑	☑ / ☑	☑ / ☑	☑ / ☑
2.2	- Support for parallel channels	☑	☑	☑	☐
3	**Building the multi-agent model**				
3.1	- Agent element	☐	☑	☑	☑
3.2	- Agent behavior models (state charts, activity, production)		☐ / ☐ / ☑	☑ / ☑ / ☐	☑ / ☐/ ☑
3.3	- Message exchange language		☑	☑	☑
4	**Simulation modeling**				
4.1	- Discrete event	☑	☑	☑	☑
4.2	- System dynamics models	☐	☐	☑	☐
5	**Expert modeling**				
5.1	- Knowledge representation model	☐	☐	☐	Production
5.2	- Logical output machine				☑
6	**Running experiments**				
6.1	- Generating experiment plan	☑	☑	☑	☑
6.2	- Automatic launch of experiment plan	☑	☑	☑	☐
7	**Analysis of experiment results**				
7.1	- Reporting on transact and tool processing	☑ / ☑	☑ / ☑	☑ / ☑	☑ / ☑
7.2	- Reporting on avoidance of bottlenecks	☐	☐	☐	☐
7.3	- Instance-based statistical logging of transacts	☐	☐	☐	☑
8	**Working with model database**				
8.1	- Import of initial data from external sources (text file / spread sheet / database)	☑ / ☑ / ☑	☑ / ☐ / ☐	☑ / ☑ / ☑	☐ / ☐ / ☑
8.2	- Export of experiment results (text file / spread sheet / database)	☑ / ☑ / ☑	☑ / ☐ / ☐	☑ / ☑ / ☑	☐ / ☑ / ☑
8.3	- Open access to model database	☐	☐	☑	☑
9	**Integration with the corporate information system**				
9.1	- Programming support (application language / Java)	☑ / ☐	☑ / ☐	☐ / ☑	☐ / ☐
9.2	- Integrated software assistants	☐	☐	☑	☑
10	**Interface convenience**				
10.1	Model design interface (programmer's interface / problem domain expert's interface)	☑ / ☐	☐ / ☑	☑ / ☐	☐ / ☑

Table 1. Comparison of features of simulation modeling systems and current enterprise software

a. System goals definitions,

b. Expert modeling support (except modeling and optimization modules),

c. Reporting with recommendations to remove bottlenecks,

d. Localizability (except Simio, modeling and optimization modules),

2. Low integration with corporate information systems,

3. Programming user orientation (except Simio, modeling and optimization modules).

8 Conclusion

Thus we conclude of applicable values for simulation speed for all simulation systems. Best results based on simulation speed are provided by PS and Simio.

Used notation of PS systems does not fully correspond to the problem domain of multi-agent resource conversion processes, since this system does not provide agent definition and setting of the tools vector for a resource conversion process. Another disadvantage is orientation for a programming user. An important advantage is integration with the corporate information system.

Simio notation corresponds to the problem domain of resource conversion processes. One disadvantage is high retail cost, which results in low popularity in Russia.

AnyLogic notation corresponds to the problem domain of multi-agent resource conversion processes. Unfortunately, it does not support expert modeling methods, and the interface requires programming skills from the user.

Looking at the convenience of defining an interface for the processes of technological logistics, definition of multi-agent resource conversion process elements, including resources and tools, the modeling module provides the best service. The modeling module requires development in the direction of system integration with the corporate information systems.

9 Acknowledgments

Research is conducted under the terms of contract № 02.G25.31.0055 (project 2012-218-03-167) with financial support from the Ministry of Education and Science of the Russian Federation.

10 References

[1] Aksyonov, K.A. Theory and practice of decision support tools, Germany, Saarbrucken: LAP LAMBERT Academic Publishing GmbH & Co. KG. ISBN 978-3-8465-0782-7, 2011

[2] Aksyonov K. A., Antonova A. S., Kiselyova M. V. Modeling metallurgical products in AnyLogic and BPsim.MAS, Proceedings of 6th science conference “Simulation. Theory and Practice” IMMOD 2013, volume 2, Kazan, Russia, 2013

[3] Karpov Y. G. Simulation of systems. Introduction to simulation with AnyLogic 5. BHV, St. Petersburg, Russia, 2005

[4] Osaba E, Carballedo R and Diaz F. Simulation tool based on a memetic algorithm to solve a real instance of a dynamic TSP // Proceedings of the IASTED International Conference Applied Simulation and Modelling. Napoli, Italy, 2012

[5] Simio Overview. URL: http://www.simio.com/index.html

[6] Tecnomatix Plant Design and Optimization: Plant Simulation. URL:

[7] http://www.plm.automation.siemens.com/en_us/products /tecnomatix/plant_design/index.shtml

Tactical Data Link Simulator with Cognitive Anti-Jamming Capability

S. Choe, E. Ko, and S. Jeon
[1] Dept. of Inform., Commun., and Electron. Engineering, The Catholic Univ. of Korea, Bucheon-si, Korea

Abstract - *In this paper, we present a Link-16-based tactical data link (TDL) simulator with cognitive anti-jamming capability (CAJC). CAJC improves anti-jamming capability of existing TDL by adding the cognitive radio (CR) spectrum sensing-based jammer detection. In the presented TDL, each terminal that senses (or detects) the jammer signal can avoid the jammer spectrum opportunistically. For jammer signal detection, each terminal employs generic CR spectrum sensing schemes like signal energy sensing (SE) and signal-to-jammer-plus-noise ratio sensing (SJNR) or hybrid CR spectrum sensing schemes like beacon plus signal strength sensing (BC_SE) and beacon plus signal-to-jammer-plus-noise ratio sensing (BC_SJNR). In this paper, we design a Matlab/Simulink-based simulator for the verification of presented TDL. This simulator includes a cyclic code shift keying (CCSK) data symbol modulator, a minimum shift keying (MSK) chip symbol modulator, a Rician fading channel module, and a partial-band noise jamming (PBNJ) module. Using this simulator, we can evaluate the bit-error-rate (BER) performance of the proposed TDL and verify that the proposed TDL is more robust to conventional TDL over PBNJ environments.*

Keywords: Tactical Data Link; Partial-Band Noise Jamming; Cognitive Radio; Spectrum Sensing

1 Introduction

Link-16, that is a TDL currently installed and operating in the U.S. Air Force, is a centric weapon system for the joint tactical information distribution system (JTIDS) [1]. Link-16, an ad-hoc-based local area network that consists of several (or many) sky or ground terminals, is typically used for the C4I information transfer from these terminals to air platform, or vice versa [2].

In this paper, we propose a Link-16-based TDL with the cognitive anti-jamming capability (CAJC) that senses the jamming signal and opportunistically avoids the jammer spectrum for secure data transmission. CAJC combines conventional frequency hopping (FH)-based anti-jamming function with cognitive radio (CR) spectrum sensing (SS)-based jammer detection function. For jammer signal detection, the presented TDL implements generic SS schemes like signal energy sensing (SE) and signal-to-jammer-plus-noise ratio sensing (SJNR) or hybrid SS schemes like beacon plus signal strength sensing (BC_SE) and beacon plus signal-to-jammer-plus-noise ratio sensing (BC_SJNR); that is, in hybrid SS, beacon detection (BC) is additionally used for the jammer signal sensing.

In this paper, we design a Link-16-based TDL simulator using Matlab/Simulink. This simulator includes a cyclic code shift keying (CCSK) modulator for baseband modulation, a minimum shift keying (MSK) modulator for chip spreading and modulation, a partial-band noise jammer (PBNJ) module, a time synchronization block, a bit-error-rate (BER) check block, and a channel module implementing additive white Gaussian noise (AWGN) plus Doppler-shifted Rician fading [3]. In that channel module, we can set the Rician fading channel parameters including *k* factor and maximum Doppler frequency.

Especially, the designed simulator also has a cognitive anti-jamming block including CR spectrum sensing (SS)-based jammer detection and jamming channel status information (JCSI) generation functions. In that block, we can set a threshold value satisfying given jammer signal miss-detection & false alarm probabilities. In this paper, the false alarm probability is assumed to be almost zero [4]. Via simulation, we evaluate the proposed TDL system performance over PBNJ environments and verify its superiority to the conventional TDL [5-6].

The rest of this paper is organized as follows: Section II explains the proposed TDL system model including TDL simulator design. Section III addresses cognitive anti-

jamming operation and jammer signal detection, and Section IV presents the simulation results of the proposed TDL over PBNJ conditions. Finally, concluding remarks are given in Section V.

2 TDL System Model and Simulator Design

TDL system requests reliable real-time data transmission over severe tactical channel conditions including jamming and fading [5].

Link-16, a time division multiple access (TDMA)-based TDL system, allocates a time slot channel with the period of 1/128 sec to its terminals (users) with a transmit packet. A Link-16 terminal operates 51 FH frequencies within the signal bandwidth of 3 MHz and randomly hops to one of those FH frequencies every 13 μsec [6]. Fig. 1 shows a block diagram of the presented Link-16-based TDL with CAJC. This TDL consists of a random data signal generator, a CCSK module for base-band modulation and demodulation, a MSK/FH module for chip signal modulation and FH spreading, a time synchronization block, a bit-error-rate (BER) check block, and a channel module for the AWGN, Rician fading, partial band noise jamming (PBNJ) channel setting. The proposed TDL also includes a cognitive anti-jamming block (the shaded one in Fig. 1) that can detect (sense) the jammer signal and opportunistically avoid the jammer spectrum. We assume that for anti-jamming operation, every TDL terminal has a frequency hopping table, termed 't-Hop-Frequency', and randomly hops to one of those hopping frequencies in that table every hop time. For simulator simplicity, we omit channel encoder/decoder.

The presented TDL can simulate the partial-band noise jammer using a jamming sub-block which is included in the channel module (see Fig. 1). In that jamming sub-block, we can set the jamming spectrum ratio ρ (= jamming bandwidth / total signal bandwidth (= 153MHz)) and $E_b/(N_o + N_j)$ (= bit energy per noise plus jammer spectral density). In simulation, we assume that no-jamming channel has E_b/N_o = 20dB.

Fig. 2 shows the block diagram of proposed TDL simulator with CAJC. For CAJC, the presented simulator implements a cognitive anti-jamming circuit (see a broken-line box in Fig. 2). This circuit combines conventional frequency hopping (FH)-based anti-jamming function with cognitive radio (CR) spectrum sensing (SS)-based jammer detection function such that TDL anti-jamming capability is improved. For practicality, we consider the imperfect jammer spectrum sensing with non-zero miss-detection probability, i.e., $P_{md} \neq 0$ (but we assume almost zero false alarm probability for the improved system robustness to channel noise conditions). As seen in Fig. 2, the two input signals that are generated by two independent binary random (Bernoulli-distributed) signal generators (one for correctly-jamming-avoided signal and the other one for incorrectly-jamming-avoided signal) pass through each different path (one for jamming signal path and the other one for no-jamming signal path). For a certain P_{md}, we can set a corresponding threshold value in terms of E_b/N_j.

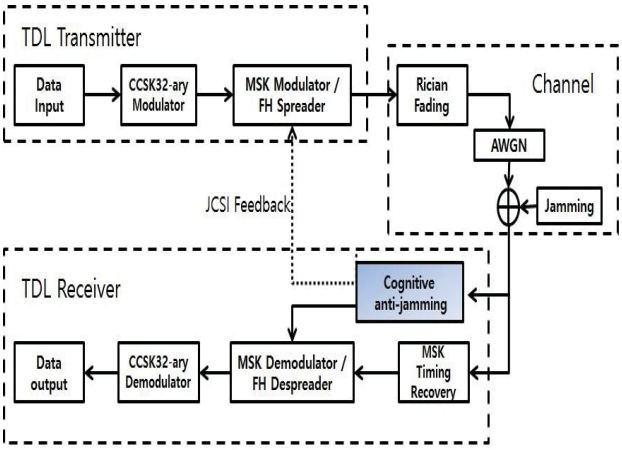


Fig. 1. Block diagram of the proposed TDL system.

3 Cognitive Anti-Jamming Operation and Jammer Signal Detection

3.1 Cognitive anti-jamming

In the presented TDL, each terminal can choose one of two FH operating modes: fixed operation mode and adaptive operation mode (or cognitive anti-jamming operation mode). While a terminal at the fixed operation mode performs conventional FH operation where a fixed frequency table 't-Hop-Frequency' is used, a terminal at the adaptive operation mode executes cognitive FH operation where a varying table 't-Hop-Frequency' depending on the jamming channel conditions is used.

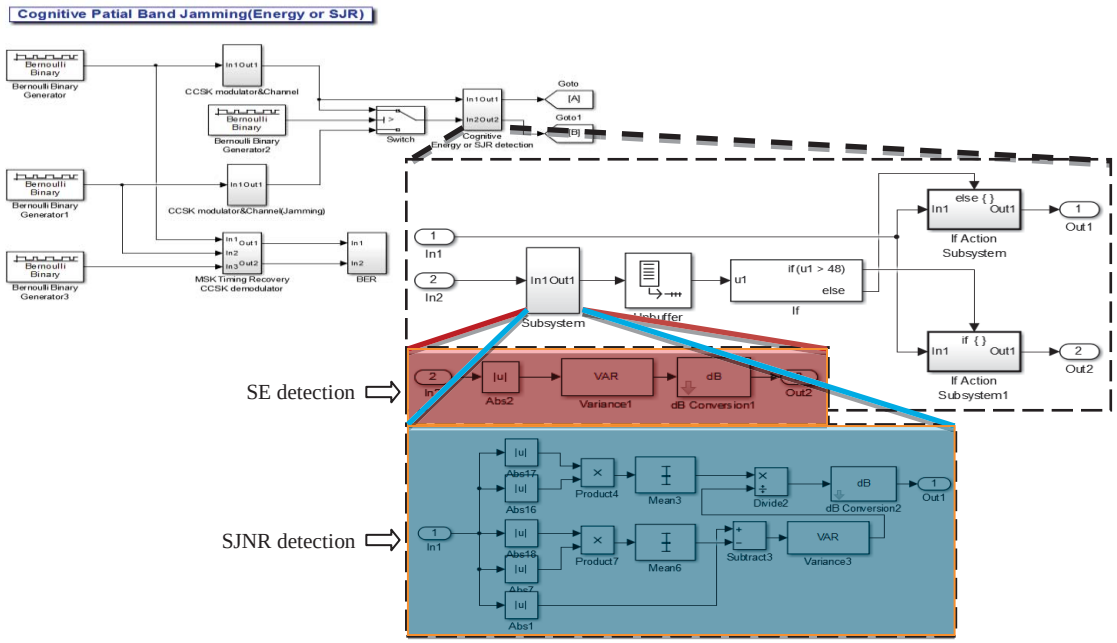


Fig. 2. TDL simulator using Matlab/Simulink (SE detection: red-coloured box / SJNR detection : blue-coloured box).

Fig. 3 shows a flow chart of the cognitive anti-jamming operation mode of the presented TDL. First, once a jammer signal is detected, the receiver terminal generates (updates) the jamming channel status information (JCSI) which includes a list of jammed FH channels and signal-to-jammer-plus-noise ratio (SJNR) per channel and feedbacks JCSI to the transmitter terminal. Then, the transmitter terminal sends ACK back to the receiver and as soon as ACK is received, the receiver terminal uses the updated t-Hop-frequency for FH operation. Hence, the presented TDL system can effectively and timely avoid the jammer spectrum for reliable data transmission.

3.2 Jammer Signal Detection

In the designed simulator, we choose either SS-based jammer signal detection techniques such as SE or SJNR or hybrid SS-based jammer signal detection techniques. In hybrid techniques, we use both beacon detection and SS-based detection such that the detection reliability is improved. Assume that the proposed TDL is able to scan the entire frequency spectrum by hopping all the allocated FH channels in equal probability matter. Below explains each of jammer signal detections in further detail.

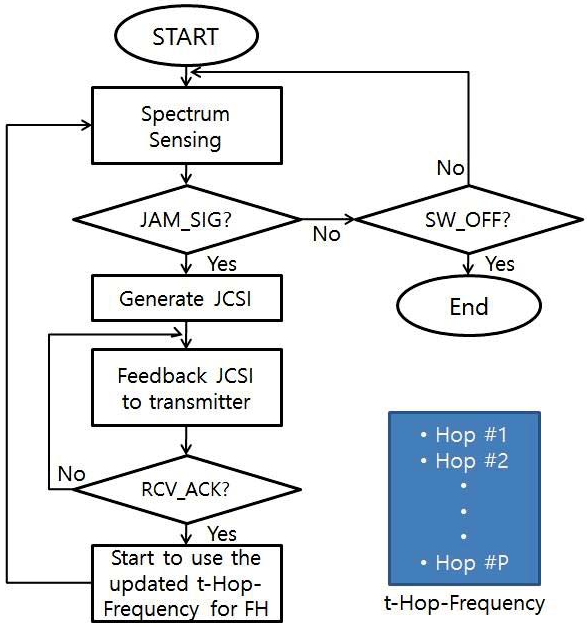


Fig. 3. Cognitive anti-jamming operation.

3.2.1 Signal Energy Detection

The light-red-coloured box in Fig. 2 indicates the signal energy detection (SE) module in the designed simulator. SE is relatively simple but has a faster processing time than other methods. Assume that average received signal energy per channel has been limited due to power resource control. Hence, in the SE module, the average signal power of the *h*th (=1, 2, …, *H*) channel $\bar{P}_{h,r} = 1/N \sum_{n=0}^{N-1} |r_{h,n}|^2$ (where $r_{h,n}$ denotes the *h*th channel received signal samples and *N* (>> 1) denotes the observed sample size) compares to a given threshold Δ_{SE}. If $\bar{P}_{h,r} \geq \Delta_{SE}$, a jammer signal is detected at the *h*th channel, otherwise a normal signal is detected. Through this scanning process, if the jammer signal is detected on adjacent *H* (> 1) channels, we can hypothesize that the PBNJ event occurs. During this scanning process, the receiver terminal periodically generates (or updates) JCSI for the jammed channel and feedbacks it to the transmitter terminal.

3.2.2 Signal-to-Jammer-plus-Noise Ratio Detection

The light-blue-coloured box in Fig. 2 indicates the SJNR detection module in the designed simulator. Assume that the received signal at the *h*th channel is denoted as $r_{h,k} = s_{h,k} + j_{h,k} + n_{h,k}$. The signal power to jammer-plus-noise power ratio (SJNR) representing signal quality is given as [7]

$$SJNR = \frac{\sigma_s^2}{\sigma_j^2 + \sigma_n^2} = \frac{\left\{\frac{1}{N}\sum_{k=1}^{N}|r_{h,k}|\right\}^2}{\left\{\frac{1}{N}\sum_{k=1}^{N}|r_{h,k}|^2 - \left(\frac{1}{N}\sum_{k=1}^{N}|r_{h,k}|\right)^2\right\}}$$

For the jammer signal detection, the measured SJNR per channel is compared to a threshold Δ_{SJNR}. If $SJNR < \Delta_{SJNR}$, then the jammer signal is detected at the *h*th channel, otherwise the normal signal is detected. Through this scanning process, if the jammer signal is detected on adjacent *H* (> 1) channels, we can hypothesize that the PBNJ event occurs. During this scanning process, the receiver terminal periodically generates (or updates) JCSI for the jammed channel and feedbacks it to the transmitter terminal.

3.2.3 Hybrid Detection

In simulator, we also implement a hybrid detection method that combines beacon (pilot data) sensing with SS-based sensing. Fig. 4 shows the hybrid detection block. In simulator, for simplicity, we consider two kinds of hybrid methods depending on the SS scheme: energy-based spectrum sensing with beacon detection (BC_SE) and signal-to-jammer-plus-noise rate (SJNR)-based spectrum sensing with beacon detection (BC_SJNR).

In hybrid detection block, a pre-known pilot data is used to double check the validity of the jamming signal detected by one of spectrum sensing-based techniques (SE or SJNR). We assume that beacon channel is uniformly distributed over all hopping channels.

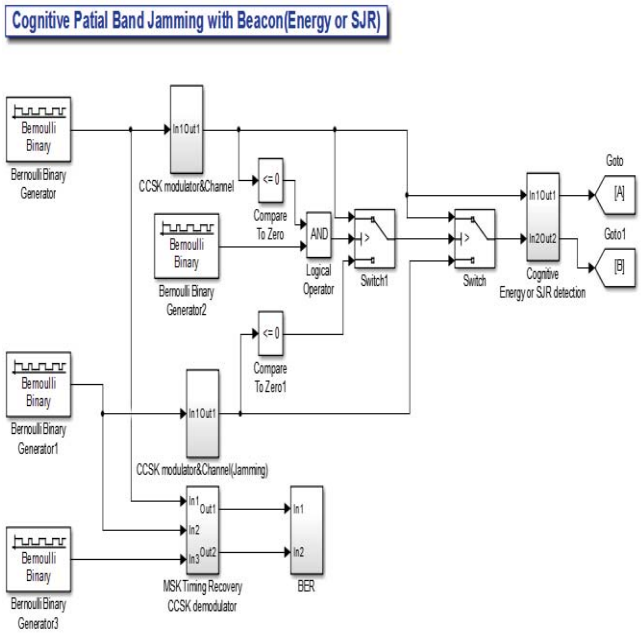


Fig. 4. Beacon + SE or Beacon + SJNR-based hybrid detection block.

In these hybrid methods, only when the jammer signal is sensed by both SS and beacon sensing, we decide that the jammer event occurs. Hence, the reliability of the jammer signal detection is improved by keeping the false alarm probability P_{fa} below threshold ($\max\{P_{fa}\} \leq 0.01\%$) while reducing the miss-detection probability P_{md}.

4 Simulation Results

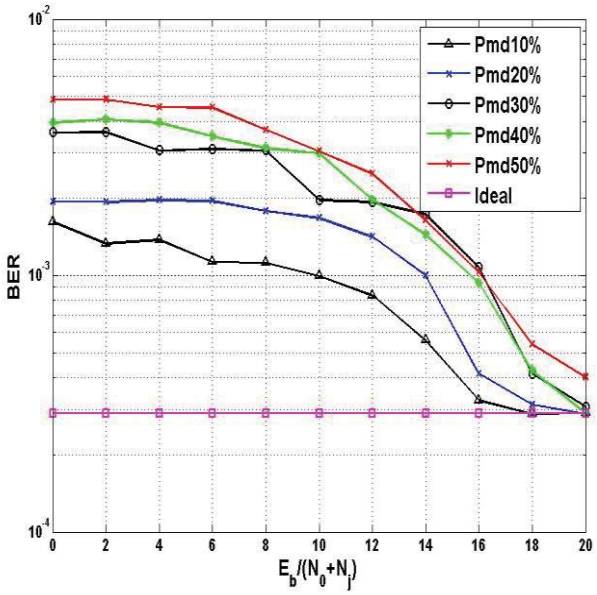


Fig. 5. BER performance versus P_{md} of the proposed TDL

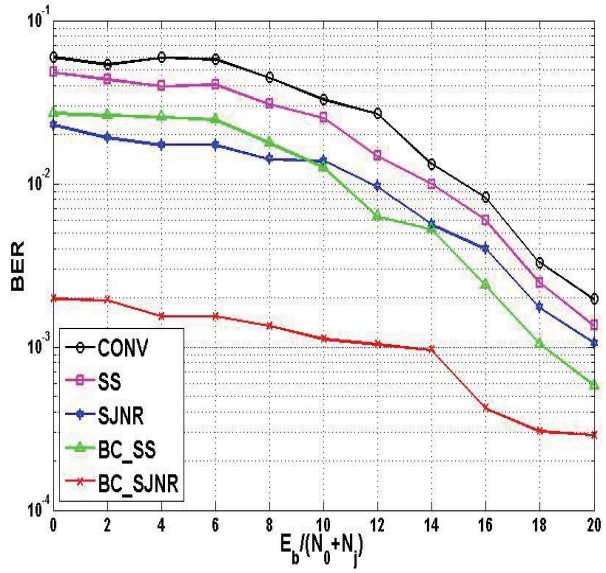


Fig. 6. BER performance of the conventional TDL and the proposed TDL over partial-band noise jamming with ρ = 10%

In simulation, we assume that P_{fa} is less than 0.01%. Assume that every normal FH channel (i.e., no jamming channel) has E_b/N_0 = 20dB. Then, SJNR, defined as $E_b/(N_j + N_0)$, of any jammed channel would be less than 20dB. We also assume that the k factor is equal to 10 and the Doppler frequency is 777 Hz (where the carrier frequency f_c = 8.4 GHz is assumed) considering the real terminal speed.

