SIMUL 2011

The Third International Conference on Advances in System Simulation

ISBN: 978-1-61208-169-4

October 23-29, 2011

Barcelona, Spain

SIMUL 2011 Editors

Aida Omerovic, SINTEF & University of Oslo, Norway

Diglio A. Simoni, RTI International - Research Triangle Park, USA

Georgiy Bobashev, RTI International - Research Triangle Park, USA

SIMUL 2011

Forward

The Third International Conference on Advances in System Simulation (SIMUL 2011), held on October 23-29, 2011 in Barcelona, Spain, continued a series of events focusing on advances in simulation techniques and systems providing new simulation capabilities.

While different simulation events are already scheduled for years, SIMUL 2011 identified specific needs for ontology of models, mechanisms, and methodologies in order to make easy an appropriate tool selection. With the advent of Web Services and WEB 3.0 social simulation and human-in simulations bring new challenging situations along with more classical process simulations and distributed and parallel simulations. An update on the simulation tool considering these new simulation flavors was aimed at, too.

The conference provided a forum where researchers were able to present recent research results and new research problems and directions related to them. The conference sought contributions to stress-out large challenges in scale system simulation and advanced mechanisms and methodologies to deal with them. The accepted papers covered topics on social simulation, transport simulation, simulation tools and platforms, simulation methodologies and models, and distributed simulation.

We welcomed technical papers presenting research and practical results, position papers addressing the pros and cons of specific proposals, such as those being discussed in the standard forums or in industry consortiums, survey papers addressing the key problems and solutions on any of the above topics, short papers on work in progress, and panel proposals.

We take here the opportunity to warmly thank all the members of the SIMUL 2011 technical program committee as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors that dedicated much of their time and efforts to contribute to the SIMUL 2011. We truly believe that thanks to all these efforts, the final conference program consists of top quality contributions.

This event could also not have been a reality without the support of many individuals, organizations and sponsors. We also gratefully thank the members of the SIMUL 2011 organizing committee for their help in handling the logistics and for their work that is making this professional meeting a success. We gratefully appreciate to the technical program committee co-chairs that contributed to identify the appropriate groups to submit contributions.

We hope the SIMUL 2011 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in simulation research.

We hope Barcelona provided a pleasant environment during the conference and everyone saved some time for exploring this beautiful city.

SIMUL 2011 Chairs

Advisory Chairs
Edward Williams, PMC-Dearborn, USA
Paul Fishwick, University of Florida-Gainesville, USA
Christoph Reinhart, Harvard University - Cambridge, USA

Research Liaison Chairs
Tae-Eog Lee, KAIST, Korea
Marko Jaakola, VTT Technical Research Centre of Finland, Finland

Industry Liaison Chairs
Diglio A. Simoni, RTI International – RTP, USA
Shengnan Wu, American Airlines, USA
Ann Dunkin, Palo Alto Unified School District, USA
Tejas R. Gandhi, Virtua Health-Marlton, USA

Special Area Chairs

Model-based system prediction
Georgiy Bobashev, RTI International -Research Triangle Park, USA
Aida Omerovic, SINTEF & University of Oslo, Norway

Process simulation
Ian Flood, University of Florida, USA
Gregor Papa, Jozef Stefan Institute - Ljubljana, Slovenia

Publicity Chairs
Nuno Melao, Catholic University of Portugal - Viseu, Portugal

SIMUL 2011

Committee

SIMUL Advisory Chairs

Edward Williams, PMC-Dearborn, USA
Paul Fishwick, University of Florida-Gainesville, USA
Christoph Reinhart, Harvard University - Cambridge, USA

SIMUL 2011 Research Liaison Chairs

Tae-Eog Lee, KAIST, Korea
Marko Jaakola, VTT Technical Research Centre of Finland, Finland

SIMUL 2011 Industry Liaison Chairs

Diglio A. Simoni, RTI International – RTP, USA
Shengnan Wu, American Airlines, USA
Ann Dunkin, Palo Alto Unified School District, USA
Tejas R. Gandhi, Virtua Health-Marlton, USA

SIMUL 2011 Special Area Chairs

Model-based system prediction
Georgiy Bobashev, RTI International -Research Triangle Park, USA
Aida Omerovic, SINTEF & University of Oslo, Norway

Process simulation
Ian Flood, University of Florida, USA
Gregor Papa, Jozef Stefan Institute - Ljubljana, Slovenia

SIMUL 2011 Publicity Chairs

Nuno Melao, Catholic University of Portugal - Viseu, Portugal

SIMUL 2011 Technical Program Committee

Ir. Ricky Andriansyah, Eindhoven University of Technology, The Netherlands
Godfried Augenbroe, Georgia Institute of Technology - Atlanta, USA
Reza Azimi, University of Alberta - Edmonton, Alberta Canada
Christian Bartsch, Research Center for Information Technology (FZI), Germany
Kiranmai Bellam, Prairie View A&M University, USA
Ateet Bhalla, NRI Institute of Information Science and Technology - Bhopal, India
Georgiy Bobashev, RTI International -Research Triangle Park, USA
Qingyan (Yan) Chen, Purdue University - West Lafayette, USA
Soolyeon Cho, The Catholic University of America, USA

Arnaud Cuccuru, CEA LIST/LISE, France
Duilio Curcio, University of Calabria - Rende (CS), Italy
Luis Antonio de Santa-Eulalia, Université du Québec à Montréal, Canada
Robert de Souza, The Logistics Institute - Asia Pacific, Singapore
Tom Dhaene, Ghent University - IBBT, Belgium
Bing Dong, Carnegie Mellon University - Pittsburgh, USA
Ann Dunkin, Palo Alto Unified School District, USA
Khaled S. El-Kilany, Arab Academy for Science - Alexandria, Egypt
Paul Fishwick, University of Florida-Gainesville, USA
Ian Flood, University of Florida, USA
Franco Fummi, Università di Verona, Italy
Tejas R. Gandhi, Virtua Health-Marlton, USA
Genady Grabarnik, St. John's University, USA
Christoph Grimm. TU Wien, Austria
Xiaolin Hu, Georgia State University, USA
Michael Hübner, Karlsruhe Institute of Technology, Germany
Marko Jaakola, VTT Technical Research Centre of Finland, Finland
Sung Min Kim, College of Bio System, Korea
SangHyun Lee, University of Michigan, USA
Fedor Lehocki, Slovak University of Technology in Bratislava, Slovak Republic
Paola Lecca, The Microsoft Research - University of Trento, Italy
Jennie Lioris, CERMICS, France
Francesco Longo, University of Calabria, Italy
Prabhat K. Mahanti, University of New Brunswick - Saint John, Canada
Don McNickle, University of Canterbury - Christchurch, New Zealand
Nuno Melao, Catholic University of Portugal - Viseu, Portugal
Yuri Merkuryev, Riga Technical University - Latvia
Marco Mevius, HTWG Konstanz, Germany
Lars Mönch, University of Hagen, Germany
Muaz Niazi, COMSATS Institute of IT - Islamabad, Pakistan
Michael North, Argonne National Laboratory, USA
Aida Omerovic, SINTEF & University of Oslo, Norway
Gürkan Özhan, Middle East Technical University - Ankara, Turkey / NATO C3 Agency - Den Haag, The Netherlands
Maurizio Palesi, Kore University, Italy
Gregor Papa, Jozef Stefan Institute - Ljubljana, Slovenia
Christoph Reinhart, Harvard University - Cambridge, USA
Revetria Roberto, University of Genoa, Italy
Claude-Alain Roulet, EPFL, Switzerland
Guodong Shao, National Institute of Standards and Technology - Gaithersburg, USA
Larisa Shwartz, IBM T. J. Watson Research Center - Hawthorne, USA
Cristina Silvano, Politecnico di Milano, Italy
Diglio A. Simoni, RTI International - RTP, USA
Vassili Toropov, University of Leeds, UK
Marija Trcka, Eindhoven University of Technology (TU/e), The Netherlands
Andrij Usov, Fraunhofer Institut IAIS, Germany
Shengyong Wang, The University of Akron, USA