Fig. 5 shows the BER performance versus P_{md} of the proposed TDL with cognitive anti-jamming capability. As P_{md} increases, we can observe that the difference from the ideal curve (i.e., P_{md} = 0%) becomes larger. Specifically, the curve of P_{md} = 10% has approximately 4dB gain as compared to the one of P_{md} = 20% at the basis of BER = 10^{-3} and approximately 6dB gain as compared to the one of P_{md} = 30% at the basis of BER = 10^{-3}.

Fig. 6 compares the BER results of existing TDL [1], [2] and proposed TDL under the PBNJ environments. Assume that the jamming spectrum rate ρ = 10% and every normal FH channel (i.e., channel without jamming) has E_b/N_0 = 20dB. In the designed TDL simulator, we set a threshold satisfying P_{md} = 30% and max{P_{fa}} ≤ 0.01% for all jammer detection methods. When comparing the proposed TDL to the conventional TDL ('CONV'), the former has 2 to 10 dB gain over the latter at the basis of 2 x 10^{-3} BER. Even for the same proposed TDL, it is observable that the system gain is so dependent upon the chosen jammer detection scheme. It verifies that hybrid SS-based scheme is much better than generic SS-based scheme. For instance, SJNR has 1 dB gain over SE, at the basis of 2 x 10^{-3} BER. However, BC_SE has approximately 2dB gain over SS and BC_SJNR has about 10dB gain over SJNR, at the basis of 2 x 10^{-3} BER. We can also confirm that BC_SJNR is much better than the other schemes including BC_SE. When comparing BC_SE to SJNR, BC_SE is superior to SJNR, especially, for $E_b/(N_j + N_0)$ > 10dB.

5 Conclusions

We designed a TDL system simulator with cognitive anti-jamming capability using Matlab/Simulink. Numerical results using the simulator proved that the proposed TDL is more robust to the conventional TDL over partial-band noise jamming conditions. We confirmed that hybrid SS-based detection with some additional complexity due to beacon channel is superior to generic SS-based detection. We also observed that even simple detection schemes like SE or BC_SE improve the system gain of existing Link-16-based TDL 0.5dB up to 3dB.

6 References

[1] B. R. Lee, E. K. Jung, and S. Choe, "Link-16 simulator design over jamming environments and Link-16 performance analysis with or without time synchronization," Telecom. Review, vol. 23, no. 2, pp. 261-275, April 2013.

[2] David S. Alberts, John J. Garstka, and Frederick P. Stein, "Network Centric Warfare," DoD C4ISR Cooperative Research Program (CCRP), Feb. 2000.

[3] Xiaohua Li and Wednel Cadeau, "Anti-jamming Performance of Cognitive Radio Networks," Information Sciences and System (CISS), pp. 1-6, March 2011.

[4] Sudarxhan Prasad and David J. Thuente, "Jamming Attacks in 802.11g-a Cognitive Radio Based Approach," IEEE MILCOM, pp. 7-10, Nov. 2011.

[5] Kwangsung Ju and Kwansue Chung, "Jamming Attack Detection and Rate Adaption Scheme for IEEE 802.11 Muti-hop Tactical Networks," International Journal of Security and Its Applications (IJSIA), vol.6, no.2, pp. 149-154, Apr. 2012.

[6] Wenyuan Xu, Ke Ma, and Wade Trappe, "Jamming Sensor Networks: Attack and Defense Strategies," IEEE Network, vol.20, no.3, pp. 41-47, May/June 2006.

[7] F. Harris and C. Dick, "SNR Estimation Techniques for Low SNR Signals," Wireless Personal Mutimedia Communications (WPMC), pp. 276-280, Sept. 2012.

The Research on the Wear Laws of the Grinding Wheel in Robot Polishing Process

[1] Yujiao Liu Engineer, [2] Yongwei Zhang Engineer, [3] Professor Xitao Zheng
[1,2]Shanghai Moses Marine Engineering Corp., Shanghai, 201306, China
[3]Zhejiang Ocean University, Zhejiang, 316022, China

Abstract-*This paper presents a method to estimate the variation of grinding wheel during robot-operated polishing process. The robot operation can provide controllable moving speed and varied contact force, which are vital to the quality of polishing. The mathematical model is provided and the calculation is compared to those of actual operation, results show that it is possible to predict the wearing off of grinding with pretested contact force data and optimize moving plan for the grinding head. With out the operation of robot arm and the speed adjustment to enable a stable contact force, the wearing off condition can be quite different than the data presented. The calculation is important to the inverse calculation of grinding head moving speed and the external force applied by the robot arms. The industry application shows that the wearing off of grinding wheel per pot is only a quarter to that of human operated polishing devices, which means some savings in the material and labor cost.*

Keywords: *grinding wheel, grinding head, pot polishing, iron polishing, robot arm*

1 Introduction

Grinding is one of the last and the most important link in traditional metal part processing, which takes the proportion of 30% - 40% in the total machining. Its quality will decide the sale of the product so it is an important procedure for pot production.

In China, the pot polishing industry is still in the manual operation period, with heavy air pollution and noise pollution. Because of the randomness of the contact force exerted, the grinding wheel will be exalted every 43 minutes. Changing grinding wheel will need the intervention of machine maintenance technician and waste time and money. While the industry is facing more and more restriction from the government, adopting robot arms to do the polishing will be a trend. In order to apply robot arm to substitute human being, the robot arm moving curve and exerted force need to be determined. By examine the grinding wheel wear off in the ideal condition; we can get the wearing off speed, which can then be used to calculate the robot control logic.

Usually, it is the morphology parameters, the passivation and wear of the grinding wheel's working surface that has an important influence in the quality of machining. The wear and passivation would make a lower grinding, and also produce large amounts of heat during the cutting, which would cause an effect to the dimensional accuracy and surface quality of the workpiece. If the passivation of grinding wheel reaches a certain extent, the grinding wheel is still in use without reparing, then not only the size of the workpiece precision won't be guaranteed, but also it may cause that the grinding heat and the grinding force rise sharply due to the passivation, which will result in the grinding burns on the workpiece surface, even worse, it may cause severe accident because of the rupture of grinding wheel.

Grinding is a high precision machining method. It is difficult to proceed precise online

measuring to the diameter of the grinding wheel because of the higher measurement accuracy requirements. Also, it is the sand on the surface of grinding wheel that makes it difficult to proceed high precise detection for the wear of grinding wheel. Therefore, at present it is difficult to realize self-compensation to the grinding wheel wear in iron pan polishing robot. For this situation, this paper proposes a wear model of grinding wheel based on iron pan polishing robot, and compares it with the actual wear of grinding wheel, and makes further correction of the model, so as to provide the basis to achieve automatic compensation to the wear grinding wheel.

So the research on the wear laws of the grinding wheel is the key technology to the development of ultra-high grinding speed and super precision and high automation technology.

2 The establishment of the grinding wheel

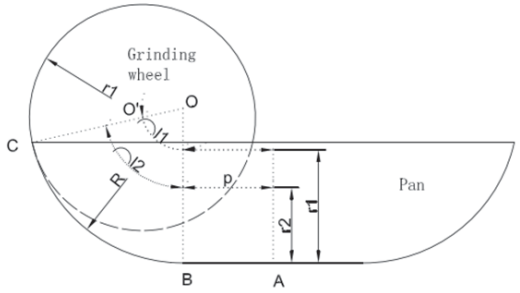


Figure 1 The influence of wheel's radius change on its walking path during inside polishing

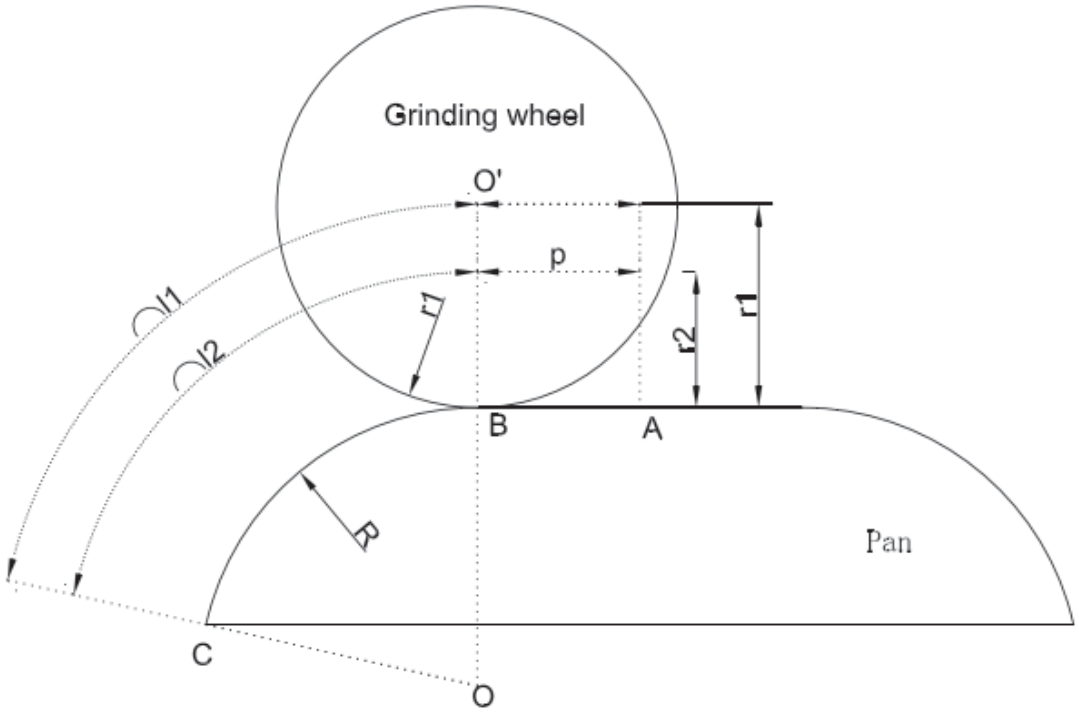


Figure 2 The influence of wheel's radius change on its walking path during outside polishing

As it is shown in figure 1 and figure 2, as the wastage of the grinding wheel, the radius becomes short, the walking path will change as follow. The walking path of both inside polishing and outside polishing can be decomposed into two parts, the walking distance p at the bottom of the pan and the walking distance *l* along the cambered surface. Their path computation formulae are as follows:

$$s_i = \alpha(R - r) + p \quad (1)$$

$$s_o = \alpha(R + r) + p \quad (2)$$

If the walking speed of wheel decelerate uniformly at the bottom of the pan, and then remains constant at the cambered surface of the pan. Assuming the wheel's walking speed in the center of the pan is *v*, and which in the cambered surface of the pan is *v*/2, the time cost of polishing the bottom of a pan can be calculated as:

$$t_p = \frac{3p}{4v} \quad (3)$$

Assume d_1 as the diameter of grinding wheel before polishing, set d_2 as the diameter of that after polishing, then the relationship between the wear volume of the wheel and its diameter can be expressed as:

$$V = (\frac{\pi}{4} d_1^2 - \frac{\pi}{4} d_2^2)B = \frac{\pi}{2} \bullet \frac{d_1 + d_2}{2} \bullet (d_1 - d_2) \bullet B \quad (4)$$

Assume d as the average diameter the wheel in the process of polishing, and Δd as the diameter variation of the grinding wheel.

$$d \approx \frac{d_1 + d_2}{2} \quad (5)$$

$$\Delta d = d_1 - d_2 \quad (6)$$

The polishing robot system can maintain a constant downward pressure of the grinding wheel to the pan, and the grinding wheel feed is proportional to the pressure. Assuming that feed speed at the bottom of the pan is *a*, and the wear rate of grinding wheel is Q, then the wear volume of grinding wheel to each pan can be expressed as formula 7 and 8.

$$V_i = Q_i \int_0^{\alpha} \frac{a\cos\alpha}{v_i/2}(R - \frac{d}{2})d\alpha + Qa\frac{3p}{4v_i} = \frac{Qa}{v_i}[\frac{3}{4}p + (2R - d)\sin\alpha] \quad (7)$$

$$V_o = Q_o \int_0^{\alpha} \frac{a\cos\alpha}{v_o/2}(R + \frac{d}{2})d\alpha + Qa\frac{3p}{4v_o} = \frac{Qa}{v_o}[\frac{3}{4}p + (2R + d)\sin\alpha] \quad (8)$$

With the above formula, we can get the relationship between the diameter change of a wheel after polishing a iron pan and its average diameter.

$$\Delta d_i = \frac{2Q_i a}{\pi B v_i}[\frac{\frac{3}{4}p + 2R\sin\alpha}{d} - \sin\alpha] \quad (9)$$

$$\Delta d_o = \frac{2Q_o a}{\pi B v_o}[\frac{\frac{3}{4}p + 2R\sin\alpha}{d} + \sin\alpha] \quad (10)$$

s_i – center displacement of inside polishing wheel (mm);

s_o – center displacement of outside polishing wheel (mm);

v_i – wheel's walking speed in the center of the pan during inside polishing (mm/s);

v_o – wheel's walking speed in the center of the pan during outside polishing (mm/s);

t_p - the time cost of polishing the bottom of a pan (s);

Q_i - the wear volume of the inside polishing wheel in unit time (mm^2/s);

Q_o - the wear volume of the outside polishing wheel in unit time (mm^2/s);

α- the central angle corresponds to the cambered surface of pan, as ∠BOC in fig.1and fig.2;

R - the radius corresponds to the surface of

pan (mm);

r - grinding wheel radius (mm);

d- the diameter of the grinding wheel (mm);

B - the width of the grinding wheel (mm);

p- the radius of the bottom of a pan (mm).

a-feed speed at the bottom of the pan(mm).

3 Experiment

3.1 Experiment conditions and methods

The experiment was carried on in the iron pan polishing robot of Shanghai Moses Marine Engineering Corp. The grinding conditions are shown in table 1. We chose ordinary brown corundum grinding wheel as the grinding wheel. While installing a grinding wheel, the center axis of the wheel should be in a horizontal plane, and be vertical with the center axis of workpiece. The wheel and pan rotate around their central axis respectively, in the meantime, the mechanical arm drives the wheel to move in the vertical way, and its effective polishing path starts from the center of pan, then goes along the surface of the pan to the edge, and always in the plane that pass through the center of the grinding wheel and parallel to the cylindrical bottom plane of the wheel.

Table 1 Grinding conditions

Iron pan polishing robot	Rotate rate of the wheel	Rotate rate of iron pan	Wheel's walking speed in the center of the pan	The theoretical feed of wheel at one time
	4100rpm	530rpm	1.5mm/s	0.2mm
Iron pan	Diameter of its upside	Diameter of its bottom	Depth	Radian
	320mm	120mm	80mm	0.213889
Grinding wheel	Material	Granularity	Hardness	Size（mm） Diameter×Thickness
	brown corundum	80	G	Inside polishing 150×20 Outside polishing150×25

The diameter of grinding wheel is measured with vernier caliper, with accuracy of 0.02 mm. The rotate rates of the wheel and pan are measured by tachometer. The wheel's walking speed in the center of the pan starts as 1.5mm/s, then the speed decelerate uniformly at the bottom of the pan, and then remains 0.75mm/s at the cambered surface of the pan.

3.2 Test results and analysis

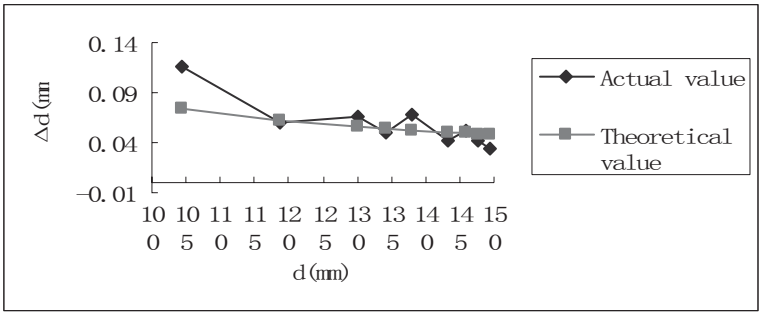


Figure 3 The relationship of grinding wheel's diameter variation of a pan with its diameter in

inside-polishing

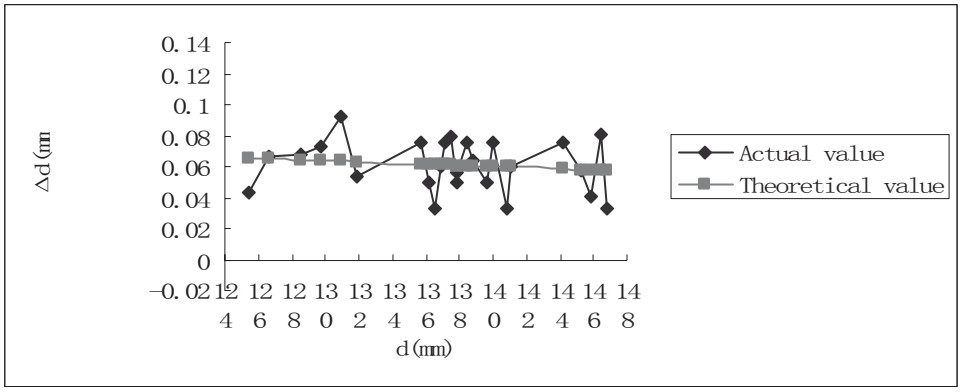


Figure 4 The relationship of grinding wheel diameter variation of a pan with its diameter in outside-polishing

Figure 3 and 4 show that the experimental values fluctuate between the theoretical value whether it is inside or outside polishing, which means that basically this model meets the actual situation.

4 Discussion

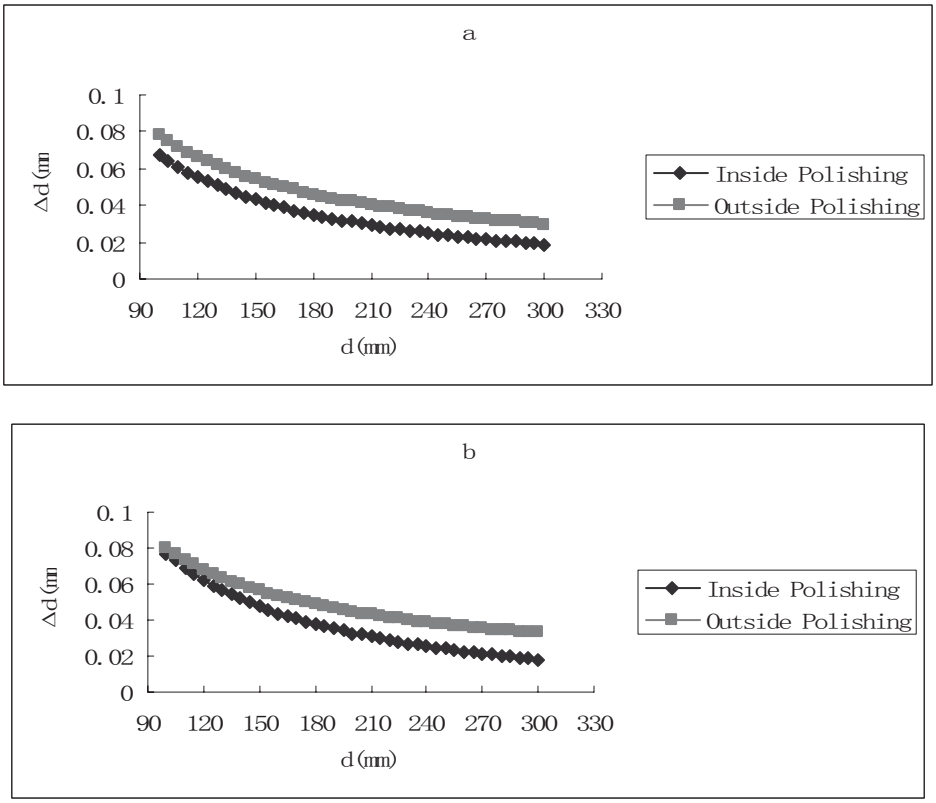


Figure 5 The relationship models between Δd and the diameter of the wheel

a. the model of which the grinding conditions are same to both inside and outside polishing

b. The model under experimental grinding condition

Δd- the wear variation of grinding wheel in polishing a pan

The figure shows that Δd increases with the decrease of the wheel diameter in both inside and outside polishing, the curve of inside polishing is steeper than that of outside polishing in the experimental grinding condition. The smaller Δd is, the less walking path change of grinding wheel is, and it also can be seen that the larger the diameter is, the smaller diameter reduction that caused by the wear loss, and the smaller the influence on the change of grinding wheel walking path. Also it can be seen from figure 5 that when the diameter of grinding wheel is larger than 180 mm, Δ d and d are linearly related, which is more conducive to adjust the walking path of grinding wheel. Thus, it is conductive that choosing larger grinding wheel on the allowable condition to improve the stability of the organization and ensure machining quality.

5 Conclusion

From the discussion we can see that the wear speed of grinding wheel can be predicted, the consumed current by the grinding motor, which is perpendicular to the contact, are also provided and fluctuation curve can be used for the robot control programming. The actual running of the system prove that the robot-operated polishing machine can be effective, easy controlled and the contact force can be fine calibrated using the data provided in this paper. Further work is necessary to simulate the frequency change of the moving part and the grinding head to enable more intelligent fault tolerance and defect detection for the robot polish system.

6 References

[1] Haiyu Liu. The Research on Some Chemical and Physical Test Methods in the Chemical and Fiber Production Process [D].Hunan University, Changsha.2006.

[2] Xiaofeng Zhang. The basic research of grinding wheel physiognomy based on binocular vision detection technology [D]. Nanjing: Nanjing university of aeronautics and astronautics.2007.09.

[3] Hanzhong Li. The morphology detection on the surface of grinding wheel [J]. Diamond & Abrasives Engineering.1986.04.

[4] Xinwang Xie. On-line Monitoring Technology of the Wear and the Degree of Passivation of the Grinding Wheel [D].Changchun University of Technology.

SESSION

POSTER PAPERS

Chair(s)

TBA

Exponential Synchronization of Lur'e Networks with Stochastic Disturbances via Impulsive Control

J.H. Park[1], **Ze Tang**[1], **S.M. Lee**[2]
[1]Department of Electrical Engineering, Yeungnam University,
Kyongsan, Republic of Korea. Email: jessie@ynu.ac.kr.
[2]School of Electronics Engineering, Daegu University,
Kyongsan, Republic of Korea.

Abstract— *In this paper, the exponential synchronization of the stochastic Lur'e networks with nonlinear and directed couplings is investigated. Through the impulsive control strategy, a criterion is derived to guarantee the realization of the exponential synchronization by means of the Lyapunov stability theorem and the comparison principle.*

Keywords: Lur'e network, impulsive controller, synchronization.

1. Introduction

Among the collective behaviors of complex networks, synchronization is one of the most important behaviors which plays an important role in practical applications [1]-[2]. The common and effective control strategies are pinning control, impulsive control, intermittent control and so on. For example, [3] analyzed a kind of complex dynamical network with time delayed coupling and non-delayed coupling. It is well known that impulse is a common phenomenon in many evolving networks. In the impulsive communication framework, the dynamics of each node is only affected by its neighbors at the impulsive instants and there exist impulsive effects in the dynamical behavior of nodes. Good prior works had been made on the synchronization of complex works with impulses controllers, see [4]-[6] and the references therein. For instance, [6] investigated the global exponential synchronization of delayed complex dynamical networks with nonidentical nodes and stochastic perturbations through adaptive control and impulsive control schemes. In this paper, the outer exponential synchronization of stochastic networks will be discussed. The dynamics of every node in the network studied in this paper follows the Lur'e system which has nonlinear local dynamical behaviors. In order to simulate the real world, we take the stochastic phenomenon into consideration since the instantaneous perturbations and abrupt changes exist in many realistic networks. The information interchange between nodes is in a nonlinear and asymmetrical way. By using impulsive control strategy, a sufficient condition is obtained to guarantee the realization of the exponential synchronization for all initial values by means of the Lyapunov stability theorem and the comparison principle.

2. Problem description

In this section, some preliminaries will be firstly given. Then, we give the complex dynamical Lur'e network model with stochastic perturbations with nonlinear coupled functions and an asymmetrical coupled matrix.

Definition 1: ([7]) The average impulsive interval of the impulsive sequence $\chi = \{t_1, t_2, \cdots\}$ is less than T_a, if there exist a positive integer N_0 and a positive number T_a, such that $N_\chi(T,t) \geq \frac{T-t}{T_a} - N_0, \forall T \geq t \geq 0$ where $N_\chi(T,t)$ denotes the number of impulsive times of the impulsive sequence χ in the time interval (t,T).