Edward Williams, PMC-Dearborn, USA
Shengnan Wu, American Airlines Inc. - Fort Worth, USA
Levent Yilmaz, Auburn University, USA
Yao Yiping, National University of Defence Technology - Hunan, China

Copyright Information

Table of Contents

Stochastic Simulation of Snow Cover 1
Marketa Prusova and Lucie Jurikovska

The Lambda Chart: A Model of Design Abstraction and Exploration at System-Level 7
Falko Guderian and Gerhard Fettweis

Increase of Robustness on Pre-optimized Production Plans Through Simulation-based Analysis and Evaluation 13
Christoph Laroque, Robin Delius, Jan-Hendrik Fischer, and Dennis Horstkaemper

Fault-tolerant Distributed Discrete Event Simulator Based on a P2P Architecture 21
Jorge Luis Ramirez Ortiz and Ricardo Marcelin Jimenez

Distributed Simulation of Dense Crowds 27
Sergei Gorlatch, Christoph Hemker, and Dominique Meilaender

Distributed Simulation on a Many-Core Processor 32
Karthik Manian and Philip Wilsey

Simulation of Particle Deposition in an Airplane Cabin Mockup 38
Miao Wang, Chao-Hsin Lin, and Qingyan Chen

Traffic and Monotone Random Walk of Particles: Analytical and Simulation Results 44
Alexander Buslaev, Alexander Tatashev, and Andrew Yaroshenko

Tunnel Simulator for Traffic Video Detection 49
Sofie Van Hoecke, Steven Verstockt, Koen Samyn, Mike Slembrouck, and Rik Van de Walle

SUMO - Simulation of Urban MObility - an Overview 55
Michael Behrisch, Laura Bieker, Jakob Erdmann, and Daniel Krajzewicz

Mechanisms Controlling the Sensitivity of Amperometric Biosensors in the Case of Substrate and Product Inhibition 61
Dainius Simelevicius and Romas Baronas

A Generic Operational Simulation for Early Design Civil Unmanned Aerial Vehicles 67
Benjamin Schumann, Jim P. Scanlan, and Kenji Takeda

Traceability Handling in Model-based Prediction of System Quality 71
Aida Omerovic and Ketil Stolen

A Simulation-Based Innovation Forecasting Approach Combining the Bass Diffusion Model, the Discrete Choice Model and System Dynamics: An Application in the German Market for Electric Cars 81
Luis Antonio Santa-Eulalia, Donald Neumann, and Jo?rg Klasen

Plaque Lesion Classification Fuzzy Model Based on Various Color Models 88
Yuslinda Wati Mohamad Yusof, Hadzli Hashim, Khairul Anam Sulaiman, Suhaila Subahir, Noor Ezan Abdullah, and Fairul Nazmie Osman

A Multi-Layer Constraint and Decision Support System for Construction Operation 94
Amir Elmahdi, Hong Ha Le, and Hans-Joachim Bargstaedt

Agent-based simulation validation: A case study in demographic simulation 101
Cristina Montanola-Sales, Bhakti S. S. Onggo, and Josep Casanovas-Garcia

Analysis and Simulation of Power Law Distribution of File Types in File Sharing Systems 108
Yuya Dan and Takehiro Moriya

Simulation hypotheses 114
Pau Fonseca i Casas

SimARC: An Ontology-driven Behavioural Model of Alcohol Abuse 120
Francois Lamy, Pascal Perez, Alison Ritter, and Michael Livingston

Agent-Based Model (ABM) Validation Considerations 126
Philip Cooley and Eric Solano

Detailed Input data Source for Construction Process Simulation 132
Jurgen Melzner, Sebastian Hollermann, and Hans-Joachim Bargstadt

Energy Simulation Supporting the Building Design Process 136
Marco Massetti and Stefano Paolo Corgnati

Simulating Counterinsurgency and Coalition Strategies 142
David Arney and Kristin Arney

Simulation of Bacterial Self-Organization in Circular Container Along Contact Line as Detected by Bioluminescence Imaging 148
Romas Baronas

Applying Simulation and Mathematical Programming on a Business Case Analysis for Setting up a Spare Part Logistics in the Construction Supply Industry 154
Wilhelm Dangelmaier, Christoph Laroque, Robin Delius, and Jenny Streichhan

Integrating Current State and Future State Value Stream Mapping with Discrete Event Simulation: A Lean Distribution Case Study 161
Amr Mahfouz, John Crowe, and Amr Arisha

Review of Spatial Simulation Tools for Geographic Information Systems 169
Luis de Sousa and Alberto da Silva

Towards Internet Scale Simulation 175
Anthony McGregor

Simulation and Sustainability 182
Andi H. Widok and Volker Wohlgemuth

Towards a SDL-DEVS Simulator 188
Pau Fonseca i Casas and Josep Casanovas

Stochastic Simulation of Snow Cover

Markéta Průšová
Institute of Geoinformatics
VSB - Technical University of Ostrava
Ostrava – Poruba, Czech Republic
Email: marketa.prusova@vsb.cz

Lucie Juřikovská
Institute of Geoinformatics
VSB - Technical University of Ostrava
Ostrava – Poruba, Czech Republic
Email: lucie.jurikovska@vsb.cz

***Abstract*—The presented paper deals with a stochastic simulation of snow cover. This study aims to find the best settings of a stochastic simulation to be able to determine the parameters of the snow cover for any point of a given territory. Next, basic statistical analyses of parameters are documented, including an analysis of relationships between the snow parameters and altitude, slope and aspect. Most current methods of spatial interpolation and multifactor evaluation are based on the weighted regression relationships. That leads to smooth results and degrades our ability to properly evaluate the existence and the probability of extreme situations and their impact on the research problem. Neither alternative techniques use neural networks to bring major improvements. This research is exploring the possibility of stochastic simulation to assess the development of values, evaluating the occurrence of extreme events, monitoring the probability of exceeding the set limits, compared with application kriging errors, the use of additional qualitative information. The variants of conditional stochastic simulation were tested in particular. The application area was chosen on data of snow cover, many land-bound factors and the results are the regular mapping of forest damage. The aim is to compare and determine the best method of interpolation of snow cover, which was succeeded.**

Keywords—interpolation; stochastic simulation

I. INTRODUCTION

Geostatistical simulation is a well-known approach for modeling spatial uncertainty of a regionalized variable [3][7][8], by generating a high number of plausible realizations of a random function, conditional to the experimental data. Most interpolation methods (including kriging) give smooth images of the spatial variable while a simulation tries to mimic the true variability described by second order functions like the covariance or the variogram [11].