Consider the following network model with nonlinear coupling, and the nodes' dynamics follows Lur'e system

$$\begin{aligned} dx_i(t) &= [Ax_i(t) + B\tilde{f}(Cx_i(t))]dt + \tilde{p}(x_i(t),t)dw(t) \\ &\quad + c\textstyle\sum_{j=1}^{N} g_{ij}\Gamma\tilde{H}(x_j(t))dt \end{aligned} \tag{1}$$

where $x_i(t) = [x_i^1(t), x_i^2(t), \cdots, x_i^n(t)]^T \in \mathcal{R}^n$, for $i = 1, 2, \cdots, N$. $A \in \mathcal{R}^{n\times n}, B \in \mathcal{R}^{n\times m}$, $C \in \mathcal{R}^{m\times n}$ are constant matrices. The constant $c > 0$ denote the coupling strength and $\Gamma = diag\{\gamma_1, \gamma_2, \cdots, \gamma_n\} \in \mathcal{R}^{n\times n}$ is the inner-linking matrix with $\gamma_i \geq 0$. Function $\tilde{f} : \mathcal{R}^m \to \mathcal{R}^m$ is memoryless nonlinear vector valued function which is continuously differentiable on $\mathcal{R}$, $\tilde{f}(0) = 0$. $G = (g_{ij}) \in \mathcal{R}^{N\times N}$ is the the coupling matrix and $g_{ij} > 0$ if there is a connection from node i to node $j (i \neq j)$, otherwise $g_{ij} = 0$ and it satisfies zero-sum-row condition and $g_{ij} \neq g_{ji}$, which means the network we study is a directed information transmission network. The nonlinear coupling functions $\tilde{H}(x_j(t)) = (\tilde{h}(x_j^1(t)), \tilde{h}(x_j^2(t)), \cdots, \tilde{h}(x_j^n(t)))^T$, satisfies the following conditions: $\frac{\tilde{h}(u)-\tilde{h}(v)}{u-v} \geq \vartheta > 0$, for any $u, v \in \mathcal{R}$. $w(t) \in \mathcal{R}^m$ is an $m-$dimensional Brownian motion, $\tilde{p} : \mathcal{R}^n \times \mathcal{R}^+ \to \mathcal{R}^{n\times m}$ is the noise intensity function matrix satisfying $\tilde{p}(t, 0^n) = 0^{n\times m}$.

The salve system with impulsive control scheme can be described as

$$\begin{aligned} dy_i(t) &= [Ay_i(t) + B\tilde{f}(Cy_i(t))]dt \\ &\quad + \tilde{p}(y_i(t),t)dw(t) + c\textstyle\sum_{j=1}^{N} g_{ij}\Gamma\tilde{H}(y_j(t))dt. \end{aligned} \tag{2}$$

We control the first l nodes in the Lur'e network with the following impulsive control strategy

$$u_i(t) = \sum_{k=1}^{\infty} \mu[y_i(t) - x_i(t)]\delta(t - t_k), \tag{3}$$

where the constant $\mu \in (-2, 0)$, which means the corresponding impulsive effects are synchronizing, the time series $\chi = \{t_1, t_2, \cdots, t_n, \cdots\}$ is a sequence of strictly increasing impulsive instants satisfying $t_{k-1} < t_k$ and $\lim_{k\to+\infty}(t_k) = +\infty$. And $\delta(\cdot)$ is the Dirac impulsive function, which satisfies $\int_{-\infty}^{+\infty} \delta(t)dt = 1$. Define $e(t) = y(t) - x(t)$, the error system can be written as

$$\begin{array}{ll} de_i(t) & = [Ae_i(t) + Bf(Ce_i(t))]dt + p(e_i(t), t)dw(t) \\ & +c\sum_{j=1}^{N} g_{ij}\Gamma H(e_j(t))dt \qquad t \neq t_k, \\ e_i(t_k^+) & = e_i(t_k^-) + \mu e_i(t_k^-) \qquad t = t_k, \end{array} \tag{4}$$

and denote $f(Ce_i(t)) = \tilde{f}(Cy_i(t)) - \tilde{f}(Cx_i(t))$, $p(e_i(t), t) = \tilde{p}(y_i(t), t) - \tilde{p}(x_i(t), t)$, $h(e_i(t), t) = h(y_i(t), t) - h(x_i(t), t)$, $H(e_i(t)) = [h(e_i^1(t)), h(e_i^2(t)), \cdots, h(e_i^n(t))]^T$, for $i = 1, 2, \cdots, N$; $j = 1, 2, \cdots, n$.

In order to derive the main results of the stochastic impulsive dynamical Lur'e network, we make the following hypothesis.

Assumption 1: There exists a constant $\zeta > 0$ such that $\| \tilde{f}(u) - \tilde{f}(v) \| \leq \zeta \| u - v \|$ holds for any $u, v \in \mathcal{R}^n$.

Assumption 2: Assume the function matrix $\tilde{p} : \mathcal{R}^n \times \mathcal{R}^+ \to \mathcal{R}^{n\times m}$ satisfy $trace[(\tilde{p}(u) - \tilde{p}(v))^T(\tilde{p}(u) - \tilde{p}(v)] \leq \| M(u - v) \|^2$ for any $u, v \in \mathcal{R}^n$, where M is a known constant matrix with compatible dimensions.

3. Main results

In the following part, we investigate the global and exponential synchronization of the synchronization. Throughout the paper, we always assume that $e_i(t_k) = e_i(t_k^-)$. Let $\xi = (\xi_1, \xi_2, ..., \xi_N)^T$ be the left unitization eigenvector of G corresponding to the eigenvalue 0.

Theorem 1: Suppose that Assumption 1 and 2 hold and the average impulsive interval of the impulsive sequence $\chi = \{t_1, t_2, \cdots\}$ is less than T_a. Let $\rho = (1 + \mu)^2, \mu \in (-2, 0)$, and consider the impulsive stochastic dynamical master Lur'e network (1) and the salve Lur'e network (2). Then the controlled impulsive stochastic dynamical network (4) is globally and exponentially stable if the following inequality is satisfied

$$\begin{array}{ll} \frac{\ln\rho}{T_a} & +\lambda_{max}(A + A^T + M^T M) \\ & +2\zeta\sqrt{\lambda_{max}(B^T B)\lambda_{max}(C^T C)} \leq 0. \end{array} \tag{5}$$

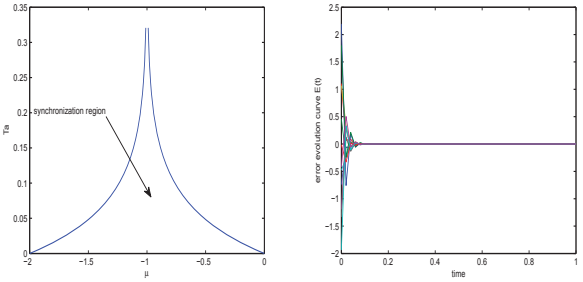
Proof: From a Lyapunov function $V(t) = \sum_{i=1}^{N} \xi_i e_i^T(t)e_i(t)$, and its $\mathcal{L}V(t)$([5]), it can be verified the theorem. The detail is omitted. ■

Example: The network (1) and (2) have the parameters as

$$A = \begin{bmatrix} -3.2 & 10 & 0 \\ 1 & -1 & 0 \\ 0 & -15 & -0.0385 \end{bmatrix}, B = \begin{bmatrix} 5.9 \\ 0 \\ 0 \end{bmatrix},$$

and $C = [1, 1, 1]$, $\tilde{f}(Cx_1(t)) = \frac{1}{2}(\mid x_1(t)+1 \mid - \mid x_1(t)-1 \mid)$. $dw(t)$ is an 3−D Brownian motion, $\tilde{p}(x_i(t), t) = 0.5 \cdot \|x_i(t)\| \cdot I_3$, $h(x(t)) = 8x(t) + 0.2\sin(x(t))$. We consider a network consists of 4 nodes. For given parameters, we derive $\rho = 0.040$, $\delta = \lambda_{max}(A + A^T + M^T M) + 2\zeta\sqrt{\lambda_{max}(B^T B)\lambda_{max}(C^T C)} = -321.8876$, and $\frac{\ln\rho}{T_a} + \delta = -293.1626 < 0$. From the above analysis, the above parameters could let the networks achieve synchronization.

In Fig. 1, we plot the estimation of boundaries of synchronization region and the synchronization error curve.



(a) (b)

Fig. 1: (a). Synchronization region with respect to μ, T_a. (b). Error evolution curve $E(t)$.)

Acknowledgement: The work was supported by Basic Science Research Program through the NRF funded by the Ministry of Education (2013R1A1A2A10005201).

References

[1] P. Buijs et al., Synchronization in cross-docking networks: A research classification and framework, *European J. Operational Research*, 239 593-608, 2014.

[2] J.W. Feng et al., Cluster synchronization of non-linearly coupled Lur'e networks with identical and non-identical nodes and an asymmetrical coupling matrix, *IET Contr. Theory Appl.*, 7, 2117-2127, 2013.

[3] L. X. Yang J. Jiang, Adaptive synchronization of drive-response fractional-order complex dynamical networks with uncertain parameters, *Comm. Nonl. Sci. Numerical Simula.*, 19, 1496-1506, 2014.

[4] J.Q. Hu et al., Synchronization of stochastic dynamical networks under impulsive control with time delays, *IEEE Tr. Neural Netw. Learning Sys.*, 25, 1758-1768, 2013.

[5] B. Liu, Stability of solutions for stochastic impulsive systems via comparison approach, *IEEE Tr. Autom. Contr.*, 53, 2128–2133, 2008.

[6] X.S. Yang et al., Stochastic synchronization of complex networks with nonidentical nodes via hybrid adaptive and impulsive control, *IEEE Tr. Circ. Sys. I: Regular Papers*, 59, 371-384, 2011.

[7] J.Q. Lu et al., Nariman Mahdavi, C. Huang, Synchronization control for nonlinear stochastic dynamical networks: pinning impulsive strategy, *IEEE Tr. Neural Netw. Learning Sys.*, 23, 285–292, 2012.

[8] D.L. Chen et al., Impulsive observer for input-to-state stability based synchronization of Luŗe differential inclusion system, *Comm. Nonl. Sci. Numerical Simula.*, 17, 2990-2996, 2012.

Simulation-Based Multiobjective Evolutionary Approach for Traffic Signal Design

H.-J. Cho[1, 2], Y.-H. Huang[2], M.-C. Hwang[3], C.-Y. Chen[4], S.-C. Hung[5] and F. Yu[2]
[1]Chairman of the Board, China Engineering Consultants Inc., Taipei, Taiwan
[2]Department of Transportation and Logistics Management, National Chiao Tung University, Hsinchu, Taiwan
[3]Microelectronics and Information Systems Research Center, National Chiao Tung University, Hsinchu, Taiwan
[4]Institute of Communications Engineering, National Chiao Tung University, Hsinchu, Taiwan
[5]Institute of Biomedical Engineering, National Chiao Tung University, Hsinchu, Taiwan

Abstract –*The traffic signal design problem is a typical multiobjective optimization problem. In this research, a simulation-based multiobjective evolutionary approach for traffic signal design problems is proposed. This approach has a robust advantage that the numerical simulators and the optimization solvers can be changed and used easily for other applications. In this research, it combined the NSGA-II with a microscopic traffic simulator, CORSIM, to solve the traffic signal optimization problem with multiple objectives. A road network which has 17 nodes, 30 links and 3 signal intersections is demonstrated for the numerical example. The Pareto front of optimal compromise solutions can provide the trade-off information between different objectives to decision makers.*

Keywords: Multiobjective optimization, simulation-based optimization, traffic signal design

1 Introduction

Due to the rapid increase in the number of vehicles and in the traffic demand, the traffic congestion becomes one of the severest problems in large cities. An efficient traffic signal design can mitigate traffic congestion, reduce environment pollution and ensure the safe crossing of opposing vehicles and pedestrians traffic in the road network. The traffic signal design problem is to find the optimal signal parameters, such as green split, cycle length, offset and phase sequence, which can improve the performance of existing facilities in the road network.

The traffic signal design problem usually has several goals to be satisfied simultaneously, and it is a typical multiobjective optimization problem. Sun et al. [1] applied the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [2] to solve a multiobjective signal timing optimization problem with minimizing the total delay and the total number of stops. Abbas et al. [3] proposed a multiobjective optimization model for timing signal system in oversaturated conditions and investigated the relationship between different objectives under various degrees of saturations. Schmocker et al. [4] applied fuzzy logic to solve a multiobjective signal control problem. The membership functions of the fuzzy logic are optimized by a GA combined with a microscopic traffic simulator. Lertworawanich et al. [5] optimized the signal timing controls of oversaturated networks based on a GA coupled to the cell transmission model. Li et al. [6] applied NSGA-II combined with a microscopic traffic simulator to solve a multiobjective signal control problem for isolated intersection under oversaturated conditions. The objectives included maximizing throughput and minimizing average queue ratio. Based on the preceding reviews, multiobjective evolutionary approach, such as NSGA-II, is a popular heuristic algorithm used to solve multiobjective traffic signal design problems. It can provide multiple Pareto-optimal solutions simultaneously and efficiently. It is also easy to cooperate with different traffic simulators or models which can describe the traffic behavior appropriately under different traffic conditions. The results can provide the trade-off information between different objectives to decision makers by evaluating the set of optimal compromise solutions.

This research attempts to solve the traffic signal design problem by a simulation-based multiobjective evolutionary algorithm (MOEA). The approach combines the NSGA-II with a microscopic traffic simulator, CORSIM [7]. Then we apply the approach to find the optimal green splits in an example network with 3 intersections. The objectives are minimizing total delay and maximizing throughput in the network. Finally, we will compare the compromise solutions produced by the multiobjective optimization with the solutions of a pure throughput-maximization and a pure delay-minimization, respectively.

2 Simulation-based MOEA methodology

The simulation-based MOEA applied in this research is an objective-oriented optimization framework for general optimization problem [8, 9] shown in Fig. 1. A specific MOEA, the NSGA-II is chosen because of its efficiency in solving multiobjective optimization problem. The design of this framework has a robust advantage that the numerical simulators and as well as the optimization solvers can be changed and used easily for other applications. In this case,

the numerical simulator is a microscopic traffic simulator, CORSIM; the optimization solver is the NSGA-II.

CORSIM is a microscopic traffic simulator developed and maintained by the Federal Highway Administration. It has a capability simulating the vehicular and driver behavior under various traffic flow conditions. At each simulation time step, CORSIM will track the position and movement of each vehicle in the network and compute the performance indices. The movement of each vehicle is primarily based on the car-following model and lane-changing model. The NSGA-II algorithm has several improvements than NSGA, such as adding elitist mechanism, crowded distance estimation, and fast non-dominated sorting strategy. Compared with NSGA, the time complexity of NSGA-II drops to $O(MN^2)$ from $O(MN^3)$, where M and N are numbers of constraints and objectives, respectively.

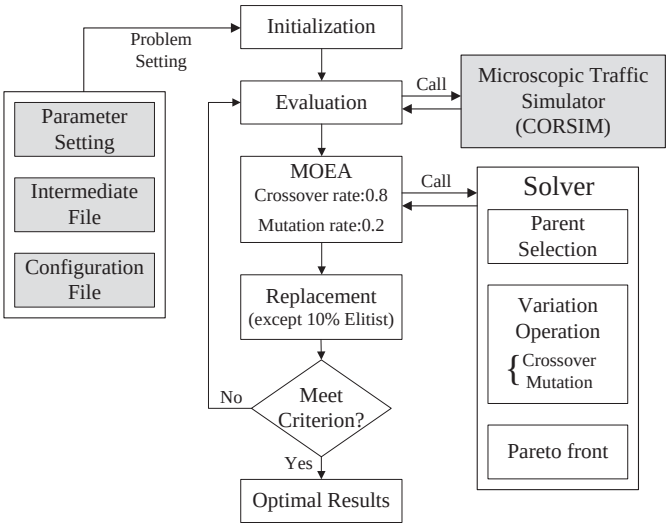


Figure 1. The simulation-based MOEA solver for traffic signal design problems

3 Numerical results

Consider a road network which has 17 nodes, 30 links and 3 signal intersections. Given the phase sequences and the offsets, we want to solve the green splits of the timing plans at these 3 intersections under minimizing total delay and maximizing throughput in the network. In this example, we set the population size is 20, and the number of generation is 70. We also solve the single-objective traffic signal optimization problems for the throughput-maximization and the delay-minimization, respectively. The Pareto front of the MOEA and the solutions of the single-objective optimizations are shown in Fig. 2.

In Fig. 2, point D represents the solution of the pure delay minimization, and the total delay is 7.92E+06; the throughput is 6373. Point T represents that of the pure throughput maximization, and the total delay is 1.08E+07; the throughput is 7041. The mean of total delay and throughput of Pareto front solutions are 8.50E+06 and 6888, respectively.

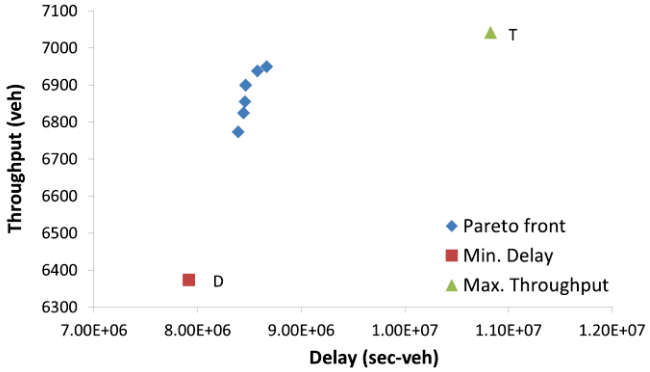


Figure 2. Distribution of the Pareto front and the single-objective solutions

4 Conclusions

A simulation-based MOEA for traffic signal design problems is proposed in this research. This simulation-based optimizer consists of two components. In this case, the numerical simulator component is a microscopic traffic simulator, CORSIM; the optimization solver component is the NSGA-II. Finally, a road network which has 17 nodes, 30 links and 3 signal intersections is demonstrated for the numerical example. The Pareto front of the multiobjective traffic signal optimization problem under minimizing total delay and maximizing throughput is illustrated in Fig. 2. The Pareto front can provide the trade-off information between different objectives to decision makers by evaluating the set of optimal compromise solutions.

5 Acknowledgement

This research was partially supported by the Ministry of Science and Technology through contract number 103-2221-E-009-158 and 103-2622-E-009-012; Ministry of Transportation and Communications R.O.C. under contract number MOTC-STAO-103-02 and MOTC-STAO-103-03; China Engineering Consultants, Inc., under contract number CECI-01921. We also acknowledge Prof. Yiming Li's valuable suggestion for this research.

6 References

[1] Sun, D., R.F. Benekohal, and S.T. Waller. "Multiobjective traffic signal timing optimization using non-dominated sorting genetic algorithm", *IEEE Intelligent Vehicles Symposium 2003 Proceedings*, 2003.

[2] Deb, K., et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Transactions on Evolutionary Computation*, Vol. 6, pp. 182-197, 2002.

[3] Abbas, M.M., H.A. Rakha, and P. Li. "Multi-objective strategies for timing signal systems under oversaturated conditions", *Proceedings of the 18th IASTED International Conference*, Anaheim, CA, pp. 580-585, 2007.

[4] Schmöcker, J.D., S. Ahuja, and M.G.H. Bell. "Multi-objective signal control of urban junctions–Framework and a London case study," *Transportation Research Part* C, Vol. 16, pp. 454-470, 2008.

[5] Lertworawanich, P., M. Kuwahara, and M. Miska. "A new multiobjective signal optimization for oversaturated networks," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 12, pp. 967-976, 2011.

[6] Li, Yan, et al. "Multi-Objective Optimization of Traffic Signal Timing for Oversaturated Intersection," *Mathematical Problems in Engineering*, Vol. 2013, 2013.

[7] CORSIM User Manual, FHWA, U.S. Department of Transportation, McLean, Virginia, 1998.

[8] Yiming Li, Shao-Ming Yu, and Yih-Lang Li, "Electronic Design Automation Using a Unified Optimization Framework," Mathematics and Computers in Simulation, Vol. 79, No. 4, Dec. 2008, pp. 1137-1152.

[9] Sheng-Chin Hung, Chieh-Yang Chen, Chien-Hsueh Chiang, and Yiming Li, "Circuit-Simulation-Based Multi-objective Evolutionary Algorithm with Multi-Level Clock Driving Technique for a-Si:H TFTs Gate Driver Circuit Design Optimization," in: W. C.-C. Chu et al. (Eds.), Intelligent Systems and Applications, IOS Press, Amsterdam, pp. 157-166, 2015.

Simulation of Depth Controllers for Underwater Glider

Joonhyuk Yoo[1] , Min-Ji Kim[2], and Moon G. Joo[2],
[1]Dept. of Embedded Systems Eng., Daegu University, Daegu, S. Korea
[2]Dept. of Information and Communications Eng., Pukyong National University, Busan, S.Korea

Abstract - *Comparison of depth control schemes for an underwater glider is presented. Considered underwater glider is a torpedo shape and has a controllable buoyancy bag and a movable center of gravity. While a underwater robot with propeller thrust needs lots of energy to surge, an underwater glider requires small amount of energy to inflate/deflate buoyancy bag and/or to move battery pack. Through the zig-zag gliding, underwater glider moves forward. The depth of the underwater glider is controlled by changing the buoyancy mass and the attitude, by changing center of gravity. The performance of PID controller, LQR controller, and Lyapunov nonlinear controller is shown by simulations using Matlab/Simulink.*

Keywords: Underwater glider, control, Matlab, Simulink

1 Introduction

Underwater glider is a sort of an autonomous underwater vehicle which operates without the tether line connected to the mother ship. Compared with the conventional underwater vehicle with propeller thrusters which require a considerable amount of energy, the underwater glider consumes less energy because it moves by the vertical zigzag motion by using the energy only to inflate the buoyancy bag and move the battery pack in the hull shown in Fig. 1. Therefore, it has a longer operation time and a wider operation range than the conventional autonomous underwater vehicle has. In this process, the underwater glider controls its depth and attitude by changing the buoyancy mass and the center of gravity, respectively. For the underwater glider to go downward, for example, the center of gravity needs to move toward the head of the vehicle while the buoyancy bag is inflating to absorb outside water.

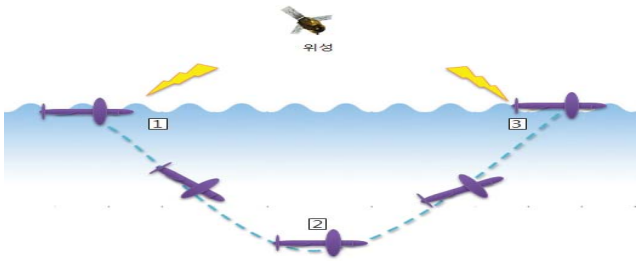


Figure 1 Basic principle of a conventional underwater glider.

Compared with the conventional underwater glider with long wings on its hull, we consider a torpedo shaped autonomous underwater vehicle equipped with a moving center of gravity and a controllable buoyancy bag in its hull as shown in Fig. 2. The hybrid underwater vehicle is to combine the gliding function into the conventional autonomous underwater vehicle. For example, the proposed hybrid underwater vehicle can approach to the far location without propeller noise and then, may operate as conventional autonomous underwater vehicle. The hybrid glider is simulated by using Matlab/Simulink. The dynamics of the underwater vehicle and the notations are well introduced in [1-4] and [5], where 12 state variables are used to describe the motion and the attitude of the vehicle in 6 degree of freedom.

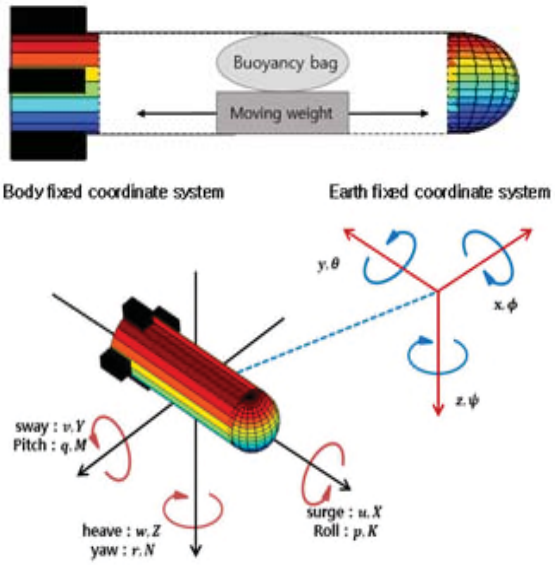


Figure 2 A hybrid underwater vehicle with a controllable buoyancy bag and a moving center of gravity.

2 Controllers of the hybrid underwater glider

For the hybrid underwater glider to glide stably the attitude is controlled by the position of the center of gravity and the depth is controlled by the inflation rate of the buoyancy bag.

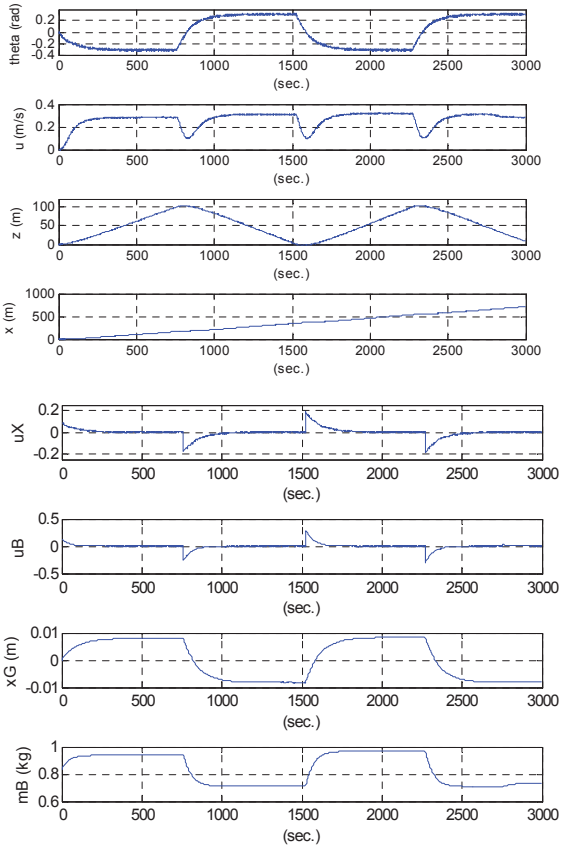


Figure 3 Control results from PID controller.

2.1 PID controller

PID controller does not require a sophisticated dynamics of the underwater vehicle. Only three gain values for tuning the PID controller were found by trial and error approach. Control results are shown in Figure 3.

2.2 LQR controller

Through the linearization of the underwater glider dynamics on the operating point, an LQR controller was developed. Solving the Ricatti equation is necessary to obtain the respective gain values satisfying a given performance index. Control results are given in Figure 4.

3 Conclusions

This paper demonstrates the performance of a simple PID controller and an LQR controller by the Matlab/Simulink simulations. Both of the controllers can maintain the gliding condition.

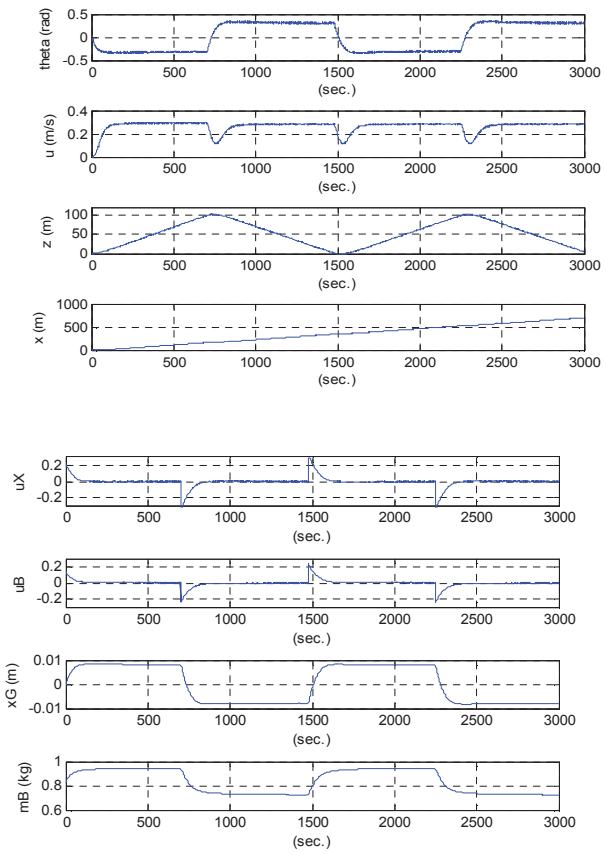


Figure 4 Control results from LQR controller.

4 References

[1] Thor I. Fossen, "Guidance and control of ocean vehicles", John Wiley & Sons, Ltd., Aug. 1994

[2] Timothy Presto, “Verification of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle”, M. S. thesis, Applied ocean science and engineering, MIT & WHOI, pp. 102-113, Aug. 2001.

[3] Joshua Grady Graver, “Underwater gliders: dynamics, control and design,” Ph. D. dissertation, Dept. mechanical and aerospace Eng., Princeton university, 2005.

[4] Naomi Ehrich Leonard and Joshua G. Graver, “Model-based feedback control of autonomous underwater gliders,” *IEEE journal of oceanic engineering*, vol. 26, no. 4, pp. 633-644, 2001

[5] Moon G. Joo and Zhihua Qu, “An autonomous underwater vehicle as an underwater glider and its depth control”, Int. Journal of Control, Automation and Systems, Vol. 13, No. 5, Oct. 2015. (in press)

INFLUENCE OF SOIL RESISTIVITY ON CATHODIC PROTECTION SYSTEMS – A NUMERICAL APPROACH

S.L.D.C. Brasil[1] , J.C.F. Telles[2], J.A.F. Santiago[3]

[1] School of Chemistry/Federal University or Rio de Janeiro, RJ - Brazil simone@eq.ufrj.br

[2] Civil Engineering Program/COPPE/Federal University of Rio de Janeiro, RJ - Brazil telles@coc.ufrj.br

[3] Civil Engineering Program/ COPPE/ Federal University of Rio de Janeiro, RJ - Brazil santiago@coc.ufrj.br

Abstract - *The protection against corrosion of buried and/or immersed metallic structures can be achieved by protective coatings and cathodic protection. Cathodic protection systems can be analyzed and optimized by the usage of numerical simulations. Among the different techniques that can be applied to this end, the boundary element method (BEM) has emerged as a strong alternative technique since boundary potential results are the only requirement needed for appropriate assessment of corrosion protection. The required information for the simulation includes experimental the following experimental data: cathodic polarization curves and medium resistivity measurements. The latter is discussed further in this work.*

Keywords: *Cathodic Protection, Soil Resistivity, Boundary Element Method, Numerical Simulation*

1. Introduction

The protection against corrosion of buried and/or immersed metallic structures can be achieved by protective coatings and cathodic protection. Galvanic or impressed current can be applied in order to achieve a minimum electrochemical potential that assure effective cathodic protection. The concepts of these techniques are widely discussed in the literature [1]. In real systems, however, it is difficult to measure the potential in order to guarantee satisfaction of the cathodic protection criterion (minimum potential of $-0.8V_{Ag/AgCl}$). So, the use of numerical simulations to evaluate and optimize cathodic protection systems has recently been widely applied. Computer simulations using three-dimensional applications of the boundary element method and experimental analysis of different resistivity media has been effectively carried out [2-5]. Parameters such as distance between anodes and pipelines, level of injected currents, efficiency of aged coatings and presence of scattered/localized defects can be taken into account. The good correlation between in-situ measurements and numerical results largely depends on the appropriate prescription of boundary conditions that are based on experimental tests obtained in laboratory. Therefore, polarization curves and resistivity values, applied as problem conditions, need to be carefully determined to guarantee accurate results. The importance of resistivity can be seen in a good number of practical applications such as: pier structures (seawater/soil interface), long buried pipelines (heterogeneous soil) and pipelines through soil/seawater media (horizontal directional drilling).

2. Methodology and Results

Figure 1 shows the experimental laboratory apparatus needed to obtain polarization curves, as typically seen in Figure 2. The soil resistivity, besides its physical-chemical properties, affects the potential distribution. The resistivity value can be obtained in laboratory using a soil box, as shown in Figure 4. In order to ensure the most unfavorable corrosion situation in the numerical simulation, the most aggressive condition (higher humidity) should be considered. As more water is added to the soil sample, increasingly lower resistivity values are obtained. A minimum resistivity is reached, for a certain quantity of water, depending on the retention capacity of the soil.

The resistivity of the medium where the structures are immersed has a great influence on the potential distribution. Horizontal directional drilling (HDD) and pier structures are examples involving different medium (soil/seawater or soil with different resistivity values) problems. Figure 4 presents the case of a pipeline passing through a HDD complex resulting in not enough cathodic protection over part of the pipeline system. Galvanic anodes protect the pipeline buried in seabed but not inside the HDD.

3. Conclusions

The present paper discusses the influence of resistivity of the medium in the design of cathodic protection systems commonly employed to protect metallic structures against corrosion. A brief presentation of how important reference parameters are determined through experimental techniques has been introduced. To illustrate the points of view put forward, a practical example is commented upon indicating that not enough protection may occur as soon as the pipeline crosses the interface between low and high resistivity media.

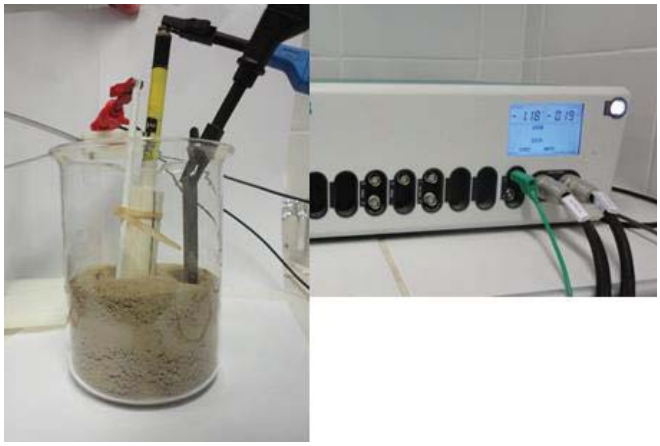


Figure 1. Electrochemical cell and equipment to obtaining polarization curves (potentiostat).

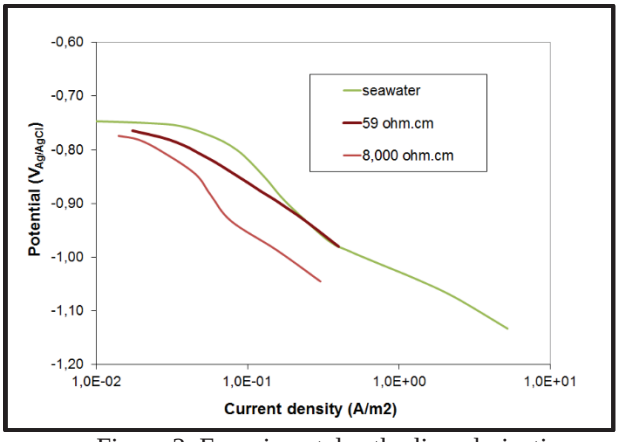


Figure 2. Experimental cathodic polarization curves obtained in laboratory.

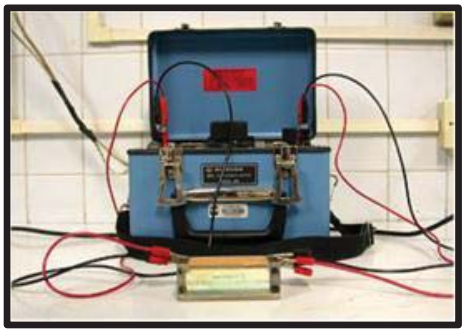


Figure 3. Resistivity measurements in laboratory using a soil box

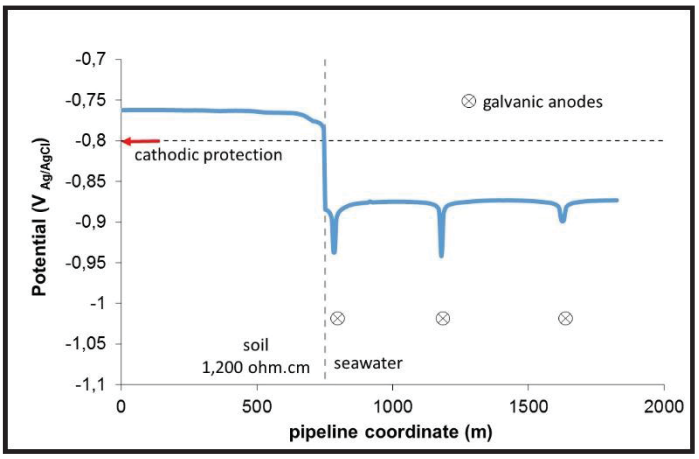


Figure 4. Influence of the resistivity on a pipeline potential distribution.

4. References

[1] *Control of Pipeline Corrosion*, 2nd Ed., Peabody, A.W. Ed. R.L. Bianchetti.

[2] Telles, J. C. F., Mansur, W. J. ; Santiago, J. A. F., "*Procat System: Recent Advances and Future Developments in Numerical Simulation of Cathodic Protection Problems*", BETECH VI, Boundary Element Technology Conference, Southampton, 1991.

[3] S.L.D.C. Brasil, D.S. Freitas, W.Baptista, J.C.F.Telles, J.A.F.Santiago, J.H.L.Oliver, "*Potential Distribution on Cathodically Protected Buried Pipelines with Aged and High Performance Coatings*", Corrosion 2006 (NACE), USA 2006.

[4]S.L.D.C.Brasil, J.C.F.Telles, J.A.F.Santiago, L.B.S.Marques, W.G.Castinheiras Jr, J.P.K.Gervásio, L.C.Vidal, "*Internal Cathodic Protection of Formation Water Pipelines*", *Materials Performance*, vol 49, n.11, pp 32-38, Nov 2010.

[5] Brasil, S.L.D.C., Freitas,D.S., Coelho,J.F.P., Telles,J.C.F, Santiago,J.A.F., Oliver,J.H.L., "*Numerical Simulation of a Cathodic Protection System in a Controlled Field*", EUROCORR 2011-The European Corrosion Congress, Stockholm - Sweden, Sep/2011.

The Home Context Verification Module of the Metaverse Assisted Living Support System

EunJin Ko
Real & Emotional Sense Convergence Service Research Section
Electronics and Telecommunications Research Institute
Daejeon, Republic of KOREA
ejko@etri.re.kr

JongHyun Jang
Real & Emotional Sense Convergence Service Research Section
Electronics and Telecommunications Research Institute
Daejeon, Republic of KOREA
jangjh@etri.re.kr

Abstract — **As the number of the old and the handicapped to increase, it is needed to take care these closely and continuously. However, it is difficult to support those people closely because members of family didn't live together or care all day long. To meet this problem, many people lived in the nursing home or they lived alone who want to stay in their hometown without continuously care by relatives or other people.**

To solve this problem, there are many efforts to provide convenience methods of this issue. Many of them tried to gather all kinds of data from home devices and sensors, filtered row data and made a decision what was happened at home. After decision, some sort of systems notified other people who were normally related with the monitored person the situation of monitored house and let them respond the situation. The related researches focus on the improvement of sensed data analysis from home devices and sensors.

However, the analysis of house situation with sensed data is not covered with the house situation because of several reasons: sensing data accurateness, model definition of house situation decision and a difference home living pattern of each person. So we turn the eye on the home living pattern which is built by a person who knows well the living pattern and is corrected by sensed data and respond data to make up for the house situation decision quality.

With this subject, this paper presents the Metaverse Assisted Living Support System (MALSS) to support the old who should need a help from relatives or others. In MALSS, we try to combine sensed data and home living pattern to find out the house situation and provide easily user interface (UI) with cyberspace technology for easily service access.

Keywords— Assisted Living Care, Metaverse, Device Virtualization, Remote Monitoring, IT Remote Helper, Living Pattern Analysis

I. INTRODUCTION

According to a prediction report of National Institute on Aging, National Institutes of Health and U.S. Department of Health and Human Service, the population of age 65+ is over than age <5 in 2030 and it is the default option to face this phenomenon let us keep to care the old and the assisted people[1]. In the fast, the old lived with their younger members of family and the members of family cared each other. However, the old lived alone recently and they didn't expect the family care anymore. So, there is a need to monitor and assist from relatives or related people remotely.

The best solution to care the old at home is a assistant person who lives in the same house, watches and supports him/her. But this is a dream. To overcome this problem in IT industry, there are many home automation services including sensors, devices and monitoring solutions. With these services and systems, it is tried to care the old or the handicapped remotely in the parts of the home automation [2].

In these approaches, there are several preconditions. The first is the trust level of sensing data. To gather more accurately sensing data, many home solution companies develop diverse sensors [3] and they try to verify what happens in the monitoring house with the sequence and the relation of sensing data. However, the analysis of house situation with sensed data is not covered with the house situation and we turn the eye on the home living pattern which is built by a person who knows well the living pattern and is corrected by sensed data and respond data. The second is that all participants should know the functional mechanisms of all devices and solutions and deeply understand why these services are needed [4]. Unfortunately, many of cared peoples didn't familiar to these services well and some of them turned off or broken devices and told the service providers that the functions of devices are hard to understand.

So, we have developed the MALSS which provides remote house monitoring, gathers sensed data, makes a decision of house situation with sensed data and provides 3D UI like Second Life [4]. A virtual world is an instantiation of

a metaverse and a 3D virtual space in which people interact with one another through avatars and metaverse client. This metaverse resembles the actual world but without its original limitations [5].

II. The Home Context Verification Module of the Metaverse Assisted Living Care System

The current researches had concentrated to know the house situation with sensors and devices and the result of this process couldn't produce a good result because sensing data sets are partial information [6]. To overcome this problem, we consider the living pattern survey and response result of assisted people who are living in the other counties or cities.

We define the House Living Pattern Types as shown in the table I. While defining the types, we are mapping the living types and sensed data sets which are gathered during living pattern type durations. After defining the living pattern types, we make a UI to get the real living pattern information from the relatives or family numbers who know the living pattern of the monitored people to make up for the error rate of house context aware decision with comparing real living pattern information timestamp and living pattern types derived from the sensed data set analysis.

TABLE I. House Living Pattern Types

Living Pattern Types	Code	Content	Devices	Location	Event Counts
WakeUp	100	Alarm on	Watch/Cell Phone	Room	Alarm Counts
		Light On	Light	Room->Livingroom->	
		IR Sensor	IR Sensor	Bathroom or Kitchen	IR Sensor Ordering
		Faucet on/off	water debit sensor	Bathroom/Kitchen	Counts
		Gas Heater on	Gas Heater	Livingroom	
Dinning	101	Faucet on/off	water debit sensor	Kitchen	
		Refrigerator open/close	Refrigerator		
		Cooker On	Cooker		
		Gas range (Gas valve Open)	Gas range		
		Microwave On/off	Microwave		
		Dinning Table	IR Sensor		
		Light On	Light		
		TV On	TV	Livingroom	
		AirCondition/Gas Heater On	AirCondition		
TV Watching	102	Livingroom Sofa	weight sensor	Livingroom	
		Light On	Light		
		TV On	TV		
Dish-washing	103	Faucet on/off	water debit sensor		
Cook	110	**Refrigerator open/close**	Refrigerator	Kitchen	
		Faucet on/off	water debit sensor		
		Microwave On/off	Microwave		
		Gas range (Gas valve Open)	Gas range		
		Light On	Light		
Sleep	112	Light Off	Light	Anywhere	
		TV off	TV	Livingroom	
		gate close	door	Entrance	
		Bed	weight sensor	Room	
		Window close	Windows	Balcony, Room	
Taking Medicine	200	**medicine bottle On/Off**	medicine bottle	medicine bottle location	
		Weight Sensor			

When the MALSS receives sensed row data, the system save these data in the sensing buffer memory. The preprocessing verification module retrieves sensed row data from buffer and makes living pattern code using pre-defined mapping table shown Table I with the continuous sensed data. In this step, the system couldn't get all sensed data sets because home living status and device operations don't work routinely every day and we have to weight device priority, sensed data types and value range. The system should have to preprocess to find out the house situation with gathered row data only.

With the priority of devices, data types and data value level, the House Context Verification Module gets the time vale of pattern code and compares it with the time zone of the same living pattern code in the real living pattern information provided by related people. If there is time line mismatch of living pattern type, the MALSS notifies the alarm of abnormal house context event code to the group of related sites and people, requires them to check the house situation and tell them to callback process whether the abnormal event is really happened or not.

Without the real living pattern information, the MALSS puts 61 % of average hit-ratio to verify the house situation and the MALSS gets 84 % of hit-ratio with the real living pattern information provided by related people who know well the monitored house.

III. Conclusions

In this paper, we look around the concept of assisted living care approaches and the metaverse service at first. To provide assisted living care service based on the IT technology, we define the living pattern types and design the model of house situation definition model which makes home context-aware events with the analysis of sensed data and comparison of the time table of house living pattern.

After testing the proposed system, we get the 84 % accuracy of the House Context Verification module of the MALSS compared with the 61% of accuracy based on the analysis of sensed data only.

In the further works, we will define over 10 house living patterns to monitor house situation more exactly and consider the abnormal cases of the analysis of sensed data. Also, we will focus on the virtualization of household electric appliances to provide the easy access way the old or the handicapped with simple click activity.

Acknowledgment

This work was supported in part by the IT R&D programs of The Ministry of Science, ICT and Future Planning (MSIP) [10043430, Development of Metaverse based Collaborative Family Safety Service Virtualization Technology for Remotely Resident Family Members].

References

[1] National Institue on Aging, National Institutes of Health and U.S. Department of Health and Human Services, "Global Health and Aging", Oct. 2011.

[2] Daniel Castellano, Jose Maria Canas, "Home automation system with web interface in the JdeRobot framework, June 2014, XV Workshope of physical agents.

[3] SmartThings, www.smartthings.com.

[4] Eunjin Ko, Jonghyun Jang, "The Virtual Device Managing Module of the Metaverse Assisted Living Support System", SAM 2014, pp 125 - 126, July, 2014.

[5] Sangwook Park, Noh-Sam Park, Jung-Tae Kim and Eui-Hyun Paik, "Provising of the Express Multisensory Adaption Platform for Heterogeneous Multimedia Devices in the Ubiquitous Home, pp 126-131, Feb. 2009

[6] Sven Meyer, Andry Rakotonirainy, "A Survey of Research on Context-Aware Homes", ACSW frontiers 2003

Numerical Investigation and Optimization of Modern Composites for the Protection against Different Threats

Arash Ramezani and Hendrik Rothe
Department of Measurement and Information Technology,
University of the Federal Armed Forces, Hamburg, Germany

Abstract - *Terrorist attacks are a major threat to civil and military structures. Threats such as sniper fire or the infamous improvised explosive devices (IEDs) have driven a rapid evolution of armor structures. To optimize weight, ballistic resistance, and costs, materials must be the main focus of attention, in particular composite armor structures. These are made up of at least two different materials, resulting in a material system with improved or completely different properties compared to the individual materials.*

Keywords: numerical optimization; modeling methodologies; CAE systems; composite armor structures; terminal ballistics

(This paper is being submitted as a poster)

1 Introduction

During an explosion, the peak pressure produced by a shock wave is much greater than the static collapse pressure. To deal with these kinds of problems which involve the release of a large amount of energy over a very short period of time, e.g. explosions and impacts, there are three approaches. Each of them underlies the problem that the processes are highly non-linear and require information regarding material behavior at ultra-high loading rates, which is generally not available. Hence, the first approach may cause tremendous expenses, as it involves the actual ballistic tests which are mostly experimental. Besides the actual tests, analytical approaches are possible if the geometries involved are relatively simple and if the loading can be described through boundary conditions, initial conditions, or a combination of the two. Finally, numerical solutions are far more general in scope and remove any difficulties associated with geometry. They have become available in the form of advanced computer programs called hydrocodes. These hydrocodes apply an explicit method and use very small time steps for stable results. Moreover, they make use of a set of equations, the equations of state (EOS), which relate the density or volume and internal energy and pressure which emanate from the material by applying the principles of conservation of mass, momentum, and energy.

This work will evaluate the effects of modern composites with numerical simulations. Due to the fact that all engineering simulation is based on geometry, the design, target, and all of its other components are simulated as CAD models. Using ANSYS as a CAD-neutral environment that supports bidirectional, direct, and associative interfaces with CAD systems, the geometry can be optimized successively. Beyond that, a brief overview of ballistic tests will be provided to offer some basic knowledge of the subject, which is needed to understand the concluding comparison between the simulation and ballistic test results.

The objective of this work is to develop and improve the composite armor systems which are deployed in the security sector. Instead of running expensive ballistic tests, numerical simulations should identify the vulnerabilities of items and structures.

By progressively changing the material parameters, the armor will be optimized. For testing new materials, a nonlinear extrapolation is used across the yield region to obtain the value of stress at the yield point that is used to fit Ludwiks law to the mechanical data. Using a sensitivity analysis, information regarding decisive variables is obtained and vulnerabilities are easily found and can be eliminated afterwards. The structure can be improved by varying the thickness of the different layers. The method of steepest descent is used to find an optimal solution for decelerating the projectiles.