Snow is a dynamic natural element, the distribution of which is largely controlled by latitude and altitude. The regional extent of snow cover is an important variable in hydrology. In the hydrological cycle, snow represents seasonal water storage from where water is rapidly released during the melting period [12]. Prediction of snow cover is important for agriculture and flood prevention.

One of the main factors governing the distribution of snow properties is topography [2][10]. Although snow property data such as snow-water equivalent are often available in considerable temporal detail from a single point, the spatial resolution of snow property data is poor [13]. Often, only a few point measurements are available in the catchment of interest. Because of the extreme spatial variability of snow properties, small samples of these point data may not be representative of spatial patterns and spatial averages [5][6].

II. METHODOLOGY

Simulation is broadly defined as the process of replicating reality using a model. In geostatistics, simulation is the realization of a random function that has the same statistical features as the sample data used to generate it (measured by the mean, variance, and semivariogram). Gaussian geostatistical simulation (GGS), more specifically, is suitable for continuous data and assumes that the data, or a transformation of the data, has a normal (Gaussian) distribution. The main assumption behind GGS is that the data is stationary—the mean, variance, and spatial structure (semivariogram) do not change over the spatial domain of the data. Another key assumption of GGS is that the random function being modeled is a multivariate Gaussian random function [1].

Stochastic simulation differs from kriging in two ways, as follows:

- Kriging provides the "best", that is, minimum variance, local estimates without regard to the resulting statistics of those estimates. In simulation, however, the aim is to reproduce the global statistics and maintain the texture of the variation, and these take precedence over local accuracy.
- A kriged estimate at any place has associated with it a variance, and hence an uncertainty, that is independent of estimates at all other places. Confidence about it is usually based on an assumed Gaussian distribution with the mean equal to the estimate and a cumulative distribution function [14].

Increased use of GGS follows a trend in geostatistical practice that emphasizes the characterization of uncertainty for decision and risk analysis, rather than producing the best unbiased prediction for each unsampled location (as is done with kriging), which is more suited to showing global trends in the data [4][5]. Simulation also overcomes the problem of conditional bias in kriged estimates (high-value areas are typically underpredicted, while low-value areas are usually overpredicted).

Geostatistical simulation (GS) generates multiple, equally probable representations of the spatial distribution of the attribute under study. These representations provide a way to measure uncertainty for the unsampled locations taken all together in space, rather than one by one (as measured by the kriging variance). Moreover, the kriging variance is usually independent of the data values and generally cannot be used as a measure of estimation accuracy. On the other hand, estimation accuracy can be measured by building distributions of estimated values for unsampled locations using multiple simulated realizations that are built from a Simple Kriging model using input data that is normally distributed (that is, data that either is normally distributed or has been transformed using a normal score or other type of transformation). These distributions of uncertainty are key to risk assessment and decision analysis that uses the estimated data values [1].

GS assumes that the data is normally distributed, which rarely occurs in practice. A normal score transformation is performed on the data so that it will follow a standard normal distribution. Simulations are then run on this normally distributed data, and the results are back-transformed to obtain simulated output in the original units. When Simple kriging is performed on normally distributed data, it provides a kriging estimate and variance that fully define the conditional distribution at each location in the study area. This allows one to draw simulated realizations of the random function (the unknown, sampled surface) knowing only these two parameters at every location, and is the reason that GGS is based on a Simple kriging model and normally distributed data.

Results from simulation studies should not depend on the number of realizations that were generated. One way to determine how many realizations to generate is to compare the statistics for different numbers of realizations in a small portion of the data domain (a subset is used to save time). The statistics tend toward a fixed value as the number of realizations increases.

In conditional simulation, however, the generator must return the data values at places where we know them in addition to creating plausible values of $Z(x)$ elsewhere. We condition the simulation on the sampled data, $z(x_i)$, $i=1,2,...,N$. Denote the conditionally simulated values by $z_C^*(x_j)$, $j=1,2,...,T$. Where we have data we want the simulated values to be the same:

$$z_C^*(x_i) = z(x_i) \text{ for all } i=1,2,...,N. \qquad (1)$$

Elsewhere, $z_C^*(x)$ may depart from true but unknown values in accord with the model of spatial dependence adopted [14].

Consider what happens when we krige Z at x_0 where we have no measurement. The true value, $z(x_0)$, is estimated by $\hat{Z}(x_0)$ with an error $z(x_0)- \hat{Z}(x_0)$, which is unknown:

$$z(x_0) = \hat{Z}(x_0) + \{ z(x_0)- \hat{Z}(x_0) \} \qquad (2)$$

A characteristic of kriging is that the error is independent of the estimate, that is

$$E[\hat{Z}(y)\{z(x)- \hat{Z}(x)\}] = 0 \text{ for all } x,y \qquad (3)$$

This feature is used to condition the simulation.

We create a simulated field from the same covariance function or variogram as that of the conditioning data to give values $z_S^*(x_j)$, $j=1,2,...,T$, that include the sampling points, x_i, $i=1,2,...,N$. When we krige at x_0 from the simulated values at the sampling points to give an estimate $\hat{Z}_S^*(x_0)$. Its error, $z_S^*(x_0)- \hat{Z}_S^*(x_0)$, comes from the same distribution as the kriging error in equation (2), yet the two are independent. We can use it to replace the kriging error to give our conditionally simulated value as

$$z_C^*(x_0) = \hat{Z}(x_0) + \{ z_S^*(x_0)- \hat{Z}_S^*(x_0)\} \qquad (4)$$

The result has the properties we desire, as below [14].

1. The simulated values are realizations of a random process with the same expectation as original:

$$E[\hat{Z}_S^*(x)] = E[Z(x)] = \mu \text{ for all } x \qquad (5)$$

   where μ is the mean.
2. The simulated value should have the same variogram as the original.
3. At the data points the kriging errors $z(x_0)- \hat{Z}(x_0)$ and $z_S^*(x_0)- \hat{Z}_S^*(x_0)$ are 0, and $z_C^*(x_0) = z(x_0)$

Conditional simulation is more appropriate than kriging where our interest is in the local variability of the property and too much information would be lost by the smoothing effect of kriging. A suite of conditional simulations also provides a measure of uncertainty about the spatial distribution of the property of interest [14].

III. Pilot area

The pilot area is defined by the Šance catchment. It is located in the Frýdek-Mistek district. Šance reservoir was constructed on the upper course of the river Ostravice. All water drains to the river Odra.

The catchment network is defined by a regular grid of 2 x 2 km oriented along the axes of the coordinate system S-JTSK, which in total contains 52 squares. The grid is used for the systematic schema of sampling. Representative places are established inside each square cell. At least one place represents an open land and another place represents a forested land. A set of measurements (52) is performed in each representative place (around the point). The final sample data represents an average of all measured values (excluding outliers) in the location. A two kilometer step was selected. Treeless (open) and forested areas are carried out in one measurement in squares.