Therefore, several steps need to be taken to achieve the best possible result: from modeling to calculation to the evaluation and subsequent improvement of the model.

In this study, different methods for the simulation of composite armor structures will be introduced. In so doing, the possibility of coupling various solver technologies will be discussed and illustrated by means of an example. For the first time, glass laminates will be modeled using coupled methods. Techniques previously applied, show considerable shortcomings in portraying the crack and error propagation in the glass. Mesh-free approaches, in turn, do not correctly present the behavior of synthetic materials. To overcome the shortcomings of these single-method approaches, this work will present an optimal solution to the problem by combining the methods.

2 Methods of Space Discretization

The spatial discretization is performed by representing the fields and structures of the problem using computational points in space, usually connected with each other through computational grids. Generally, the following applies: the finer the grid, the more accurate the solution. For problems of dynamic fluid-structure interaction and impact, there typically is no single best numerical method which is applicable to all parts of a problem. Techniques to couple types of numerical solvers in a single simulation can allow the use of the most appropriate solver for each domain of the problem. The most commonly used spatial discretization methods are Lagrange, Euler, ALE (a mixture of Lagrange and Euler), and mesh-free methods, such as Smooth Particles Hydrodynamics (SPH).

3 Ballistic Trials

Ballistics is an essential component for the evaluation of our results. Here, terminal ballistics is the most important sub-field. It describes the interaction of a projectile with its target. Terminal ballistics is relevant for both small and large caliber projectiles. The task is to analyze and evaluate the impact and its various modes of action. This will provide information on the effect of the projectile and the extinction risk.
In order to develop a numerical model, a ballistic test program is necessary. The ballistic trials are thoroughly documented and analyzed – even fragments must be collected. They provide information about the used armor and the projectile behavior after fire which must be consistent with the simulation results.
In order to create a data set for the numerical simulations, several experiments have to be performed. Ballistic tests are recorded with high-speed videos and analyzed afterwards. The experimental set-up is shown in Figure 1. Testing was undertaken at an indoor ballistic testing facility. The target stand provides support behind the target on all four sides. Every ballistic test program includes several trials with different glass laminates. The set-up has to remain unchanged.

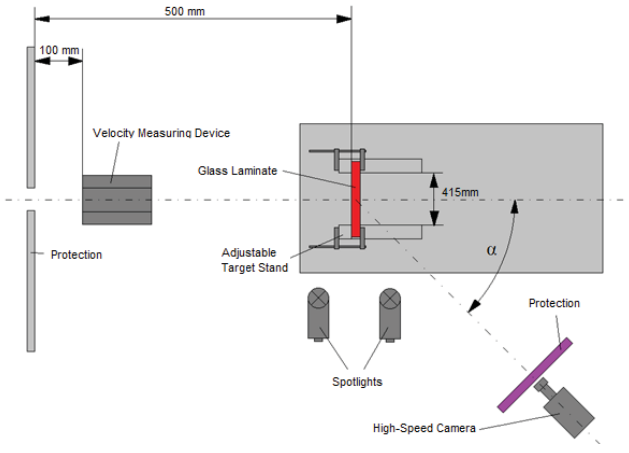


Figure 1. Experimental set-up.

4 Numerical Simulation

The ballistic tests are followed by computational modeling of the experimental set-up. Then, the experiment is reproduced using numerical simulations.
Numerical simulation of composite armor structures requires the selection of appropriate material models for the constituent materials and the derivation of suitable material model input data. The comparison is presented in this paper.
Before the evaluation starts, it has to be noticed that the Euler method is not suitable for numerical simulations dealing with brittle materials. It is generally used for representing fluids and gases, for example, the gas product of high explosives after detonation. To describe solid behavior, additional calculations are required. Cracking cannot be simulated adequately and the computation time is relatively high.

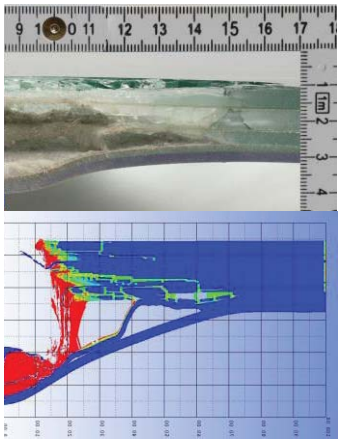


Figure 2. Comparison between simulation results and ballistic trial.

With the coupled multi-solver and optimized material parameters, the simulation results adequately mirror the observations made in the ballistic experiments. Fragmentation and crack propagation are almost equal to the ballistic test shown in Figure 2.

5 Conclusion

This work demonstrated how a small number of well-defined experiments can be used to develop, calibrate and validate solver technologies used for simulating the impact of projectiles on complex armor systems.
Existing material models were optimized to reproduce ballistic tests. High-speed videos were used to analyze the characteristics of the projectile – before and after the impact. The simulation results demonstrate the successful use of the coupled multi-solver approach. The high level of correlation between the numerical results and the available experimental or observed data demonstrates that the coupled multi-solver approach is an accurate and effective analysis technique.
The gained experience is of prime importance for the development of modern armor. By applying the numerical model a large number of potential armor schemes can be evaluated and the understanding of the interaction between laminate components under ballistic impact can be improved.

GPS Positioning Algorithm, its Errors and Solution

Mohammed Abdi [1], **Hohite Fetene**[2], **and Naima Naheed** [2]
[1]Department of Mathematics and Computer Science, Benedict College, Columbia, SC, USA
[2]Department of Physics and Engineering, Benedict College, Columbia, SC, USA

Abstract - *GPS, space-centered satellite with 24 basic satellites carrying atomic clocks navigation that deliver location and time information prudently. If three satellites are available, three spheres are known whose intersection consists of two points. The problem is to solve the three sphere equations. The receiver clock is not perfectly in sync with the satellite clock. To solve the inaccurate timing one more satellite need to be added, which brings a prodigious difference by several kilometers on the positioning. Two further problems arise when GPS is deployed, the conditioning system of equations and the transmission speed of the signals. For accuracy, we increased the satellites from four to seven. Our goal was to solve the least squares system of seven equations in four unknown variables (x, y, z, d) using Gauss-Newton iteration method. We used two types of satellites, tightly and loosely bunched. Results indicated that system becomes ill-conditioned when satellites are bunched closely in the sky.*

Keywords: GPS Errors, Least Squares Systems, Gauss-Newton Iteration Method

1 Introduction

GPS (Global Positioning System) contains 24 satellites carrying atomic clocks which orbit the earth at an altitude of 20,200 km [4]. These satellites transmit carefully synchronized signals from space to GPS receivers on earth. These receivers take the signals and translate them via arithmetic and algebra to determine the (x, y, and z) coordinates of the receiver. Discussion of GPS terminology and positioning algorithms can be found, for example, in [4], [9], and [12].At a given prompt, the receiver gathers the signals from a satellite and determines the transmission time (t), the aggregate time between the signal and the receiver. In theory, the speed of the signal is accurately taken as the speed of light. To find the receiver to the satellite multiply the transmission time by the speed of light, putting the receiver on the surface of the sphere centered at the satellite position with a radius equal to the distance from the receiver to the satellite.

If three satellite are used, then there are three spheres with the receiver at their intersection. However, the clocks in the receivers incline to be of truncated exactitude which means that using only three satellites can result in position errors of several kilometers. Adding a fourth satellite to this problem fixes the error but on the other hand complicates the arithmetic form to solve the equation with four equations and four unknowns. The system of equations shown below are the simplified form of the GPS problem for four satellites. The position of each satellite in the atmosphere is denoted by (Ai, Bi, Ci) in which the transmission time is denoted by $\mathbf{t}_i$. "d" is defined to be the difference between the synchronized time on the four satellites and the receiver clock. Then the intersection of the spheres, denoted as (x, y, z),

- r1(x, y, z, d)= $\sqrt[2]{(x-A1)2+(y-B1)2+(z-C1)2}$−c (t1−d)=0
- r2(x, y, z, d)= $\sqrt[2]{(x-A2)2+(y-B2)2+(z-C2)2}$−c (t2−d)=0
- r3(x, y, z, d)= $\sqrt[2]{(x-A3)2+(y-B3)2+(z-C3)2}$ −c (t3−d)=0
- r4(x, y, z, d)= $\sqrt[2]{(x-A4)2+(y-B4)2+(z-C4)2}$−c (t4−d)=0

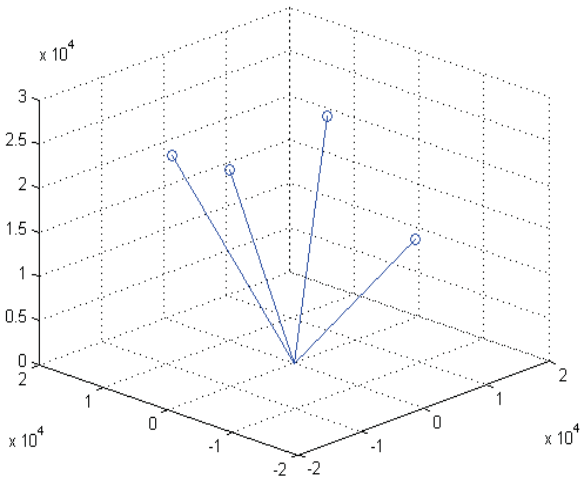
2 Solution for the system of equation

We tried to solve the system of equation by using both Multivariate Newton's Method as well as algebra, the results became accurately same on both cases. Using Multivariate Newton's Method, to find the receiver's position (x, y, z) near the earth and difference between the coordinated time on the (four) satellite clocks and the earth-constrained receiver clock d for known, simultaneous satellite positions (15600, 7540, 20140), (18760, 2750, 18610), (17610, 14630, 13480), (19170, 610, 18390) in Km, and measured time intervals 0.07074, 0.07220, 0.07690, and 0.07242 in seconds, respectively. We then set the initial vector to be (x0, y0, z0, d0) = (0, 0, 6370, 0). The approximate results are (x, y, z) = (-41.77271, -16.78919, 6370.0596) kms and d = - 3.201566 * 10-3 seconds. And using algebra, first we removed the quadratic variables by subtracting the rest of the equations from the first one. Next we are left with three equations with four unknowns, the quadratic variables will be cancelled. Then in terms of d we solved x, y and z by using the determinant equations. The two coordinates found will be (x, y, z) = (-41.772709570837364,-16.789194106526850, 6370.059559223344) and (-39.747837348142937, -134.2741443606658, -9413.624553735754). Then in order to distinguish which one to choose, we calculated the nom of each coordinates. The nom of the first coordinates is

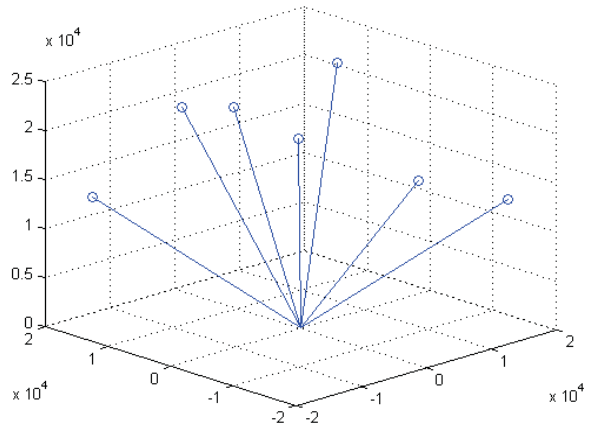
6370.218648080797 and the second is 9414.666041613707, but only the first one is closely related to the radius of the earth. So the first coordinates is the solution and d = -0.003201566 (- 3.201566 * 10-3) seconds.

3 Testing the conditioning of the GPS problem

In order test the satellite conditioning we first defined the satellite positions (Xi, Yi, Zi) from the spherical coordinates **(ρ, ϕ, θ_i). $X_i=\rho\cos\phi_i\cos\theta_i$, $Y_i=\rho\cos\phi_i\sin\theta_i$, $Z_i=\rho\sin\phi_i$.** Where ρ=26570 kilometers is fixed while $0 \leq \phi_i \leq \pi/2$ and $0 \leq \theta_i \leq 2\pi$ for i= 1… 4 are chosen arbitrarily. The ϕ coordinate is restricted so that the four satellites are in the upper hemisphere**.** For the initial vector (x0, y0, z0, d0) = (0, 0, 6370, 0.0001). We defined t= d + Ri/c, the satellite ranges to be $R1=\sqrt[2]{Ai2 + Bi2 + (C1 - 6370)2}$ and calculated the error magnification factor of by dividing change in position by the speed of light and multiplying it by the change in transmission time. The atomic clocks are accurate up to 10 nanoseconds. The results found for 4 satellites with $\phi=[3\pi/8,\pi/4,3\pi/8,\pi/4]$ and $\theta=[\pi/8,\pi/2,\pi,3\pi/2]$ defining the satellite's positions is:



Next we added 3 more satellites to the previous 4 satellites to reduce the position error and the condition number. We used Gauss Newton method to solve the system of equation which is as follows: The initial factor values of (x, y ,z, d)= (0, 0, 6370, 0.0001), $\phi=[3\pi/8,\pi/4,3\pi/8,\pi/4,\pi/6,\pi/3,\pi/6]$ & $\theta=[\pi/8,\pi/2,\pi,3\pi/2,3\pi/4,5\pi/4,7\pi/4]$



4 Conclusions

We concluded that by adding more number of satellites to the system, the average change in position becomes very smaller compared to only using four satellites in our case, and significantly smaller than using four clustered satellites. The maximum error magnification factor we happened to find with seven satellites is close to the minimum found with four satellites initially. As we have seen the condition number of the seven satellites is much smaller than using four satellites (Condition No of seven satellites < Condition No of four satellites). Clearly when we use seven clustered satellites would produce the smallest error. So from this we can deduce that by adding more satellites the error can be reduced as much. As a conclusion more satellites reduces the error by thousands of meters.

5 References

[1] J. Farrell and M. Barth, The Global Positioning System and Inertial Navigation: Theory and Practice. New York: McGraw-Hill, 1998.

[2] E. Kaplan, Ed., Understanding GPS: Principles and Applications. Boston, MA: Artech House, 1996.

[3] B. Parkinson, P. Axelrad, and P. Enge, Eds., Global Positioning System: Theory and Applications: AIAA, 1996.

[4] Rai, Anil. INTRODUCTION TO GLOBAL POSITIONING SYSTEM (n.d.): n. pag. Web

SESSION

LATE BREAKING PAPERS:MODELING, VISUALIZATION, AND SIMULATION

Chair(s)

TBA

Modeling and Simulation of the Continuous Time Discrete Space Korteweg-de Vries Equation

Louiza Sellami
Electrical and Computer Engineering Department
US Naval Academy
Annapolis, Maryland, USA
sellami@usna.edu

Robert W. Newcomb
Electrical and Computer Engineering Department
The University of Maryland
College Park, Maryland, USA
newcomb@umd.edu

Abstract— ***A discretized version of the well-known Kortweg-de Vries equation is here derived to yield a set of discrete equations. These equations represent a system of nearest-neighbor coupled oscillators that form a cascade of Fermi-Pasta-Ulam lattices. Signal flow diagram of a single lattice is derived, and a cascade of thirty of these lattices is simulated using Simulink. Simulation results and relation to the traveling wave transistor are included and discussed.***

Keywords*: Korteweg-de Vries equations, Fermi-Pasta-Ulam problem, lattice, Solitons, Simulink, nonlinear differential equations, hyperbolic solutions.*

1 INTRODUCTION

One of the most studied nonlinear partial differential equation in mathematical physics is the Korteweg-de Vries equation (KdV) which arises in the study of a number of different physical systems. Historically, it was first formulated as part of the modeling of shallow water waves with weakly non-linear restoring forces [1], but subsequently it has been found to have connections to a wider range of physical phenomena, especially those exhibiting traveling waves and solitons [2-3]. Solitons or solitary waves are large amplitude excitations with finite spatial widths. They are found in certain nonlinear systems, and can propagate over long distances without changing their shapes even when they interact with other waves. The KdV equation has been studied and applied for decades, and many analytical, series-approximation, and numerical solutions have been developed for particular situations, some of which we allude to in the next section.

In addition to being the governing equation of the string in the Fermi–Pasta–Ulam (FPU) problem in the continuum limit, it approximately describes the long time evolution of one-dimensional waves in many physical settings, including ion acoustic waves in a plasma and acoustic waves in a crystal lattice. From detailed studies of properties of the equation and its solutions, the concept of solitons [3] was introduced and the method for exact solution of the initial-value problem using inverse scattering theory was developed [4]. Many extensions of these ideas to a number of other nonlinear evolution equations of physical interest and to other classes of equations have been developed [5].

In this paper, following a treatment in [6], we derive a discrete version of the KdV equation and formulate the set of discretized equation in terms of lattices. A Simulink model of a lattice is developed, and a system of 30 lattices is then simulated in Simulink for a particular set of inputs and parameters. The numerical solutions generated by the Simulink model are further examined and interpreted.

The paper is organized as follows. In the first section we present the KdV equation in its original form. In the second section we derive the discrete versions from the FPU one dimensional string equation. Section three is devoted to the formation of lattices and signal flow diagrams for the Simulink model. In section four we present the simulations results, and finally in the conclusion, we discuss the significance of these results, as well as their relation to traveling wave transistors.

2 THE KDV EQUATION

The KdV family of equations is a one-dimensional equation (that is one spatial dimension plus time) nonlinear partial differential equation of the third order, and which takes the general form in (1.1), using the

short denotations as $u_t(x,t)=\frac{\partial u(x,t)}{\partial t}$; $u_x(x,t)=\frac{\partial u(x,t)}{\partial x}$; $u_{xx}(x,t)=\frac{\partial^2 u(x,t)}{\partial x^2}$; and $u_{xxx}(x,t)=\frac{\partial^3 u(x,t)}{\partial x^3}$

$$u_t(x,t)+u_{xxx}(x,t)+P(u)u_x(x,t)=0 \quad (1)$$

where *u(x,t)* is a function of space and time, and *P(u)* is some polynomial of *u(x,t)*. One can place various normalizing constants in front of the $u_{xxx}(x,t)$ and *P(u)* terms, but they can usually be scaled out. The function *u* and the polynomial *P* are usually assumed to be real. This equation represents the simplest case of only one space variable *x*. The solution, *u(x,t)*, to this equation describes the elongation of a wave a location *x* and time *t*.

KdV equation is completely integrable, and gives rise to an infinite number of solitonic solutions. Exact analytical solutions to the KdV equation exist in the literature, depending on the nature of the polynomial *P*. For instance, Vvedenskii in [7] derived general hyperbolic solutions without initial nor boundary conditions for *P(u)* = 6*u(x,t)*, by means of elementary operations as well as by using Backlund transform. Davenport, on the other hand, derived hyper-complex solutions in [8] using the Cauchy-Reimann equations. In one general solution, he found he could implicitly model traveling wave trains, shock waves, and solitons, depending on the selections of the three constants of integrations. Others such as Nijhoff have looked at solving the discrete time version of the KdV equation [9].

Figure 1 shows the plot of a soliton like wave solution to the Kdv equation (1) which is an hyperbolic secant solution given as [2]:

$$u(x,t)=a\sec h^2\left(\frac{x-vt}{\xi}\right) \quad (2)$$

Where *v* is the velocity of the wave, *a=3(v-1)* its amplitude, and $\xi=\frac{2}{\sqrt{v-1}}$ its spatial extent. The plots of Figure 1 are obtained for *v= 5*, and t=0, 2.5 and 5.

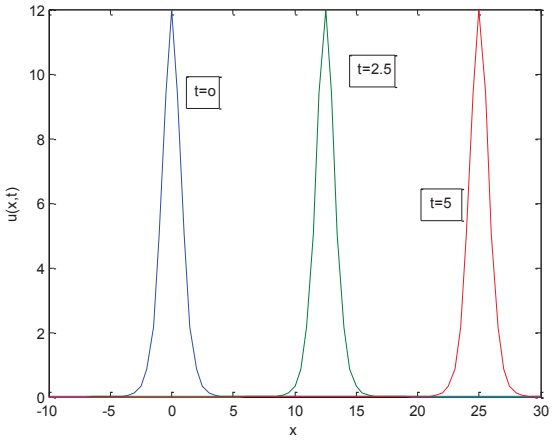


Figure 1: A soliton-like wave solution to the KdV equation (1) plotted at different times.

3 DISCRETE SPACE KDV SYSTEM: FPU LATTICE

The FPU lattice system is a non- linear system consisting of chain of equal masses connected by identical springs. The springs are not the usual Hook's Law type, but include nonlinear components in the relationship between deformation and restoring force. Since the strings are vibrating, this is equivalent to a system of nearest-neighbor coupled oscillators. Following the treatment in [6], we consider a system of *N* oscillators representing a string of length *L*, with equilibrium positions $p_j=jh,\ j=0,1,\ldots,N-1$ where $h=\frac{L}{(N-1)}$ is the lattice spacing. The position of the jth oscillator as a function of time is defined as $X_j(t)=p_j+x_j(t)$ where $x_j(t)$ is the displacement from the equilibrium position as a function of time. Writing Newton's second law, with a quadratic restoring force of the form $F(x)=kx+\beta x^2$ (note the first term is Hook's force), for the discrete system yields:

$$m\frac{d^2x_j(t)}{dt^2}=k(x_{j+1}(t)+x_{j-1}(t)-2x_j(t))[1+\alpha(x_{j+1}(t)-x_{j-1}(t))] \quad (3)$$

Upon introducing the wave speed $v=\sqrt{\frac{K}{\rho}}$, where $K=\frac{k}{h}$ is the Young's modulus for the string, and $\rho=\frac{m}{h^3}$ the density, equation (2) can be rewritten as follows:

$$\frac{d^2x_j(t)}{dt^2}=\frac{v^2}{h}\left(x_{j+1}(t)+x_{j-1}(t)-2x_j(t)\right)[1+\alpha\left(x_{j+1}(t)-x_{j-1}(t)\right)] \quad (4)$$

This equation expresses the dynamics in time of the displacement from equilibrium position of the j[th] oscillator as a function of the displacements of its neighbors. We point out that it has been demonstrated that, in the continuum limit, equation (3) maps onto the KdV equation. The details of the proofs are presented in [6] and [10].

4 SIMULINK MODEL

We take equation (4) as the basis for the Simulink model, for which, as a first step, we implement the j[th] lattice section as shown in Figure 2. The model shows the dynamics of the continuous time system with the input signals being $x_{j+1}(t)$ and $x_{j-1}(t)$ labeled as In3 and In2, respectively. The constant input α is labeled In1, and the model integrates twice to produce the output $x_j(t)$. Also, for simplicity, the system is normalized so that the term $\frac{v^2}{h}$ is one.

5 SIMULATION RESULTS

The full system that is simulated is composed of 30 lattice sections. It is shown in Figure 3 as a cascade of six subsystems, each of which models five lattice sections of the type of Figure 2. Here the constant input α is set to two for all sections. Additionally, specific boundary conditions are set for the first and last lattice sections. In particular the input signals for the first section are pulse like signals, and for the 30[th] section, the term $x_{j+1}(t)$ is set to one. The output signals of the first and last subsystems are plotted in Figures 4 and 5, respectively. Note that the equilibrium position for the

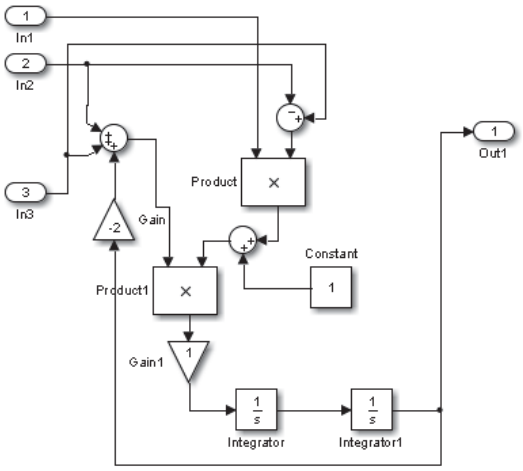


Figure 2: Simulink model of the j[th] FPU lattice section.

displacement of the 5[th] lattice is zero and that of the 30[th] lattice is 1. Based on these graphs, which represent the displacements for the 5[th] and 30[th] lattice sections, the system exhibits a quasi-periodic behavior, in that it regularly recovers its initial state. It is worth noting that the system does not relax into a state of equipartition, whereby the influence of the initial modes of vibration subsides, leading to a system with equally likely modes of vibration. The simulation results are consistent with the behavior expected of a FPU system.