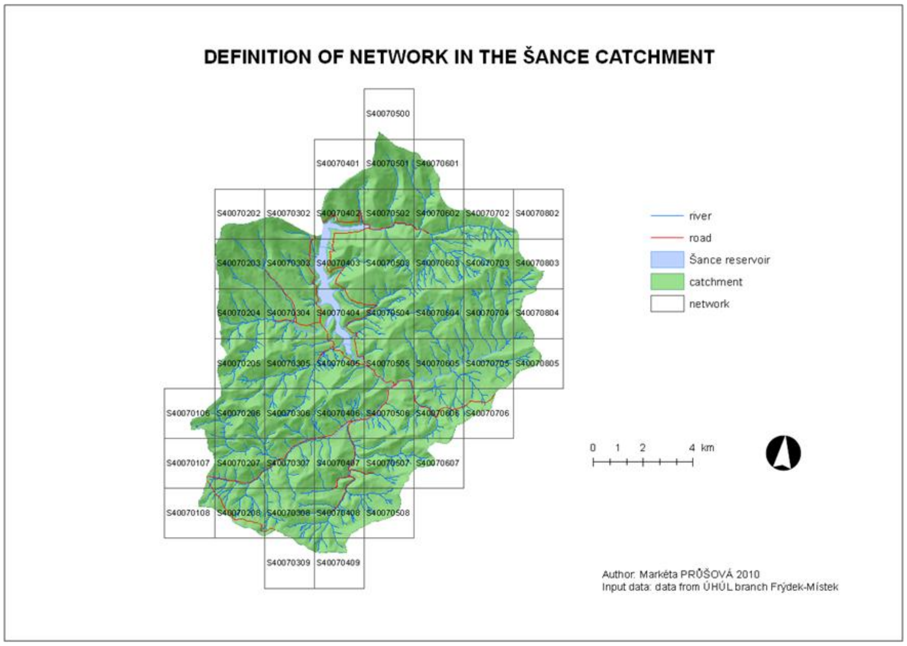


Fig. 1. Definition of network.

IV. Exploratory Data Analysis

It was decided on the basis of differences in the data (in the homogeneity file), and test consensus of the medians to obtain the corresponding results. It will be necessary to perform a statistical analysis for each area (open, forested) separately. Figure 2 demonstrates the systematic difference between average snow heights for two types of lands. The figure also depicts large differences in snow cover among years. It is therefore advisable to separate the processing of data from different measurement campaigns.

Result of exploratory analysis is the statistical analysis of snow cover parameters. All results are largely influenced by time (a year and campaign measurements). Appraise: the snow covers were abundant in 2006; therefore parameters for each snow have good statistical significance. On the contrary, the snow covers were poor in 2008; therefore statistics of measured data do not have a very high predictive ability. Most data has a moderately leftward distribution. In four cases, the snow density parameter has of rightward distribution. The box plot is part of processing. It shows both the progress of time and also the count of extreme outliers during the campaign. Progress was the highest for height of snow in 2006, and then decreased in 2007. Variability of water parameters is similar to the snow height. It can also be seen at the extreme parameter values, whose occurrence is due to the fact that large amounts of snow and the current weather conditions, that especially the melting and recrystallization processes have a significant effect on the local density of snow.

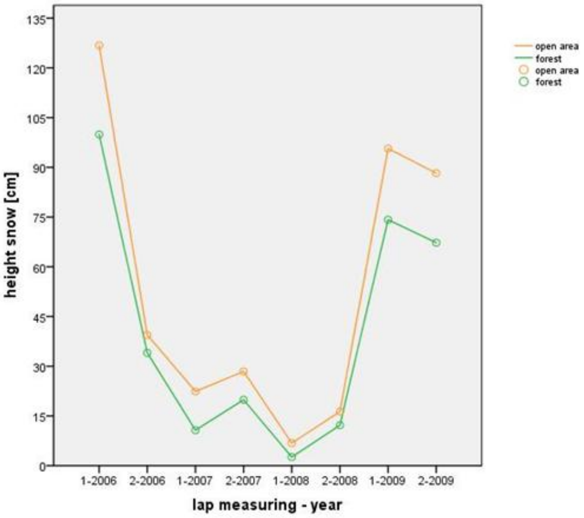


Fig. 2. Average of snow cover in time series.

A further part of EDA investigates normality of data. Most interpolation methods are based on linear estimates and require a normal distribution of sample data. If data fails in normality testing it is necessary to make an appropriate data transformation to reach the normal distribution. The following methods of transformation were tested: natural logarithm transformation, the transformation of the square and power transformation using a linear interpolation coefficient of skewness, which approximates the optimal value of the constant transformation estimate based on linear interpolation [9]. The last method is chosen as the best transformation.

During testing of the parameters of snow, it was found that altitude played the main role. This is particularly the existence of extreme outliers in Lysá Mountain and Smrk Mountain. The terrain factor in comparison with the individual characteristics of snow cover constitutes an important element in studying and evaluating the results.

Furthermore correlation and regression analysis were performed of relations between local morphological characteristics (altitude, slope and orientation) and snow

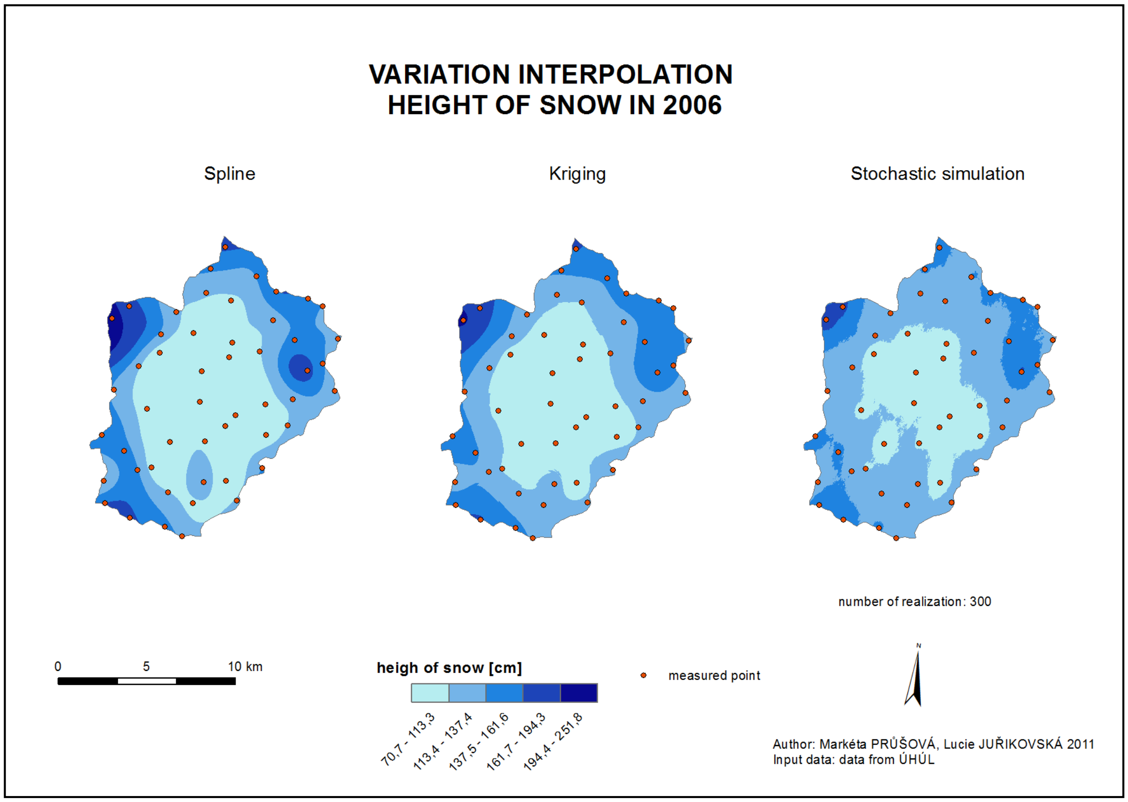


Fig. 3. Interpolation methods.

parameters. The results confirmed a clear dependence of the amount of snow on the altitude, and showed a partial dependence on slope and orientation.

V. STOCHASTIC SIMULATION

Interpolation is carried out separately for the open and forested areas, according to the results of exploratory statistics. We compared these methods: simple kriging, ordinary kriging, universal kriging, simple cokriging, ordinary cokriging and universal cokriging. Interpolation results visually compared the isolines. We examine development of the shapes of contour lines and their credibility, especially in border areas. Furthermore, the methods explored relationship of known (measured) values in the basin.

We selected the stochastic simulation for height of snow. It was chosen due to its better local estimation ability than classical interpolation methods. This claim can prove the following figure (Fig.3), from which it is evident. Simulation provides better and more exact results than the methods of approximation, which leads to smoothing values even with a known value.

The simulation was carried out with software products such as ArcGIS 10 and SGeMS.

The stochastic simulation was performed with software ArcGIS 10. Gaussian geostatistical simulation was chosen as the stochastic simulation. These parameters (Table I.) configured for simple kriging. Simple kriging is a necessary condition for simulation.

TABLE I. PARAMETERS OF VARIOGRAM

Parameter	Value
Lag	542 m (according to the minimal distance Among locations)
Nugget	0
Number of lags	8
Angle tolerance	22.5 °
Minimal Range	2500
Maximal range	4200
Direction of maximum range	22°
Direction of minimum range	112°

The next step was another parameter directly for simulation.

Settings:

1. Simple kriging
2. Number of realization: 300 (1000)
3. Input feature: snow cover
4. Conditional field: height of snow
5. Cell size: 10

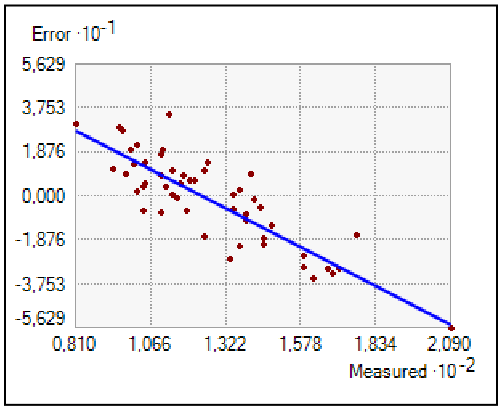


Fig. 4. Error plot.

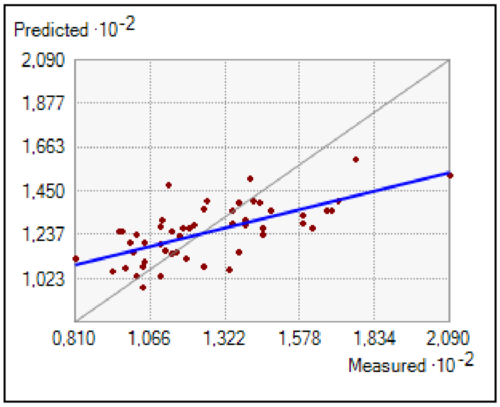


Fig. 5. Graph of prediction.

TABLE II. PREDICTION ERRORS OF SIMPLE KRIGING

Parameter	Value
ME	-1,646
RMS	19,302
ASE	18,961
MS	-0,065
RMSS	1,021

A stochastic simulation result follows in picture (Fig.6).

Of course, the higher number of realizations is simulation results more accuracy and better reflects the trend in the area. The example simulations are compared with the number 300 and 1000 implementation. The simulation with the number 1000 is seen repeating the beginning smoothing interpolation, but in comparison with the spline or kriging method it is a negligible problem.

We solve the stochastic simulation because of its good explanatory power at the measurement point.

Stochastic simulation generally gives better results than the conventional interpolation method. This assertion is based on the comparison of interpolation methods to stochastic simulations.

Stochastic simulation does not produce better results for the determination of the mean on the ground, but it provides the necessary opportunity to determine the probability of exceeding certain limits. These limits are important to the application area (for example, height of snow or water supply, causing a significant increase in crown fractures and fallen trees).

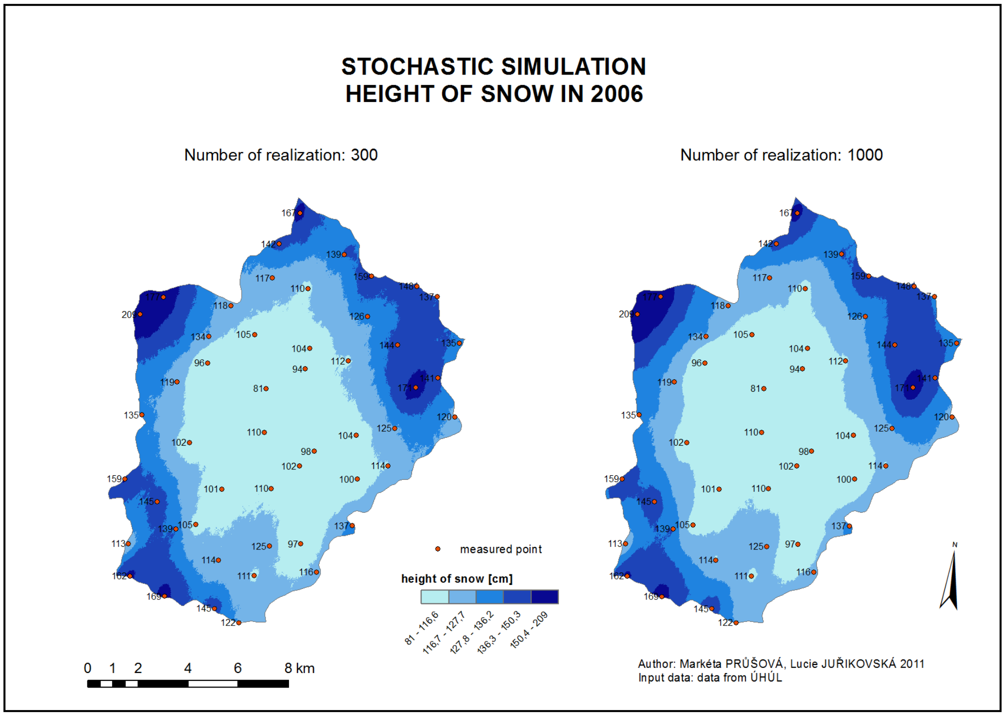


Fig. 6. Stochastic simulation in ArcGIS.