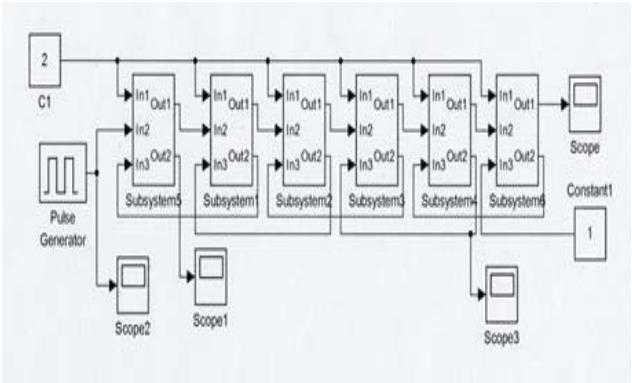


Figure 3: Simulink Schematic of six subsystems each containing five lattice sections for a total of 30 sections.

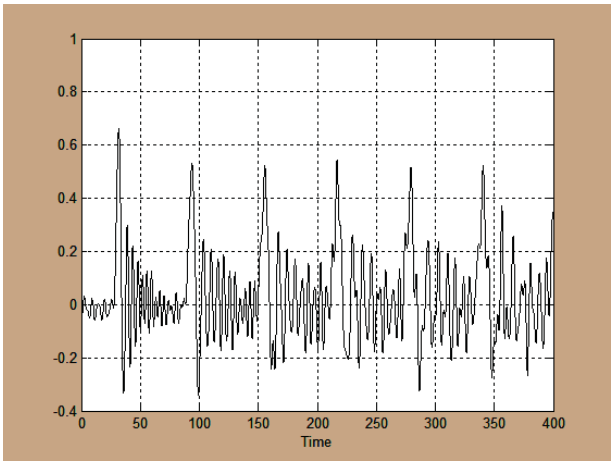


Figure 4: Output signal of first subsystem, or 5th lattice section. Displacement is shown on the Y-axis, time on the X-axis.

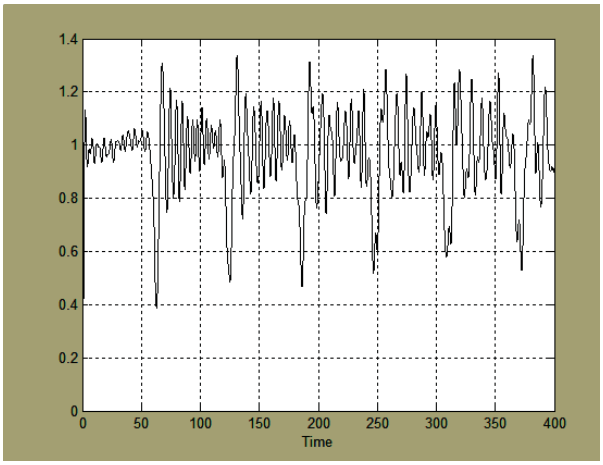


Figure 5: Output signal of the last subsystem, or 30th lattice section. Displacement is shown on the Y-axis, time on the X-axis.

6 CONCLUSION

In this paper, we presented a discretized version of the KdV equation in terms of a system of nearest-neighbor coupled oscillators that form a cascade of FPU lattices. Signal flow diagram of a single lattice is derived, and a cascade of thirty of these lattices is simulated with Simulink. Simulation results are found to be consistent with the behavior of a FPU system, which generates quasi-periodic modes of vibrations which relax back to their respective initial states. As an important application, this work could be extended to describe the line voltages in the gate and drain of nonlinear traveling-wave field effect transistors These special transistors, whose electrodes behave as a nonlinear transmission line, are used to amplify soliton like pulses [11], hereby compensating for the wave attenuation that would normally occur in a lumped transmission line.

7 REFERENCES

[1]. Korteweg, D. J. and de Vries, G, "*On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary wave*," Philosophical Magazine, Vol. 39, 1895, pp. 422-443.

[2]. Riseborough, P. S., "*The Korteweg-de Vries Equation: It's place in the development of nonlinear physics,"* Philosophical Magazine, 2010, pp.1-10.

[3]. Ablowitz, M. J. and Carlson, P. A., "Solitons, nonlinear evolution equations and inverse scattering," Lect. Notes, Vol. 149, *Cambridge University Press*, 1991.

[4]. Ablowitz, M. J., Kamp, D., Newell, A. C., and Segur, H., "*The inverse scattering Fourier analysis for nonlinear problems,"* Stud. Appl. Math. Vol 53, 1974, pp. 249-315.

[5]. Drinfeld, V. G. and Sokolov, V. V., "*Equations of KdV type and simple Lie algebras,"* Sov. Math. Dokl., Vol. 23, 1985, pp. 457-462.

[6]. Palais, R., "The Symmetry of Solitons," *Bulletin of the American Mathematical Society,"* Vol. 34, No. 4, 1997, pp. 339-403.

[7]. Vredenski, D., "Partial Differential Equations with Mathematica," *Addison-Wesley*, 1992.

[8]. Davenport, C. M., "*The general Analytical solution of the Korteweg-de Vries Equation,"* http://home.comcast.net/~cmdaven/korteweg.htm.

[9]. Nijhoff, F. and Capel, H., "*The discrete Korteweg-de Vries Equation,"* Acta Applicandae Mathematicae, Vo l. 39, 1995, pp. 133-158.

[10]. Zabusky, N. J. and Kruskal, M. D, Phys. Rev. Lett. Vol 15, 1965, p. 240.

[11]. Narahara, K and Nakagawa, S., "*Nonlinear traveling wave field effect transistors for amplification of short electrical pulses,"* IEICE Elctronics Express, Vol. 7, No. 16, 2010, pp. 1188-1194.

Virtual Wind Turbine for Augmented Reality Gearbox Inspection

Jichao Wang, John Moreland, Yihong Liu, Hao Zhang, Huizhong Pan, Yuxuan Li, Gengwei Zhang, Mengmeng Du, Chenn Zhou
Center for Innovation through Visualization and Simulation
Purdue University Calumet, Hammond, IN, USA

Abstract— *This research describes the motivations and technical approach for developing an interactive 3D virtual wind turbine and module for augmented reality gearbox inspection. Finite Element Analysis (FEA) is used to determine fatigue within the wind turbine gearbox, and the resulting model is displayed using a mixed reality interface, using a physical borescope, cardboard box with interior fiducial markers, and external display. The result leverages the benefits of existing simulation methods and emerging interactive technologies to allow users to practice using a borescope to conduct gearbox inspections.*

Keywords: Augmented Reality, Gearbox, Wind Turbine, FEA, Mixed Reality, Virtual Training

1 Introduction

Wind energy is a clean, renewable energy and it help avoid a variety of environment impacts due to its low impact emitting zero greenhouse gas emission or conventional pollutants and consuming virtually no water. Wind energy is also in sourcing jobs in the manufacturing sector that the wind energy companies employ 50500 people across construction, development and engineering. Currently the U.S. goal is to have wind energy account for 20% of the nation's electricity by 2030. To reach this goal, a significant increase in number and quality of wind energy professionals is needed. One of the barriers to meeting this demand is applying the learned concept to practical application. Students and technician will learn the concept for each component of wind turbine but they are often having trouble in understanding those knowledge by traditional method. Concerning the safety issue and high cost not every student will get opportunity to come into a real wind turbine.

To address this problem, the Department of Education's Fund for the Improvement of Postsecondary Education (FIPSE) and Purdue University Calumet have been working together to improve education in the wind energy field through the use of technology, simulation, and visualization sciences, a reformatory educational tool will be developed [1].

This paper introduces a computer software package that is developed by using 3ds Max and Unity 3D and will provide students with many of the benefits of hands-on experience working on wind turbines through virtual environment. Students and technicians can using mouse and keyboard to operate like in real world. This simulator will bring more expressive method in training area instead of traditional teaching, which is high efficiency and low cost.

This simulator is being expanded to include a module for gearbox simulation. Modern equipment design methods make use of Computer Aided Engineering (CAE) software and computational numerical analysis such as Finite Element Analysis (FEA) to obtain a high degree of precision during the design process. FEA is the method that is used in the project to structurally analyze a general gear box and predict the life of a wind turbine. The result of the stress and fatigue analysis is presented using Augmented Reality (AR), using a set of fiducial markers inside a cardboard box for the borescope camera to recognize the current location and display the interior models as if the user was inspecting a real gearbox.

Augmented reality (AR) is a type of technology that superimposes graphics and information on top of real world imagery from a camera in real-time [2, 3]. AR has been used in various educational settings to help students and trainees understand the connection between data and the real world [4-7]. This work builds on specifically on previous efforts to combine visualization of simulated data with virtual reality and augmented reality for industrial applications [8-10].

2 Methodology

2.1 Modeling

The first step for the project is to build the model of the gear box. The figure 1 shows the result after proper mesh for the model of the gear box.

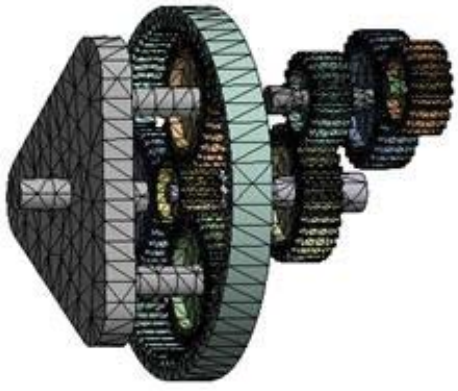


Figure 1 Gear Box Model Mesh

Throughout the project, kinds of modeling software were formally utilized to create the geometry of critical components of the wind turbine. It is important to model the standard gear based on engineering criterion and make the gears fit well with each other. This process helps to get familiar with the operating of a gearbox and the structural inside the wind turbine.

Before the analysis, the geometry were transferred into another software which is especially used for mechanical analysis. Generating mesh is the last step before conducting analysis. Generating mesh means dividing the geometry into small part to make it simple when conducting calculation. Also, mesh is the core of Finite Element Analysis, which means making the countless points of the geometry into finite element. This is a reasonable approximation to get precise and reliable result. It is asked to mesh in rational size in order to get proper result. Generally speaking, mesh generation is one of the most critical aspects of engineering simulation. Using too many cells can result in long solver runs, and too few may lead to inaccurate results. This is the core part of FEA. Then the static and dynamic analysis will be conducted on the geometry. Both static analysis and dynamic analysis are applied to get the significant data, which was made to identify the cause of stress concentration and stability disorder.

2.2 Simulation

In this part, external loads will be added in momentum to the geometry as it operates in real time. Proper constraint for every part of the model will be applied. For the static state, every part is fixed, which means the geometry cannot move. For the dynamic state, all the axis will be defined to revolve compared to ground and the contact of axis and gear is fixed. In this way, the gear can be put in motion with the revolution of the axis. The contact of the gears should be defined as frictional, which means they can match each other well when they are moving.

2.3 Interface

After post-processing in Virtual Reality Visualization, these data could be revealed intuitively as simulation results. All these results are imported into Unity3D as assets for the simulator. This is based on a methodology that has been developed to combine visualization of simulation data for interactive 3D applications [11].

Unity3D is a flexible and powerful development platform for creating multi-platform 3D and 2D interactive applications [12]. It provides users the ability to create animations, visual effect and menu systems to develop software. In Unity3D, a virtual environment for a wind farm is developed as shown in figure 2. And figure 3 shows all the components imported in the turbine nacelle.

This simulator is designed to be operational and educational for wind energy students and personnel. For the aim of developing a learning tool, the concept of the system is well designed. This simulator contains a welcome page, a control manual and a brief introduction of how to operating it to ensure its usability.

Figure 2 Virtual Environment of the Wind Farm

First, after importing 3D model in unity and matching the scale, a terrain and sky are added to build the main scene. Proper roads are added to simulate real wind farm site. Due to the recent update of unity software, the whole User Interface design is created by two method. Previously, all the UI was created by using C# language programming. Now, the new UI editor is applied to create higher efficiency menu system. All the functional work is intractable by clicking on buttons. The menu allow users to enable the nacelle render and view outside components with their definition displayed on the side. When disable the nacelle, the components inside the nacelle will be displayed.

Figure 3 All components in the Nacelle

From the programming level, there is a hierarchy tree created behind each button to control each components logical relationship. More functions are created including changing the camera view by using mouse and keyboards and highlight a component by clicking on an object body. Figure 4 shows the design of the new UI menu system.

2.4 Augmented Reality

The borescope module is designed to visualize the simulation result in the real situation of gear box inspection. The Unity simulator input is keyboard, mouse or joystick. But in this module, it can't be adopted to simulate the action of human's

hand such as stretching or pulling the wire of borescope to move the position of camera. To solve this problem, the technique of Augmented Reality is introduced. In this module, trainee moves a real borescope by hand in the training box as shown in figure 5 (a real box with necessary components inside, which is to simulate the real gearbox).

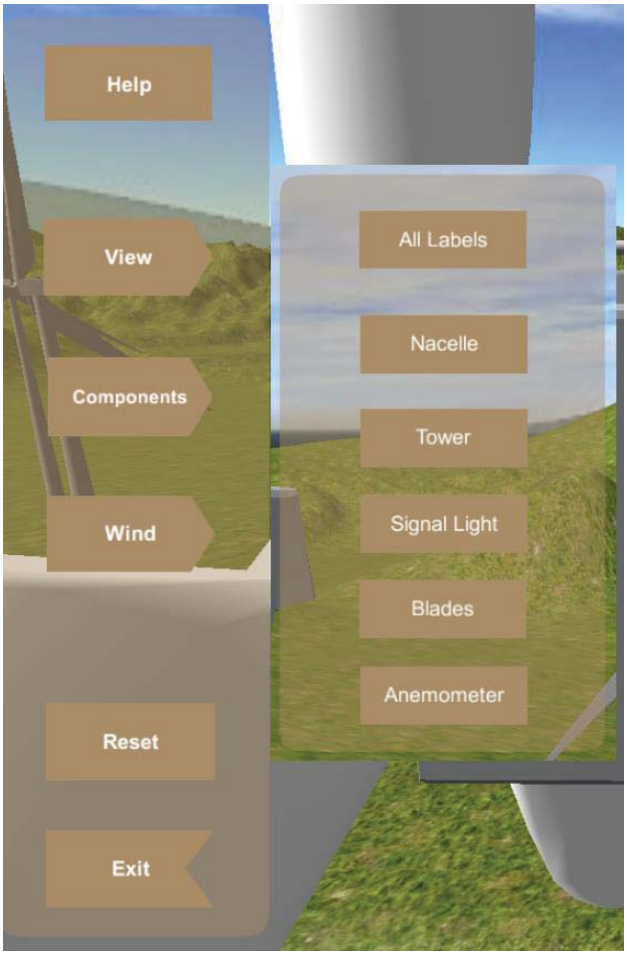


Figure 4 Menu UI

Figure 5 Training Box with recognizable patterns.

The software then collects the real-time images from the borescope camera and show the virtual sight of gearbox model with the FEA result correspondingly. The trainee has the ability to move the borescope and view both the regular inspection camera and the overall position of the scope inside the gear box. Figure 6 shows the software interface displaying the borescope camera view and AR gears overlay.

Switches for changing viewing angle and swapping FEA results are also provide in the simulator to help gain better experience while perform inspection.

Figure 6 Camera View and AR Gear Display

When using the borescope, the virtual collision system is also applied that when the virtual camera hit the gearbox body or gears it will stop moving forward even the physical camera is trying to. This provides accurate and natural operation training condition.

3 Results

The result of this project consist two parts. One part is the computer software that the user can explore and study the turbine. The other part is the gearbox module that the user controls physical borescope to perform inspection and review FEA result.

Figure 7. Main Interface of the Software

Figure 7 shows the main interface of the software. Clicking on the "Menu" button to open all sub-buttons with include "Nacelle", "Label", "Animation", "Components" and "Exit" buttons that each can toggle model rendering, trigger

information to display or animation to start. Specifically, the "Components" button opens a panel with 18 buttons. Select one button highlights its corresponding component along with a label and detailed definition.

The user can click directly on a component object to trigger the same function to highlight it and pop up the description. The keyboard and mouse are always available to fly through the environment and get close to see details of the turbine.

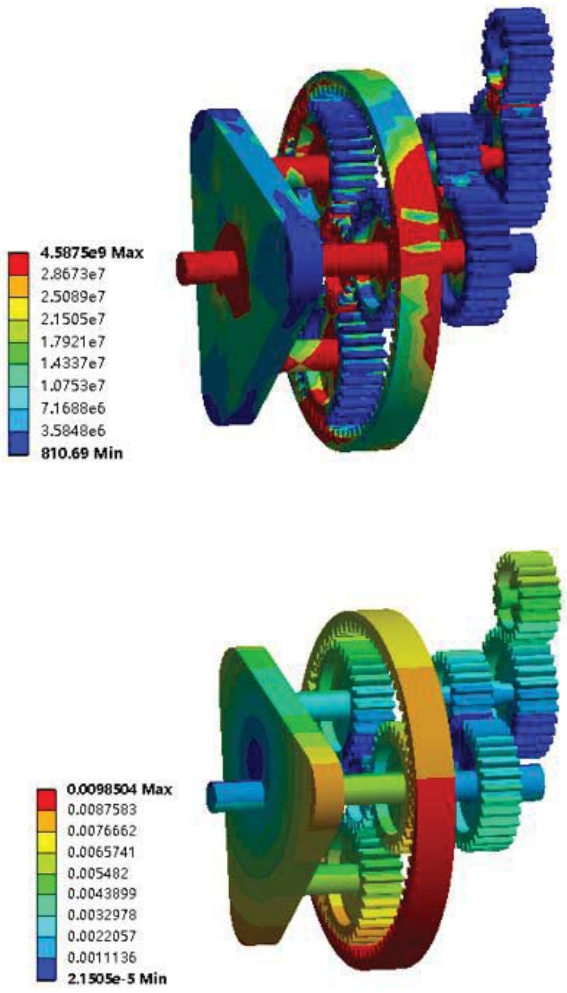


Figure 8 FEA Stress (top) and Deformation (bottom) for Static State

In the gearbox module, different FEA analysis is processed. For the static analysis, all the geometry is fixed to find the part with most stress, strain, and deformation, which will be helpful to define the dangerous part for dynamic simulation.

From figure 8, the red part of the geometry is the part under most stress and deformation. The color bar shows the value of the different part.

For the dynamic state, the geometry will rotate as it would in reality. The total rotation angle, time step as well as the rotation angle per time step will be defined before calculating.

From figure 9 the top part clearly shows that the contacting part of the two gears is under the most strain. The bottom part is showing the dynamic deformation analysis, every tooth of the gear deforms the same as it rotates. The life prediction for the most loaded part is 42535.1 cycles before being destroyed.

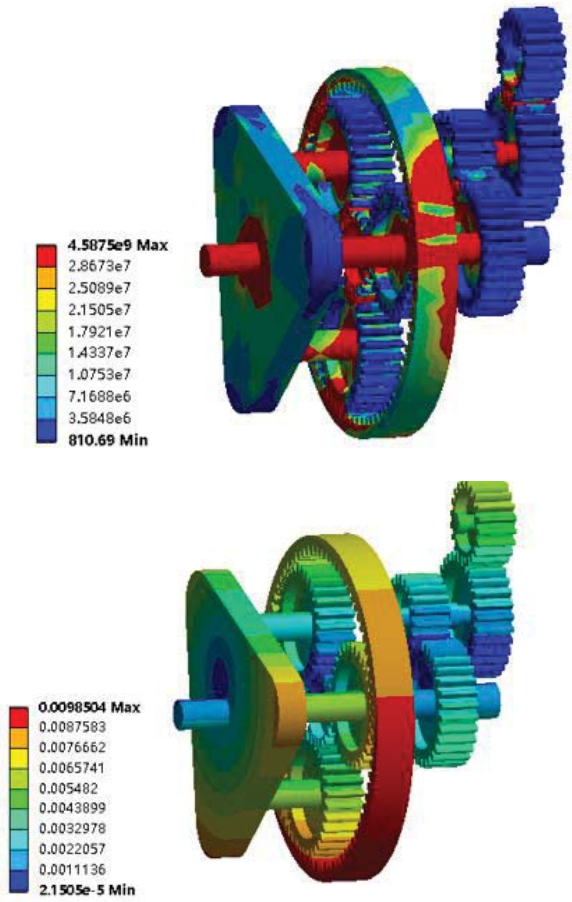


Figure 8 FEA Strain (top) and Deformation for Dynamic State (bottom)

All these FEA results are provided in the gearbox module. The virtual components are then displayed on the computer screen. Figure 10 shows the color shaded gear along with different viewing perspectives which guide the user to inspect the highlighted region for stress analysis and life prediction. By clicking on the menu, the user can select to start either keyboard control mode or AR inspection mode. More function is available including to swap the stress analysis result to deformation or fatigue base on the current inspection purpose or to switch viewing angel in the virtual environment.

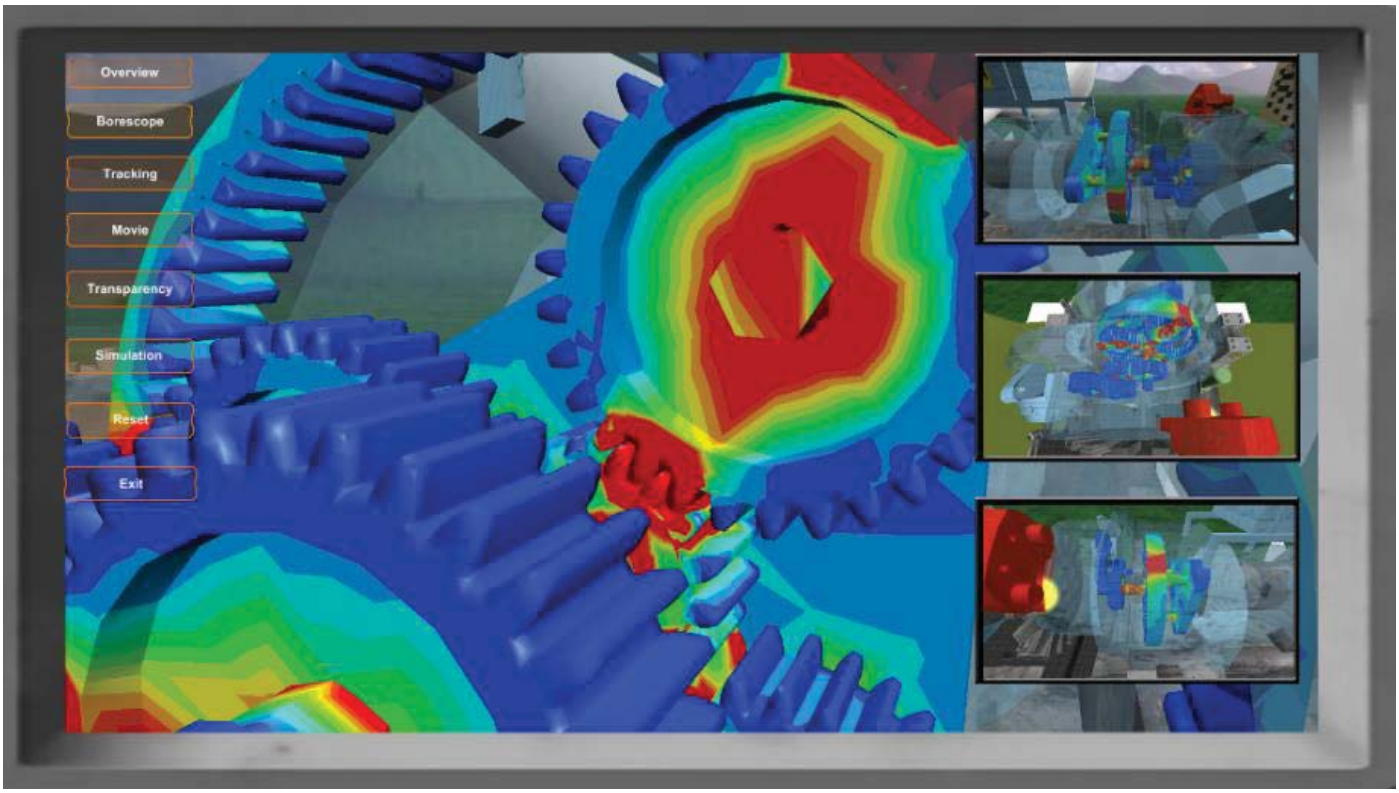


Figure 10 The gearbox inspection module shows multiple view points to help keep the user oriented during inspection.