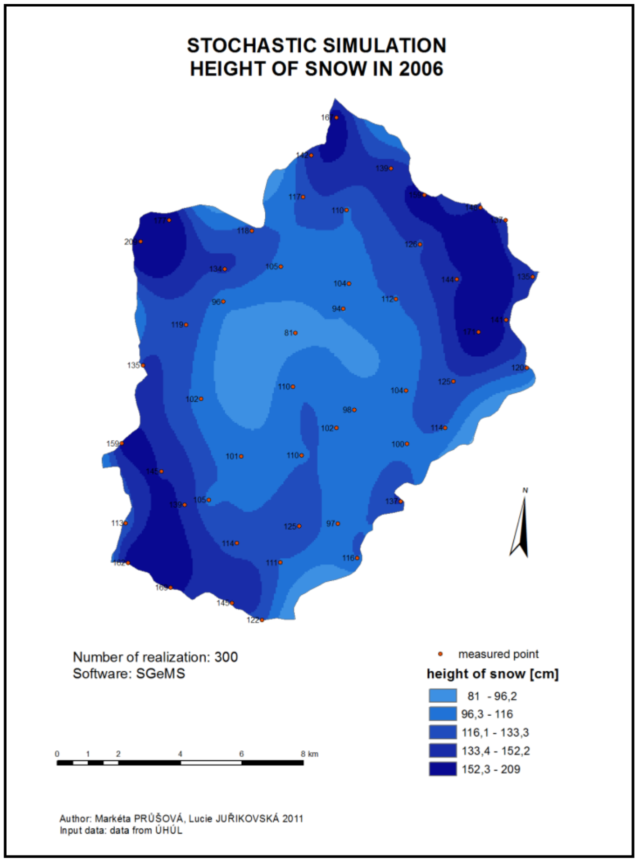


Fig. 7. Stochastic simulation in SGeMS.

VI. Conclusion and Future Work

Majority users use standard interpolation methods and takes at face value and we want to show that there are other methods that can provide better results. Of course, depends on the user which method he chooses.

When processing data, we should not forget the basic statistics of data. Some of these statistics are either completely omitted or performed it without important aspects such as testing the normality of data. This approach can completely distort the results and regardless of the choice of the best interpolation methods.

Stochastic simulation claimed his good properties in compared with others interpolation methods.

Future work will include the following enhancements to our approach: Creation conversion between two different programs for their simulation, finding the best way for compare resultant statistic with other interpolation methods and integrate information about damage of forest.

Acknowledgment

We would like to thank VŠB - Technical University of Ostrava for providing financial resources to this conference.

References

[1] ArcGIS Resources Centres, Key concepts of geostatistical simulation, 2010.

[2] B. Balk and K. Elder, " Combining binary decision and geostatistical methods to estimate snow distribution in a moutain watershed, " Water Resources Research, vol. 36, 2000, pp. 13-26.

[3] N. Cressie, Statistics for spatial data, Wiley, New York, 1991.

[4] C.V. Deutsch and A.G. Journel, GSLIB geostatistical software library and user´s guide, 2nd edition, Oxford University Press, New York, 1998.

[5] K. Elder, J. Dozier and J. Michaelson, "Snow Accumulation and distribution in an alpine watershed," Water Resour. Res., vol. 27, 1991, pp. 1541-1552.

[6] T.A. Erickson, M.W. Williams and A. Winstral, "Persistence of topographic control on the spatial distribution of snow in rugged moutain terrain, Colorado, United States, Water Resour. Res., vol. 41, 2005, doi 10.1029/2003WR002973.

[7] P. Goovaerts, Geostatistics for natural resources evaluation, Oxford University Press, New York, 1997.

[8] E.H.Isaaks and R.M.Srivastava, Applied Geostatistics, Oxford University Press, 1989.

[9] M. Kaňok, Statistical methods in management, Czech Technical University in Prague, 1996.

[10] N.P. Molotch, M.T. Colee, R.C. Bales and J. Dozier, " Estimating the spatial distribution of snow water equivalent in an alpine basing using binary regression tree models: the impact of digital elevation data and idependent variable selection, " Hydrological Processes, vol. 19, 2005, pp. 1459-1479.

[11] E. Pardo-Iguzquiza and M. Chica-Olmo, "Geostatistical simulation when the number of experimental data is small: an alternative paradigm," Stoch. Environ. Res. Risk. Assess, vol.22, Apr.2008, pp. 325-337, doi 10.1007/s00477-007-0118-1.

[12] A. Rango, "Spaceborne remote sensing for snow hydrology applications," Hydrological Sciences Journal, vol. 41, 1996, pp. 477-494.

[13] D.G. Tarboton, G. Bloschl, K. Cooley, R. Kirnbauer and C. Luce, Spatial snow cover processes at Kűhtai and Reynolds Creek, Spatial Patterns in Catchment Hydrology: Observation and modelling, Cambridge University Press, 2000, pp.158-186.

[14] R. Webster and M.A. Oliver, Geostatistics for environmental scientists, 2nd edition, John Wiley & Sons, 2007.

The Lambda Chart: A Model of Design Abstraction and Exploration at System-Level

Falko Guderian and Gerhard Fettweis
Vodafone Chair Mobile Communications Systems
Technische Universität Dresden, 01062 Dresden, Germany
Email:{falko.guderian, fettweis}@ifn.et.tu-dresden.de

***Abstract*—In recent years, chip design complexity is further increasing through multi-processor system-on-chip built up from macro blocks. System-level design promises to close the growing productivity gap between hardware and software design but more sophisticated design models are needed. Therefore, we developed a new model of system-level design abstraction. Therein, three views divide the design space to reduce design complexity. A unified design process with five steps has been defined for the design views. Furthermore, we show that the model is able to formalize system-level design exploration. Finally, exploration has been considered as walk through the design views connected via inter-view links. With the proposed system-level model, the authors provide conceptual foundations to more holistic design and introduce formalization of design exploration.**

***Keywords*-system; design; abstraction; exploration;**

I. Introduction

As integration technology continues to shrink and processor clock frequency stagnates, design complexity grows through the transition from traditional system-on-chip (SoC) towards multi-processor system-on-chip (MPSoC) built up from macro blocks. The increasing complexity motivates to use modeling and simulation languages, such as SystemC, which allow for simulation of the complete design at system-level including the hardware and software. We believe that more sophisticated models are needed to close the growing productivity gap between hardware and software forecasted by the International Technology Roadmap for Semiconductors [1]. Existing models of system-level design require the application and architecture description as starting point and apply exploration in similar design views and process steps. Therefore, we developed a model of system-level design abstraction to bring them more into line. Our model defines three design views, called administration, computation and communication, wherein a unified design process is followed. We introduce the administration view due to the growing importance of management, e.g., of scheduling, power consumption, reliability etc., in future MPSoCs and Many-Core systems. The separation into three views and the reuse of the design process aims at reducing design complexity. Furthermore, system-level design exploration tools become more important but separation and integration of design views and design process steps during exploration is hardly supported in current approaches. Hence, relationships between views and steps should be considered. We show how our model is able to formalize system-level exploration. More specifically, exploration is defined as walk through the design views via inter-view links representing the relationships. During exploration, each view applies one or more design process steps. Our approach is independent on the application domain and the formalization aims at automated exploration. Moreover, we derive three exploration types (classes) from our model and present their usage based on design examples.