4 Conclusions

This research has developed a method of integrating FEA simulation results of a wind turbine gearbox with an interactive 3D module and augmented reality interface for the purpose of training for gearbox inspection. The method includes numeric simulation, post processing, modeling of associated 3D structures, and registration with a physical box standing in for a gearbox. A physical borescope is used to feed the optics into the AR box, resulting in a mixed reality interface that allows users to practice gearbox inspection with a borescope.

Future work will build additional options to interface with additional data through AR, explore additional methods to integrate AR into training and assess their effectiveness.

5 Acknowledgments

The contents of this paper were developed under grant P116B100322 from the U.S. Department of Education. However, those contents do not necessarily represent the policy of the U.S. Department of Education, and you should not assume endorsement by the Federal Government.

6 References

[1] Manwell, J., McGowan, J., & Rogers, A. L. (2010). *Wind energy explained: theory, design and application.*

[2] R. T. Azuma, "A Survey of Augmented Reality," *Media*, vol. 4, no. August, pp. 355-385, 1997.

[3] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, *Recent advances in augmented reality*, vol. 21, no. 6. IEEE Computer Society, 2001, pp. 34-47.

[4] M. Dunleavy, C. Dede, and R. Mitchell, "Affordances and Limitations of Immersive Participatory Augmented Reality Simulations for Teaching and Learning," *Journal of Science Education and Technology*, vol. 18, no. 1, pp. 7-22, Sep. 2008.

[5] H. Kaufmann and D. Schmalstieg, "Mathematics and geometry education with collaborative augmented reality," *Computers & Graphics*, vol. 27, no. 3, pp. 339-345, 2003.

[6] F. Liarokapis et al., "Web3D and Augmented Reality to support Engineering Education," *Engineering and Technology*, vol. 3, no. 1, pp. 11-14, 2004.

[7] D. Schmalstieg and I. Systeme, "Geometry Education with Augmented Reality," *Technology*, no. 9225889. 2004.

[8] Chen, G., Moreland, J., Ratko, D., Jin, L., & Shen, H. (2010). Virtual Reality for Engineering Applications. *ASME World Conference on Innovative Virtual Reality.*

[9] Dekker, G., Zhang, Q., Morland, J., & Zhou, C. (2013). MARWind: Mobile Augmented Reality Wind Farm Visualization. *WorldComp 2013 International Conference on Modeling, Simulation and Visualization Methods.* Las Vegas, NV.

[10] Moreland, J., Wang, J., Liu, Y., Shen, L., Wu, B., & Zhou, Q. C. (2013). Integration of Augmented Reality with Computational Fluid Dynamics for Power Plant Training. *WorldComp 2013 International Conference on Modeling, Simulation and Visualization Methods.* Las Vegas, NV.

[11] Wang, J., Phillips, L., Moreland, J., Wu, B., Zhou, C. (2015). "Simulation and Visualization of Industrial Processes in Unity". Proceedings of Summer Simulation Multi-Conference (SummerSim'15), Chicago, IL, July 26-29, 2015.

[12] *Unity Technologies*. (2014). Retrieved from Unity 3D: http://unity3d.com

Preparation, Characterization and Kinetic Modeling of Co–Mo for Fischer–Tropsch Synthesis in a Fixed Bed Reactor

Saba A. Gheni[1], Alla Al Din H. Saleh[2], Aysar T. Jarullah [2],

[1]Chemical Engineering Department, University of Missouri Columbia, Columbia, MO, USA

[2]Chemical Engineering Department, Tikrit University, Tikrit, Salah Aldin, Iraq

Abstract

Comprehensive kinetics of Fischer–Tropsch synthesis (FTS) on a homemade CoMo catalyst in a water–gas shift reaction (WGS) was studied using a fixed bed reactor. The present model expresses the H_2 conversion a wide range of reaction conditions (523.563K, 20-25 bar, H_2/CO ratio: 2.5, and space velocity: 1000-2500 ml g^{-1} cat h^{-1}). The process is based on synthetic gas as a feed. Kinetic parameters were estimated by fitting experimental data. The catalyst was characterized by XRD, SEM and TPR. It is revealed, that the kinetic of this reaction followed power law model and the hydrogen gas affects the rate of chemical reaction more than carbon dioxide and the reaction temperature affected the chemical reaction exponentially according to the Langmuir-Hineshelwood and power law models.

Keywords*: Kinetic modeling, Fisher Tropsch, Co-Mo catalyst, fixed bed reactor*

1 Introduction

Fischer–Tropsch synthesis (FTS) process is one of the most appealing ways of ultra-clean liquid fuels and petrochemicals production. On the other hand, it is suitable for addressing the problem of environmental pollution and remote gas utilization [1]. FTS is the production of hydrocarbons from H_2/CO feed mixture and generally regarded as a promising technology for the manufacture of clean fuels due to depleting resources (crude oil) and environmental regulations. Fe and Co are the most common catalysts for the FT reaction [2]. However, there is renewed interest in finding alternative catalysts which are more resilient to sulfur poisoning and yield fuels with reduced SO_x and NO_x emissions[3].

Early transition metal carbides and particularly molybdenum carbide have been receiving significant attention as favored alternatives for different catalytic reactions, namely; NH_3 synthesis [4], hydrotreating[5-7], NO reduction[8], and methane reforming[9] Sources of feedstock's used to produce syngas in FTS are: natural gas in gas-to-liquid (GTL), coal in coal-to-liquid (CTL), and biomass in biomass-to-liquid (BTL) conversion technologies [10]. Each source has its own merits and demerits which sometimes dictate syngas production, processing and composition, adjustment technologies, and catalyst selection. For example when the feedstock is biomass, ultimate analysis of the biomass may dictate the use of either gasification or pyrolysis. Also the low hydrogen production potential of biomass versus natural gas will require the use of iron over cobalt since the former catalyzes the water-gas-shift (WGS) reaction which internally raises the H_2 partial pressure, thereby adjusting the H_2: CO ratio for good catalytic performance. Recently, the FTS is at the center of efforts to produce drop-in fuels from biomass especially JP-8/jet fuels for the aviation industry [11]. This is because BTL catalytic conversion via the FTS process will not require modifications to the existing automotive infrastructure; thereby representing a huge saving in the initial capital investments. Generally, kinetics for cobalt based catalysts present different expressions when compared to those for the iron-based catalysts, and kinetic equations are based on the rate-determining step which involves a dual-site surface reaction; resulting in a squared expression in the denominator in the rate equation. Co catalyst presents high selectivity toward C_{13}-C_{17} paraffin's and kinetic model from experimentation is essential to express the product distribution [12], FTS reactors including the multi-tubular fixed bed reactors (MTFBR), fluidized bed reactors (FBR), and slurry bubble column reactor (SBCR), these reactors are different approaches selected to achieve temperature control and to arrive at a better choice of the catalyst and reaction products. The main objectives of this research is to study the kinetics of converting synthesis gas (H_2/CO) using Co catalyst and to study the effect of different operating conditions on conversion.

2. Experimental Work

2.1. Preparation of catalyst

Co-Mo/Al_2O_3 catalyst was prepared by wet incipient precipitation (WIP). The precursor (Mo/Al_2O_3) was dried at

110°C with air for 2 hours then placed in a flask prior to impregnation step. This method (WIP) consisted in putting in adding 200gm of cobalt chloride to 150 ml of distilled water. The catalyst is evacuated for 1 hour at 100°C to remove air from catalyst's pores. The procedure of impregnation starts with impregnating 50g of the catalyst with 200ml of an aqueous solution of Co metal precursor under vacuum (5mm Hg) accompanied with shaking for 30 min at 100°C. The solution was mixed during 30 minutes and further dried at 125ºC. The dried catalyst was crushed into a fine powder (125-250μm particle size) in order to provide high specific surface area for reaction. The calcination step was performed at 300ºC.

2.2. Characterization of catalysts

X-ray diffraction patterns were recorded in the 30 to 120° angle range using CuKα radiation (λ = 0.5 Å) in a Rotaflex RU-200 rotating anode x-ray diffracto meter (Rigaku). Surface Area and Pore Volume Analyses were performed in a Micromeritics ASAP 2000 instrument by N_2 sorption analyses at -195 °C and vacuum pressure of 20μmHg. Temperature programmed Reduction test (TPR) was carried out in Netzsch Simultaneous TGA/DTA which is a thermal analysis instrument that measures both thermo gravimetric (TGA) and differential thermal analysis (DTA) simultaneously with a single sample in a controlled environment (i.e. argon, nitrogen or compressed air) from room temperature to 1500ºC. In the TPD experiments, 0.20 g of the Co/Mo catalyst sample, with pre-adsorbed NH_3, was heated in argon at 30cm^3/min. A few selected samples were pre-reduced at 400 °C in 3 volume% H_2/N_2 gas (30 cm^3/min), followed by re-adsorption of NH_3 (0.1 volume% NH_3/N_2 gas) at ambient temperature for 30 min. Scanning Electron Microscopy (SEM) analysis was performed in Hitachi S-570 Scanning Electron Microscope which is equipped with a lanthanum hexaboride (LAB6) electron gun operated at 2.5 KV and 16 mA for 6mins followed by sample analyses at 25 KV and a magnification of 5,000. All catalyst characterization tests were performed in Advanced Materials Characterization Laboratory/Missouri University of Science and Technology.

2.3 Gas analysis (thermal conductivity)

Hydrogen percentage in synthesis gas (SG) and outlet stream are evaluated according to thermal conductivity. The estimation of the outlet concentration of unreacted H_2 can be accomplished by reading the value in in feed gas and exit streams K6050, HITECH Instruments Limited, UK by measuring the changes of carbon dioxide constituent in synthesis gas mixtures which is behave as pseudo binary gas mixture (i.e, mixtures in which only one constituent changes, the hydrogen gas). Hydrogen percentage in feed gas and exit streams are determined by gas analyzer (K6050, HITECH Instruments Limited, UK) by measuring the changes of hydrogen constituent in synthesis gas mixtures which is behave as pseudo binary gas mixture(i.e, mixtures in which only one constituent changes, the hydrogen gas). The technique of measuring gas percentage using this analyzer is based on its thermal conductivity of H_2 which defiantly differs from that of the other components.

The non-depleting cathetometer sensor which is imbedded inside the gas trap measures the thermal conductivity of the gas. The sensor incorporates a highly sensitive element of low thermal capacity. Signal processing is provided by a microprocessor (accuracy ±1% of span, stability up to 1% fsd/month) to give a level of accuracy and a range not normally associated with this type of sensor. Calibration is required after 20 tests to ensure the system's inherent and high stability. The ranges provided depend upon the gas being measured but can be from high ppm to 100%. The operating conditions are: sample flow rate for optimum performance 100 to 300ml/min, sample temperature have to be in the range of –10 to +55°C (non-condensing), and sample pressure is 3 barg maximum.

2.4 Catalyst evaluation

Catalytic reaction was carried out in a fixed bed reactor unit in chemical engineering department/college of engineering/Tikrit University. Figure (1) shows a schematic diagram of fixed bed reactor unit. The reactor tube (I.D=11mm, L= 700 mm) was made of 316 L modified stainless steel and heated by a ceramic beaded electrical resistance coil placed in grooves on the external surface of the reactor to supply the required heat of reaction throughout the reactor. All catalyst characterization tests were performed in Advanced Materials Characterization Laboratory/Missouri University of Science and Technology. A sample of 10.0 gm, on dry basis, of the catalyst was weighed. The catalyst was loaded into the reactor and embedded with two layers of ceramic balls. Nitrogen was flown upward with a volumetric flow rate of 5 L/h for heat to supply heat and maintain the reactor at the temperature of reaction for each run. The temperature controller was turned on and set to the desired temperature according to a calibration curve. Prior to the reaction, the catalyst (0.8 g) mixed with 2.4 g quartz sand to eliminate a hot spot was placed in the reactor and pure hydrogen was introduced at a flow rate of 100 mL/min under atmospheric pressure. The reactor temperature was increased from room temperature to 400°C at a rate of 10°C/minute, maintained at this activation condition for 10 hours and the catalyst was reduced in situation. After the activation period, the reactor temperature was decreased to 180°C under flowing hydrogen. The gaseous mixture of syngas (90 vol. % with H_2/CO ratio of 2 and argon 10 vol. %, as internal gas for quantitative analysis) was fed to the reactor at a flow rate of

100 mL/min and the reactor pressure was increased to 2 MPa. The reactor temperature was then increased to 230°C at a rate of 0.5°C/min.

The temperature was increased at a rate of 10 ^{o}C/min until reaching 400 ^{o}C. The gas analyzer was turned on to record the hydrogen percentage in the outlet gas stream. A flow of 5L/h of 25% hydrogen and 75% of nitrogen was flown in the reactor at 1 bar pressure. Reduction of catalyst was lasted for 4 hrs. After reduction, hydrogen stream was turned off and the reactor was cooled down to 170^{o}C by nitrogen. The synthesis gas (H_2/CO=2.5) was fed to the reactor. To avoid reactor runaway, a slow temperature ramping was used during startup. The nitrogen was switched off after running of the synthesis gas. After 60 minutes, the syngas was turned off and nitrogen is passed through the system to remove the remaining synthesis gas and to get the system ready for the next run.

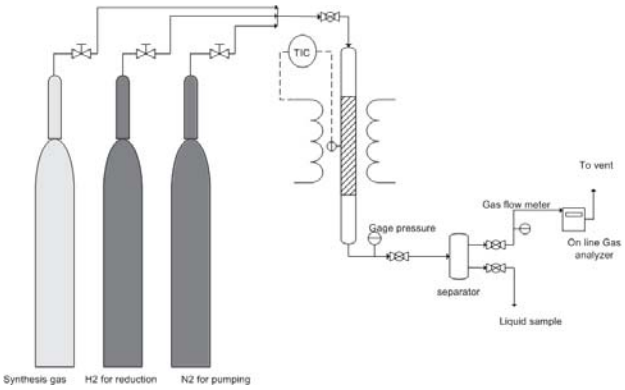


Figure 1. Schematic flow diagram of fixed bed FTS rig

3. Results and Discussion

3.1Catalyst characterization

3.1.1 Temperature programmed Reduction test (TPR)

The spectrum of temperature-programmed-reduction (TPR) for the prepared catalyst is shown in Figure (2). The reductions were performed up to 450°C and the pattern shows a complete reduction of the cobalt element in the catalyst. Water evaporation and residual oxides decomposition most likely occurred from 15–220°C. The peaks having maximum at 260°C and 600°C are assigned to the reduction of CoO to Co. Molybdenum interaction with alumina is greater than interaction of cobalt with alumina because the peaks of TPR plot for the prepared catalyst appeared at higher temperature. This temperature shift attributes to the strong interaction between molybdenum and alumina rather than the interaction of cobalt with alumina, so that it needs higher temperature to be reduced. This spectrum was used in designing a reduction profile to maximize reduction and minimize water evaporation rates in order to maintain low product water concentrations in the catalyst pores, and thereby limit hydrothermal sintering of the catalyst[11].

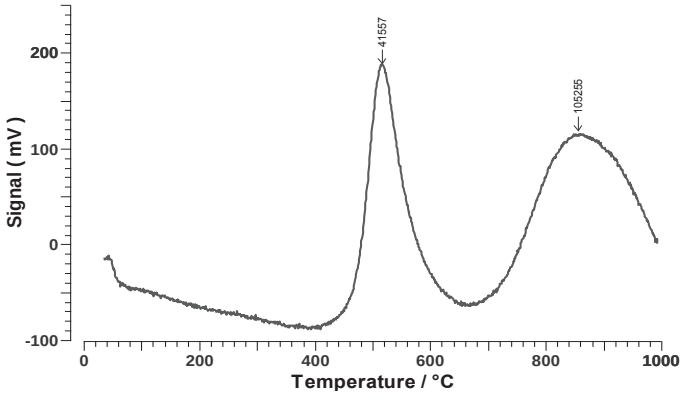


Figure 2. Temperature programmed reduction for the homemade CoMo/alumina

3.1.2. X-ray diffraction

By comparing the synthesis activity to the XRD results, it is possible to correlate the species on the catalyst surface to the activity in order to identify or detect different crystalline phases present in the catalyst. The XRD pattern of prepared catalyst (CoMo) is given in the Figure 3. It can be concluded from this figure that there is no negative effect of calcinations temperature on the XRD pattern, X-ray diffraction revealed no peaks other than that of alumina, indicating high dispersion of CoMo supported compounds on the catalyst surface. The calcinations temperature of catalyst does not change the crystalline structure of the, especially for bimetallic catalyst when a metal active phase is loaded over the support.

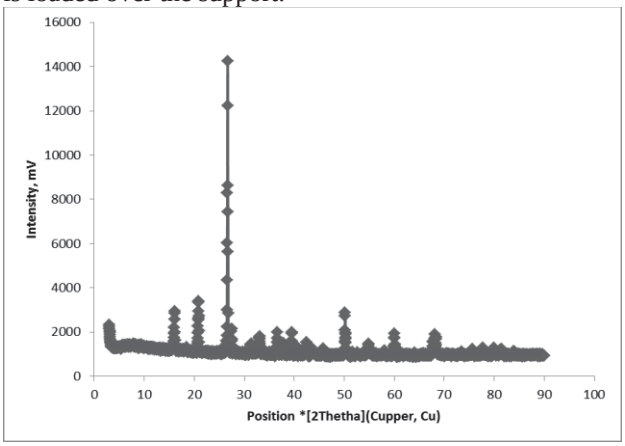


Figure 3.X-ray diffraction of the homemade CoMo/alumina

3.1.3 Scanning electron microscope.

Figures 4 shows surface mapping obtained by SEM of the homemade (CoMo/alumina) respectively. This figure shows a good distribution of active metals (CoMo) which represent small particle size while the alumina support represents large particle or surface which contains small

particles, the morphology of the catalyst and particle sizes was clearly shown. With such information, catalyst preparatory conditions can be adjusted to prevent particle agglomeration and to give better activity. Figure 4 shows a cluster of large particles and well distributed small particles. The large particles appeared triangle in shape with smooth edges. This might be due to the preferential interaction of Co with the support, causing metal particles to migrate to the surface [14, 15]. The impregnated particles did not have a significant effect on the shape of the large particles.

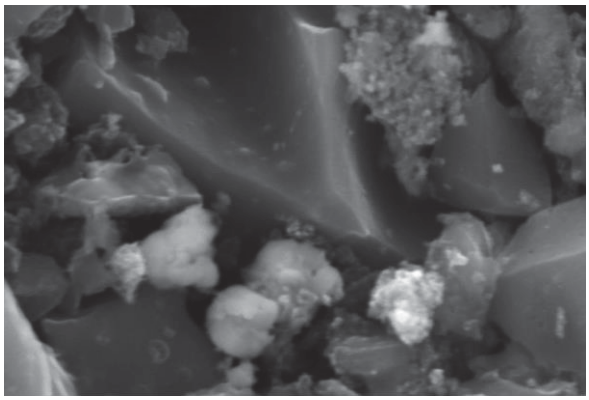


Figure 4. SEM of the homemade CoMo/alumina

3.1.4 Surface area

The surface area and pore volume of the prepared catalysts are listed in Table (1). The method of wetness impregnation yielded high surface area catalyst close to that of pure Al_2O_3.

Table 1. The surface area and pore volume of the prepared catalysts.

Property	value
surface area m^2/gm	468
Pore volume , cm^3/gm	36.81

3.2 Catalyst evaluation

3.2.1 Effect of operating pressure on H2 conversion

Figure 5 shows the effect of pressure on the FTS studied at different pressure levels (22 to 25 bar), under the reaction conditions of H_2/CO = 2.5/1 and 260-300 °C at 30 min time on stream, the increase of the total pressure enhances slightly the H_2 conversion for all temperatures, from 32.04% at 20 bar to 41.37% at 25 bar for 260 ºC and from 41.26% at 20 bar to 60.10% at 25 bar for 300 ºC. Similar results had also been reported with Sari et al [16] using Fe-Co catalyst at the same range of operating conditions and 3 H_2/Co ratio. It was found that the selectivity to heavier hydrocarbons (C_5^+) strongly increases with pressure, while the selectivity to both methane and olefins declines. The effect of pressure on process selectivity can be interpreted by considering the olefins reactivity during low-temperature FTS. Increasing pressure would generally result in the condensation of hydrocarbons[17-19]. Under typical reaction conditions, a thin layer of liquid hydrocarbon covers the catalyst particles. As a result, before the reactants reach the catalyst surface they have to diffuse inside this layer, while reaction products have to do the same in the opposite direction before being desorbed. It is well known that olefin, in contrast to paraffin, can be reabsorbed on the active sites, reinserted in the chain growth process, or can be hydrogenated to the corresponding paraffin [20-22].

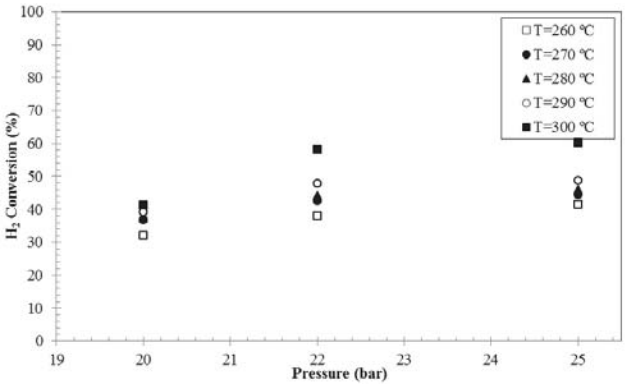


Figure 5. Influence of the total pressure on H_2 conversion at different temperatures and 0.4hr^{-1} and 30 min for fixed bed reactor.

3.2.2 Effect of operating temperature on hydrogen conversion.

Figures 6 was plotted at the end of time on steam which was 30 min. It is evident for the fixed bed reactor that the increase in temperature enhances H_2 conversion at all pressures strongly, from 32.04% at 260 ºC to 41.26% at 300 ºC for 20 bar, from 37.97% at 260 ºC to 58.21% at 300 ºC for 22 bar and from 41.37% at 260 ºC to 60.10% at 300 ºC at 25 bar. After a short transient period for both conditions, the conversion of H_2 quickly peaked at 60%. Steady state performance was maintained at 22 atm, while deactivation of the catalyst was observed after 5 hours at 20 atm. It had been reported in the literature that condensation of gas phase hydrocarbons on the catalyst surface leads to deactivation through carbon nan-rods formation [23]. Hydrogen conversion increases strongly with temperature for all operating pressures and it is expected that selectivity to methane also increases with temperature and selectivity toward light olefin until 260°C and then decreases with temperature [24]**.** Temperature has a modest effect on chain growth probability (C5+ selectivity) for the homemade catalyst.

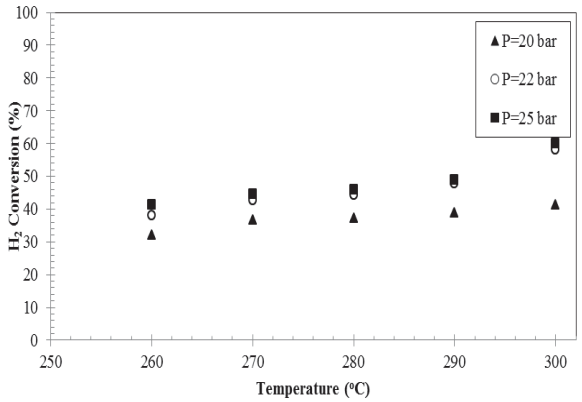


Figure 6. Influence of the reaction temperature on H_2 conversion at three different pressures and 0.4hr-1 and 30 min for fixed bed reactor.