In this work, we describe system-level design abstraction and exploration with regard to MPSoC design. Nevertheless, the authors see a general relevance of the model for complex systems based on networks. For example, computer networks, telecommunication networks and sensor networks are also composed of computation, communication and administration. Hence, these systems could be designed and explored using our model.

In the remainder of the paper, Section II provides an overview of the model of system-level design abstraction. In Section III, we compare our model with existing work. Then, Section IV describes the system-level design views. The system-level design process is explained in Section V. Section VI presents different types of design exploration based on system-level design abstraction. Finally, Section VII concludes our work.

II. Overview of System-Level Design Abstraction

In Figure 1, the tripartite representation called λ-chart is our model to realize system-level design abstraction. We use three axes to describe the design views: administration, computation and communication. Along each of the axes, we unified the design process to five steps which are given as concentric bands. The process starts with modeling and partitioning, e.g., of application, architecture and administrative algorithm. Provisioning describes the selection and dimensioning of system components, e.g., cores and processors respectively. As depicted in Figure 1, we separate scheduling and allocation. Finally, the system is validated to decide for an additional design iteration. Hence, the axial loop indicates the iterative design process within each design view.

The example in Figure 2 illustrates the relationship between λ-chart and Y-chart which was introduced by Gajski and Kuhn [2] and refined by Walker and Thomas [3]. We refer to the Y-chart in [3] which is a model of design representation and synthesis. It defines the structural, behavioral and physical domain description. “Structural” means the abstract implementation of the design, “behavioral” describes

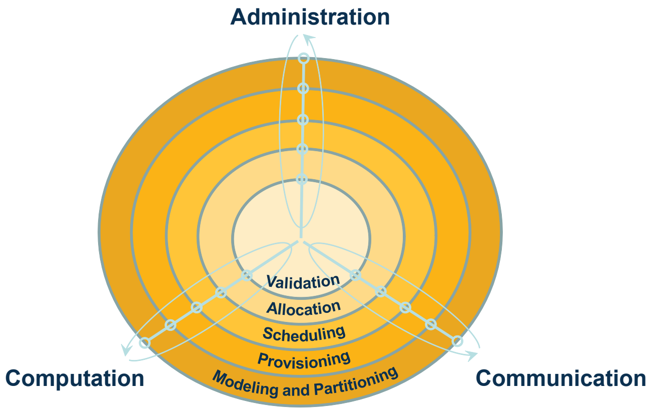


Fig. 1. The λ-chart: A Model of System-Level Design Abstraction.

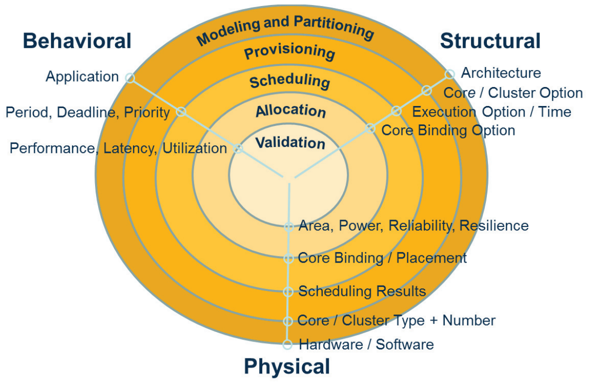


Fig. 2. Relationship between λ-chart and Y-chart [3] illustrated for the Computation View and our Unified Design Process.

the functionality of the design and "physical" relates to the physical realization of the abstract structure. Hence, synthesis is defined as transition from the behavioral model to the structural model ending in the physical realization. Figure 2 contains relevant domain descriptions for the computation view and the unified design process. In the example, not all process steps include a domain description. For example, provisioning is limited to the structural and physical description. In contrast to Y-chart synthesis, the λ-chart jointly uses structural and behavioral descriptions to realize the physical description. For example in Figure 2, the simulation of application and architecture creates scheduling and binding results to physically map application tasks to cores. In the λ-chart, each design view and process step is further characterized with the three domain descriptions. This is demonstrated in Section IV-VI. Moreover, the λ-chart can be considered as refinement of the architectural level within the Y-chart in [3]. This corresponds to the fact that our model addresses decisions at early design stages.

III. Related Work

Thomas et al. [4] present a model and methodology for mixed hardware-software design. According to given performance goals, the design process results in optimal hardware and software realizations. In contrast, our model aims at abstraction of system-level design. Therefore, we separate several views on the system to reduce design complexity. Software functionality is covered through high-level task graphs.

In [5], the authors propose interface-based design which abstracts design with decomposed components. Moreover a communication view is implicitly considered separating communication and component behavior. In addition to structural compositions we also decompose behavioral and physical descriptions independently on each other. Our approach is more generic because it uses communication as separate design view to support also less communication-centric designs.

Blickler et al. [6] define system-level synthesis as mapping of task-level specifications onto heterogeneous hardware/software architectures. They introduce a new formal definition for system-level synthesis which includes several process steps. Our model refines this approach to a unified design process applicable to several design views.

In [7], system design at different abstraction levels is proposed which reduces design complexity by separating concerns, such as function, architecture, computation and communication. The authors also use platform-based design to map applications to abstract representation of micro-architectures similar to [6]. In [8], the author proposes platform-based design as unified methodology applicable to future systems with heterogeneous subsystems, such as electronic and mechanical components. Despite our model supports platform-based design, we extend the work unifying the design process and also considering an administration view.

Gerstlauer and Gajski [9] have developed a design process starting with system specification to build the architecture by mapping communication and computation. The authors focus on model refinement between abstraction levels, e.g., system-level and algorithmic level. In addition, Kienhuis et al. [10] have developed a tripartite representation of system-level design. Therein, application models are mapped to architecture models and evaluated afterwards. We also consider modeling, mapping and evaluation in the design process. In general, our separation of design view and process allows a more generic description of system-level design and exploration.

IV. Views on System-Level Design

Our model of system-level design abstraction includes three design views: administration, computation and communication. Each view represents a separate portion of the total design space. Hence, inter-view links are necessary to synchronize with the other views. Although other divisions of the system-level design have been published, e.g., by [9] and [10], we believe that our division is the most natural. Thus, the administration view includes the design of planning, monitoring and control tasks of a system and its subsystems. In the computation view, all designs related to the code execution are covered. The design of data storage and data exchange between components is considered within the communication view. In the following, each design view is characterized in more detail via the domain descriptions.

A. Administration View

The "Administration View" considers all design tasks for planning, monitoring and control in the system and its subsystems. A structural description is the administrative architecture, e.g., central or distributed. In the behavioral description, the administrative algorithm would be realized either in a static or dynamic manner, such as static or dynamic scheduling. Considering physical descriptions, administrative units, such as hardware schedulers, are placed at a geometric position in the design. Hence, hardware-based vs. software-based administration corresponds to the physical realization.