4. Kinetic Modeling

By taking advantage of the lump principle, the kinetic data of the experiment was fitted to power law model. The model parameters were obtained by non-linear estimation analysis. The residual sum of square between was considered as the objective function and the parameter estimation to the kinetic model was developed in the Levenberg-Maquardt method. The model parameter in terms of power function was acquired, which was defined as:

$H_2 + 2S \leftrightarrow 2H.S$	$r_{H_2}=k_{H_2}.P_{H_2}.\theta_v^2 - k_{-H_2}.\theta_H^2$
$CO + 2S \leftrightarrow C.S + O.S$	$r_{CO}=k_{CO}.P_{CO}.\theta_v^2 - k_{-CO}.\theta_C.\theta_O$
$C.S + H.S \rightarrow CH.S + S$	$r_C=k_C.\theta_C.\theta_H$
$H.S + O.S \rightarrow OH.S + S$	$r_{OH}=k_{OH}.\theta_O.\theta_H$
$OH.S + H.S \rightarrow H_2O + 2S$	$r_w=k_w.\theta_{OH}.\theta_H$

From equation (1):

$$\theta_H=\left(K_H . P_{H_2}\right)^{0.5} . \theta_v \quad (6)$$

From equation (2):

$$\theta_C=\frac{K_{CO} . P_{CO} . \theta_v^2}{\theta_O} \quad (7)$$

$$r_C = r_{OH} = r_w$$

From equations (3) and (4):

$$\theta_O=\frac{k_C \theta_C}{k_{OH}} \quad (8)$$

Substitute equation (8) in equation (7):

$$\theta_C=\left(\frac{K_{CO} . k_{OH} P_{CO}}{k_C}\right)^{0.5} . \theta_v \quad (9)$$

Substitute equation (9) in (8)

$$\theta_O=\frac{k_C \theta_C}{k_{OH}}=\frac{k_C}{k_{OH}} . \left(\frac{K_{CO} . k_{OH} P_{CO}}{k_C}\right)^{0.5} . \theta_v \quad (10)$$

Substitute equations (6) and (9) in equation (3):

$$r_C=r_{FT}=k_C.\theta_C.\theta_H=k_C.\left(\frac{K_{CO} . k_{OH} . P_{CO}}{k_C}\right)^{0.5} . \theta_v . \left(K_H . P_{H_2}\right)^{0.5} . \theta_v$$

$$r_{FT}= k_C . \left(\frac{K_{CO} . k_{OH} . K_H}{k_C}\right)^{0.5} . P_{CO}{}^{0.5} . P_{H_2}{}^{0.5} . \theta_v^2 \quad (11)$$

$$r_{FT}= K_{FT} . P_{CO}{}^{0.5} . P_{H_2}{}^{0.5} . \theta_v^2$$

Among the adsorbed species during chemisorptions, CH, CH_2 and OH coverage are very low, therefore these species can be neglected. As a result one could write the total area coverage fractions as [25]:-

$$1= \theta_v + \theta_C + \theta_O + \quad (12)$$

Substitute equations (6) , (9) and (10) in (12)

$$1= \theta_v + \left(\frac{K_{CO} . k_{OH} . P_{CO}}{k_C}\right)^{0.5} . \theta_v + \frac{k_C}{k_{OH}} . \left(\frac{K_{CO} . k_{OH} . P_{CO}}{k_C}\right)^{0.5} . \theta_v + \left(K_H . P_{H_2}\right)^{0.5} . \theta_v$$

$$1=\theta_v . \left(1+k' . P_{CO}^{0.5} + \frac{k_C}{k_{OH}} . k' . P_{CO}^{0.5} + k_H^{0.5} P_{H_2}^{0.5}\right)$$

$$r_{FT}= K_{FT} . P_{CO}{}^{0.5} . P_{H_2}{}^{0.5} . \frac{1}{\left(1+k' . P_{CO}^{0.5} + \frac{k_C}{k_{OH}} . k' . P_{CO}^{0.5} + k_H^{0.5} P_{H_2}^{0.5}\right)^2}$$

$$r_{FT}= \frac{K_{FT} . P_{CO}{}^{0.5} . P_{H_2}{}^{0.5}}{\left(1+\left(k' . + \frac{k_C}{k_{OH}} . k'\right) . P_{CO}^{0.5} + k_H^{0.5} P_{H_2}^{0.5}\right)^2}$$

$$r_{FT}=k_o . EXP\left(\frac{-E}{R.T}\right) \frac{P_{CO}{}^{0.5} . P_{H_2}{}^{0.5}}{\left(1+k_1 P_{CO}^{0.5} + k_2 P_{H_2}^{0.5}\right)^2} \quad (13)$$

$$k' = \left(\frac{K_{CO} . k_{OH}}{k_C}\right)^{0.5} \quad (14)$$

$$k_1 = \left(k' . + \frac{k_C}{k_{OH}} . k'\right) \quad (15)$$

$$k_2 = k_H^{0.5} \quad (16)$$

$$r_{FT}=k_o . EXP\left(\frac{-E}{R.T}\right) \frac{P_{CO}{}^{0.5} . P_{H_2}{}^{0.5}}{\left(1+k_1 P_{CO}^{0.5} + k_2 P_{H_2}^{0.5}\right)^2} \quad (17)$$

Using another mechanism of dissociation:

$H_2 + 2S \leftrightarrow 2H.S$	$r_{H_2}=k_{H_2}.P_{H_2}.\theta_v^2 - k_{-H_2}.\theta_H^2$	(18)
$CO + S \leftrightarrow CO.s$	$r_{CO}=k_{CO}.P_{CO}.\theta_v - k_{-CO}.\theta_{CO}$	(19)
$CO.S + S \leftrightarrow C.S + O.S$	$r_O=k_O.\theta_{CO}.\theta_v - k_{-O}.\theta_C.\theta_O$	(20)
$C.S + H.S \rightarrow CH.S + S$	$r_C=k_C.\theta_C.\theta_H$	(21)
$H.S + O.S \leftrightarrow H.S + S$	$r_{OH}=k_{OH}.\theta_O.\theta_H - k_{-OH}.\theta_{OH}.\theta_v$	(22)

$OH.S + H.S \rightarrow H_2O + 2S$ $r_w = k_w . \theta_{OH} . \theta_H$ (23)

The rate expression is developed according to the same steps of equations 6 through 16:

$$r_{FT} = k_o . EXP\left(\frac{-E}{R.T}\right) \frac{P_{CO}^{0.5} . P_{H_2}^{0.75}}{\left(1 + k_3 . P_{CO}^{0.5} P_{H_2}^{0.25} + k_2 . P_{H_2}^{0.5} + K_{CO} . P_{CO}\right)^2} \quad (24)$$

Assume the reaction rate follow the power low as:-

$$r_{FT} = k' . P_{H_2}^{\alpha} . P_{CO}^{\beta} \quad (25)$$

Where k' is function to temperature as Arrhenius' law

$$k' = k_o EXP\left(\frac{-E}{R.T}\right) \quad (26)$$

$$r_{FT} = k_o . EXP\left(\frac{-E}{R.T}\right) . P_{H_2}^{\alpha} . P_{CO}^{\beta} \quad (27)$$

Tables 2 and 3 illustrated the kinetic parameters for rate expressions in equations 17 and 24. The estimated values of activation energy (E) for FTS were 14866.74 and 14865.58 kJ/mol respectively whereas a value of 17184.86 kJ/mol was obtained by applying power law model as shown in Table 4, this value agrees with those obtained previously at the same examined temperature range manifested in the literature[26]. It was demonstrated that H_2 was easily absorbed on the surface of the catalyst and CO absorption was saturated. This standpoint corresponds with the view reported in the literature [27]. By the other hand, the comparison between the values of the experiments and the model resulted in a percent of error not more than 10%.

5. Conclusions

The homemade cobalt-molybdenum catalyst exhibits good characteristics upon testing where the method of wetness impregnation yielded high surface area catalysts close to that of pure Al2O3 with a good performance in a fixed bed reactor. Hydrogen pressure and operating temperature affected the reaction rate significantly using the homemade cobalt catalyst. The kinetic parameters estimated by applying the experimental results of catalyst evaluation showed a fair percent of error and agreement with open literatures.

6. References

[1] KS Ha, JW Bae, KJ Woo, KW Jun. "Efficient utilization of greenhouse gas in a gas-to-liquids process combined with carbon dioxide reforming of methane"; Environ Sci Technol. 15;44(4):1412-7, 2010.

[2] Anderson RB. "The Fischer–Tropsch synthesis". New York: Academic Press; 1984.

[3] Franco A, Diaz AR. "The future challenges for ''Clean Coal Technologies'': joining efficiency increase and pollutant emission control"; Energy; 34:348–54, 2009.

[4] Kojima R, Aika KI. "Molybdenum Nitride and Carbide Catalysts for Ammonia Synthesis"; Appl Catal A: Gen, 219:141–7, 2009.

[5] Li S, Lee JS, Hyeon T, Suslick KS. "Catalytic Hydro-Denitrogenation of Indole over Molybdenum Nitride and Carbides With Different Structures"; Appl Catal A: Gen; 184:1–9, 1999.

[6] Ghani, S. A., Al Sarag M. A., Jada'a W. A., Ultra Deep Hydrotreatment of Iraqi Vacuum Gas Oil Using a Modified Catalyst, Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, 37:283–290, 2015

[7] Ghani, S. A., Jada'a W. A. Inhibitory Study for Joint Reactions of Hydrodesulfurization and Hydrodenitrogenation during Hydrotreating of Vacuum Gas Oil, Proceedings of the World Congress on Engineering and Computer Science 2012 Vol II WCECS 2012, October 24-26, 2012, San Francisco, USA

[8] Yao Z, Shi C. "Development of a catalytic cycle in molybdenum carbide catalyzed NO/CO reaction"; Catal Lett; 130:239–45, 2009.

[9] Tominaga H, Nagai M. "Theoretical study of methane reforming on molybdenum carbide"; Appl Catal A: Gen; 328:35–42, 2007.

[10] Suk-Hwan Kang, Kwang-Jae Woo, Jong Wook Bae, Ki-Won Jun, and Yong Kang; "Hydrogenation of CO on supported cobalt γ-Al_2O_3 catalyst in fixed bed and slurry bubble column reactors"; Korean J. Chem. Eng., 26(6), 1533-1538, 2009.

[11] de Klerk A., Edward Furimsky "Catalysis in the Refining of Fischer–Tropsch Syncrude, 2010

[12] Victor R. Ah, Paulo L. C. Lage, Carlos D. D. de Souza, Fabiana M. Mendes, and Martin Schmal, Kinetic Rates of the Fischer Tropsch Synthesis on a Co/NbaOs Catalyst" ; Journal of Natural Gas Chemistry 15, 307-312, 2006.

[13] Botchwey C., Syntheses, Characterization and Kinetics of Nickel-Tungsten Nitride Catalysts for Hydrotreating of Gas Oil, PhD thesis, University of Saskatchewan, Saskatoon, 2010

[14] Ferdous, D., A. K. Dalai and J. Adjaye, "X-ray absorption near edge structure and X-ray photo electron spectroscopy analyses of NiMo/Al_2O_3 catalysts containing boron and phosphorus"; Journal of Molecular Catalysis A: Chemical. 234, 169-179, 2005.

[15] Sundaramurthy, V., A.K. Dalai and J. Adjaye, "The effect of phosphorus on hydrotreating property of NiMo/g-Al_2O_3 nitride catalyst"; Applied Catalysis A: General. 335, 204-210, 2008.

[16] Sari A., Zamani Y., Taheri S.A., "Intrinsic kinetics of Fischer–Tropsch reactions over an industrial Co–Ru/–Al_2O_3 catalyst in slurry phase reactor"; Fuel Processing Technology 90, 1305–1313, 2009.

[17] Griboval A., Khodakov, A. Y., Bechara, R., Zholobenko, V. L. "Studies in Surface Science and Catalysis" 144, 609, 2002.

[18] Nawaf Amer T., Gheni Saba A., Jarullah Aysar T., Mujtaba Iqbal M., 2015. Improvement of fuel quality by oxidative desulfurization: Design of synthetic catalyst for the process, Fuel Processing Technology, 138, 337-343.

[19] Nawaf Amer T., Gheni Saba A., Jarullah Aysar T., Mujtaba Iqbal M., Optimal Design of a Trickle Bed Reactor for Light Fuel Oxidative Desulfurization Based on Experiments and Modeling, Energy Fuels, 29, 3366− 3376

[20] Chu W, Chernavskii P A, Gengembre L, Pankina G A, Fongarland P, Khodakov A Y. "Cobalt species in promoted cobalt alumina-supported Fischer–Tropsch catalysts"; J Catal, 252(2): 215–230, 2007.

[21] Li Y W, He D H, Yuan Y B, Chen Z X, Zhu Q M. "Synthesis of isobutene from CO hydrogenation over nano-sized zirconium dioxide-based catalysts"; Chinese Journal of Catalysis, 23(2): 185–190, 2002.

[22] Xiong H F, Zhang Y H, Liew K Y, Li J L. "Catalytic performance of zirconium-modified Co/Al_2O_3 for Fischer–Tropsch synthesis"; J Mol Catal A Chem, 231(1): 145–151, 2005.

[23] Noritatsu T, Yoshiharu Y, M. Keisuke, J. Kaoru. ;Catal. Communication. 4, 108–111, 2003.

[24] Sarkari M., F. Fazlollahi,b A. Razmjooie, and A. A. Mirzaeid. "Fisher-Tropsch Synthesis on Alumina Supported Iron-Nickel Catalysts: Effect of Preparation Methods"; Biochem. Eng. Q. 25 (3) 289–297, 2011.

[25] Haghtalab A., M. Nabipoor, S. Farzad. "Kinetic modeling of the Fischer–Tropsch synthesis in a slurry phase bubble column reactor using Langmuir–Freundlich isotherm"; ", Fuel Processing Technology xxx; xxx–xxx, 2011.

[26] Schulz, H.,Claeys, M. "Reaction of α-olefins of different chain length added during Fischer -Tropsch synthesis on a cobalt catalyst in a fixed bed reactor"; Applied Catalysis A: General, 186, 91-107, 1999.

[27] Chang, J., Teng, B.T., Bai, L.,et al. "Detailed kinetic study of FTS on $Co/ZrO_2/SiO_2$ catalyst"; Chinese Journal of Catalysis, 26(10):859-868, 2005.

Development of Educational Virtual Reality Modules for Underground Fuel Pipeline Failure

Litao Shen[1], John Moreland[1], Chandramouli Viswanathan[2], Chenn Zhou[1]
[1]Center for Innovation through Visualization and Simulation
[2]Mechanical Engineering Department
Purdue University Calumet, Hammond, IN, USA

Abstract— *This research describes the motivations and technical approach for developing educational 3D Virtual Reality modules to teach about the movement of contaminants due to underground fuel pipeline failure. The research uses MT3D modeling software and the Unity 3D game engine to produce interactive content that allows users to explore contaminant results and learn about potential remedial strategies.*

Keywords: Fuel Pipeline, Contamination, Virtual Reality, Virtual Education, Hydrology, Simulation

1 Introduction

Environmental Protection Agency (EPA) established standards for different water quality constituents for surface and ground water. Understanding the behavior of the contaminating sources, the contamination movement with ground and surface waters are extremely crucial in keeping the environment clean and maintaining the ecosystem. Visualizing the solute transport through porous media is very essential in planning the remedial strategies and viable economical options. This outreach and research project proposes developing simple user friendly immersive virtual 3D visualization modules for understanding the complex process of contaminant movement with the ground water when an underground fuel pipeline failure occurs and subsequently the contamination joins a surface water stream. This will be extremely helpful to educate the public during new pipe line development. The work is based on methodology used for previous developments with groundwater simulation and interactive 3D visualization of groundwater contamination [1-4].

There are a few crude oil pipe lines running through Northern Indiana from oil and gas companies. These pipes are installed underground which makes it difficult to evaluate the safety problem of these pipes. For example, when there is a leak in the pipe, how the crude oil will disburse is difficult to understand, and to do experiment will be very difficult and dangerous. To solve this problem, some visualization method of the crude oil distribution underground will be necessary and the models should be developed by incorporating the field boundary conditions. Several scenarios should be created to study the possible variations. To address this problem, this project was developed using MODFLOW [5] and MT3D [6] software initially. The results were used develop 3D modules for education.

This application provides different models that can show different scenarios of the contaminant distribution under the effect of underground water. These models are developed using scientific computing software and then processed into 3D models. This application also tries to build an environment that is similar to the leak with ground model, pipes and initial leak. Inside this application, there are many widgets to allow the use interact with the models easily like view perspective control and plane visibility control.

2 Methodology

The model development process involve multiple software. In stage one, the groundwater movement and solute transport were built using MODFLOW and MT3D. An interfacing software called GroundWater Vistas [7] was used to form grids, to provide data input and capture the results.

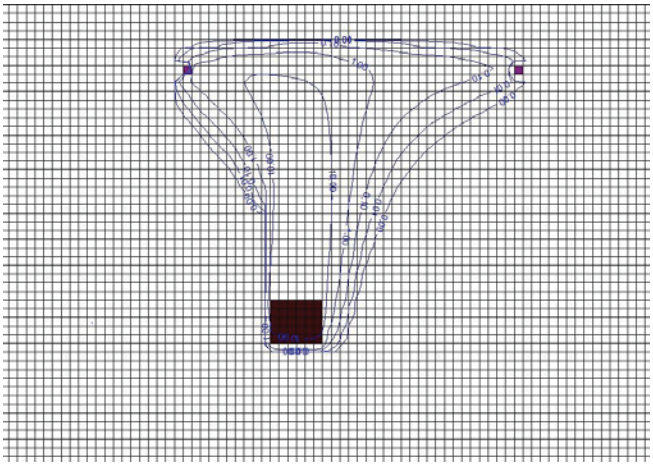


Figure 1. Mathematical Model in GroundWater Vistas

By using the real data, models with different configurations were built. This software provides results in 2D. The rendering effect of Groundwater Vistas is shown in Figures 1 & 2.

After the stage 1 model development simulations were conducted for 10 years and the results were captured from the quasi 3D groundwater model. Model results were imported into Paraview [8]. Using the the concentration contours of each layer, the data were covered to 3D datasets. Inside Paraview, different concentration gradients are picked to render the 3D model. These datasets were exported to *.vrml 3D data format. However, in order to use the file in 3Ds Max [8] for further processing, the format name was changed to *.wrl. There are more than 10 contour models for every time frame and every scenario. Total number of 3D models could be more than 400.

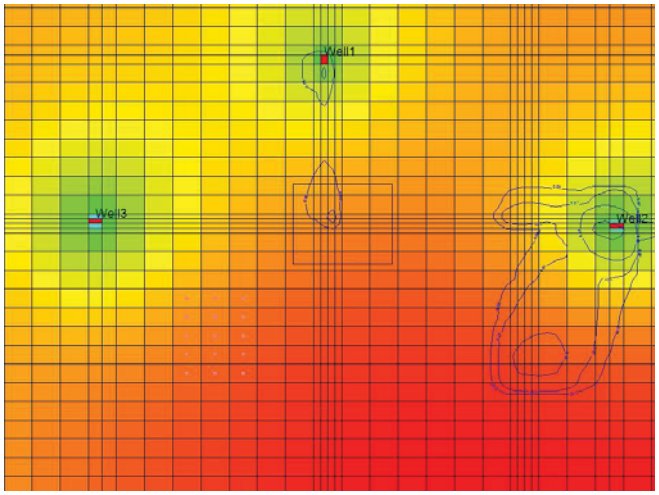


Figure 2. Contour map of the numerical results

A script was written to do document them automatically. In this application, we captured the results from Paraview and used a Python code to batch process the data for changing the file formats. Then these models were imported into 3Ds Max to turned to the format that Unity 3D [10] can handle. In 3Ds Max, the models were imported one by one and then exported to *.FBX file. Using the game engine Unity 3D the results were imported as the FBX file, and C# was selected as the script language to write the control logic code. In Unity 3D, there are two ways to represent the model, one has the color with the models which makes the model looks better but lacks the intuitive information of the concentration (Figure 2). Another way is to use the transparency to represent the model. All of the models have the same transparency, however, the layer by layer structure of the models makes it looks like the inside layer will have higher concentration thus illustrating the difference of concentration of oil. These models will change their shader and the transparency channel to achieve the best results. In this application, in order to represent the distribution of wells multiple cylinders have been made and in each scenario. Different cylinders have been grouped into the same object so that they can controlled at the same time. In Figure 2, there are two wells shown and the size is magnified many times to allow the user be able to see them, otherwise it would be too difficult to see a well at such scale.

Figure 3. 3D model that can show different scenarios

Multiple GUI widgets were made to aid the user to interact with the models. This is done by using the different distribution of the pumping wells. User would be able to see different models at different time with different well distributions. And with the help of slider bar, user would be able to see the models at different angles. Another thing user would be able to have control is the transparency of the plane or ground.

3 Results

This application is still under development with all the essential functions already finished for several different scenarios (Figure 3). It can visualize the mathematical model in 3D, and there are models for various situations for which the pumping wells are distributed in different places with different geometry, the time span ranges from 1 month up to 6 months. Some of the models will get to steady state in the fourth or fifth month. The main GUI layout can be shown in Figure 4. The images below the scenario label show the basic distribution of the wells and the calculation

result of the scenario. Some are chosen at lower or higher layers thus contains no black initial region. It should be noticed that the crude oil pipe's size has been scaled in order to allow the user to able to see it.

Figure 4. Main GUI for the controls with 6 scenarios displayed

This application also provides widgets that can aid the interaction between user and model. User will be able to view the model from different perspective using either the significant angle selection or the slider to have finer manipulation ability with capability of 90 degrees' difference. Since everything is underground, this application also provide the ability to change the earth place to allow the models be seen above or below or without ground. The control widgets can be shown in Fig 5.

Figure 5. Left is the control for ground level and right is the control for view angle

To illustrate the concentration difference, transparency has been utilized, however it cannot represent the results accurately, and in order to solve this problem there should be some object that be placed into the scene to pick the data at where it is. Figure 6 shows the effect of the transparent models and the sphere which will be used as data picker. The data picker function is still under development. This function is to give the user ability to see the calculated data directly instead of just using transparency to illustrate the concentration.

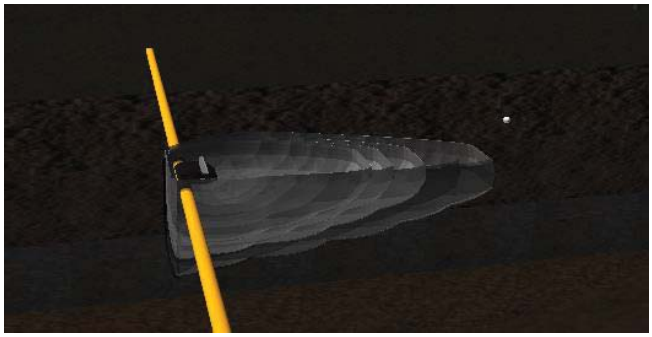


Figure 6 Transparent contour layers

4 Conclusion

This application is built from field data documentation. The 3D models are developed using MODFLOW and MT3D software using the field data. User would be able to see how the oil plume movement will respect to time and space. The time span of the calculation is also long enough to get steady state achievement. The number of contours also guarantees the geometry shape of the distribution can be depicted accurately. Thus the result is significant in helping people understand what will happen when there is leak in the oil pipes. Also there are multiple widgets that allow the user to interact with the application easily and study the effects of contamination with different scenarios and understand the process easily.

5 Acknowledgement

This research was supported by the Indiana Department of Natural Resources.

6 References

[1] Viswanathan, C., Moreland, J., Guo, S., and Zhou, C. (2011). "Usefulness of Virtual 3D Modeling to Visualize the Effect of Uncertain Data In Groundwater Solute Transport". Proceedings of the ASME 2011 World Conference on Innovative Virtual Reality, WI

[2] Guo, S., Moreland, J., Viswanathan, C., Zhou, C. (2012). "Virtual Reality System for Solute Transport in Groundwater", Proceedings of 2012 Annual ASEE IL/IN Section Conference, Valparaiso, IN, USA.

[3] Viswanathan, C., Narayana, M., Duruvai, V., Guo, S., Moreland, J., Merwade, V., Zhou, C. Q., Song, A., Hu, Y., and Zhang, F. (2011). "Generalized Visualization Modules for Solute Transport in Groundwater", 9th Symposium on Groundwater Hydrology, Quality, and Management EWRI ASCE Conference, Palm Springs, CA, pp. 934-942.

[4] Chandramouli, V., Ormsbee, L., Kopp, J., (2007). "Land Acquisition Study at Paducah Gaseous Diffusion Plant Site using MODFLOWT Modeling". American Society of Civil Engineers ASCE World Environmental and Water Resources Congress, May 15-19, Tampa, Florida, USA.

[5] Duffield, G.M. (1996). MODFLOWT: A modular Three-Dimensional Groundwater Flow and Transport Model, HydroSOLVE, Inc. and GeoTrans, Inc., Sterling, Virginia.

[6] McDonald, M.G., and A.W. Harbaugh (1988). "A Modular Three Dimensional Finite-Difference Groundwater Flow Model", Techniques of Water Resources Investigations, Book 6, Chapter A1, United States Geological Survey, Reston, VA.

[7] Rumbaugh, J. O., & Rumbaugh, D. B. (2004). Guide to using groundwater vistas. *Environmental Simulations, Reinholds, PA, USA*.

[8] Henderson, A., Ahrens, J., & Law, C. (2004). *The ParaView Guide*. Clifton Park, NY: Kitware.

[9] Murdock, K. L. (2011). *3ds Max 2012 Bible* (Vol. 783). John Wiley & Sons.

[10] Higgins, T. (2010). Unity-3d game engine.