B. Computation View

The "Computation View" considers all design tasks related to code execution. The computational architecture could be designed as central system or divided into subsystems (clusters). Further examples of structural descriptions are the decision for a heterogeneous or homogeneous set of processing elements (PEs) and subsystems (clusters). In contrast, the computational behavior is characterized by the degree of application parallelism or the number of inputs and outputs. Physical descriptions relate to binding and placement which typically take geometric and area constraints into account.

C. Communication View

The "Communication View" considers the design of data storage and data exchange between components, such as memory, router and peripherals. The topology of memory and network are examples of structural descriptions. Strategies for routing and data caching are considered as behavioral description. For the physical description, the dimensioning of links, routers (buffers) and memories are important realizations.

D. Further Characterizing the Design Views

The domain descriptions in the λ-chart include components and behavior in different levels of hierarchy. The hierarchical compositions occur in each design view and we divide them into structural, behavioral and physical compositions. This enables to further specify the design views and allows the formal representation of inter-view relationship, described further below. Figure 3-5 illustrate exemplary compositions for each design view as tree-based graph representations. Figure 3 shows an example of the administration architecture. Therein, a software-based application balancer supplies several hardware schedulers with application code. Hence, this describes structural and physical details of the design view. Referring to Figure 3, a behavioral composition is given for different scheduling strategies, such as As Soon As Possible (ASAP), Earliest Deadline First (EDF), priority-based etc. In Figure 4, structure and behavior of the computation view are depicted as hierarchical compositions. The system consists of several subsystems (clusters) which include either heterogeneous or homogeneous sets of PEs. A PE could be a RISC core, ASIC core, DSP core etc. Referring to Figure 4, the behavioral composition shows that sets of computation tasks are scheduled and that synchronization between these sets is necessary. Here, the term "task" means a computation kernel which is executable on a PE. Task examples are FFT, FIR, ENC, DCT etc. In Figure 5, the memory structure of the communication view is represented as hierarchy of main memory and local cache. In the physical composition, memory is realized as DDR3 and SRAM. An example for the behavioral composition is the selection of the switching technique, here guaranteed service, wormhole switching etc.

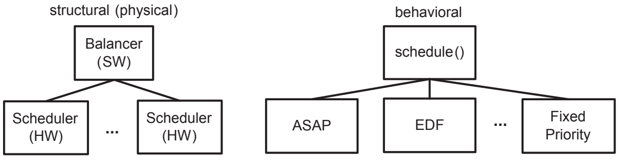


Fig. 3. Hierarchical Compositions in the Administration View.

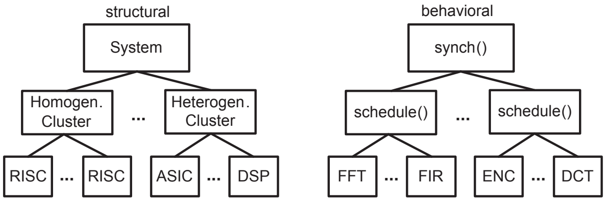


Fig. 4. Hierarchical Compositions in the Computation View.

In general, relationships between the design views exist. For example, the structure of communication and computation are closely coupled as depicted in Figure 6. Therein, all subsystems (clusters) access the main memory and each PE includes a local cache. Hence, formal representations are necessary to integrate inter-view relationships into system-level design. Referring to Figure 6, graph representations are suited to model the relationships between design views.

E. Importance of the Administration View for Dynamic Behavior

We introduce the administration view in system-level design abstraction because administrative tasks and the design of such hardware units become important in future MPSoC and Many-Core systems. For example, administration units can be used to improve reliability, power consumption, programmability, product reuse etc. In general, administration can handle static

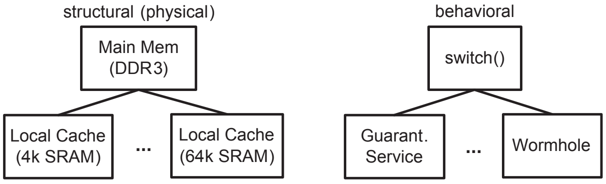


Fig. 5. Hierarchical Compositions in the Communication View.

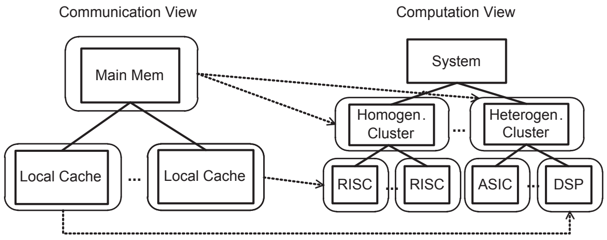


Fig. 6. Relationship between Computation and Communication.

TABLE I
LIMITATIONS FOR STATIC ADMINISTRATION OF DYNAMIC BEHAVIOR

No adaption to	Computation	Communication	Administration
Failure/ Resiliency	Processing element	Router, Link, Memory	Administrative unit
Hotspots	Thermal issue	Transfer bottleneck	Monitoring/ control bottleneck
System/ User changes	Performance scaling, Energy saving		Non-deterministic task dependency
Data dependent control structures	Non-deterministic operation		
Concurrent applications			

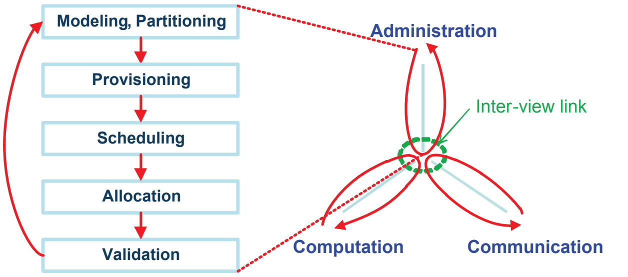


Fig. 7. Unified System-Level Design Process.

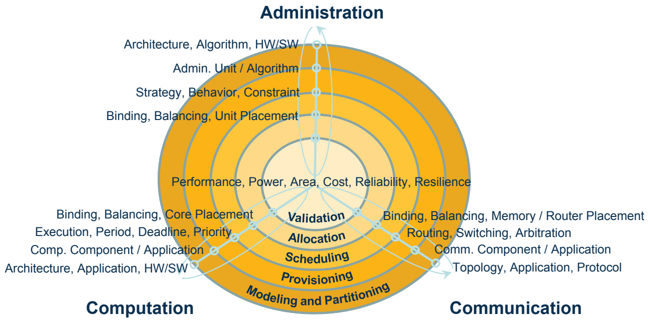

Administration
Architecture, Algorithm, HW/SW
Admin. Unit / Algorithm
Strategy, Behavior, Constraint
Binding, Balancing, Unit Placement
Performance, Power, Area, Cost, Reliability, Resilience
Binding, Balancing, Core Placement
Execution, Period, Deadline, Priority
Comp. Component / Application
Architecture, Application, HW/SW
Validation
Allocation
Scheduling
Provisioning
Modeling and Partitioning
Binding, Balancing, Memory / Router Placement
Routing, Switching, Arbitration
Comm. Component / Application
Topology, Application, Protocol
Computation
Communication

Fig. 8. λ-chart refined by Domain Descriptions for the Design Views and Design Process Steps.

and dynamic behavior with static or dynamic mechanisms. If the changing conditions are known in advance, dynamic behavior can be administrated statically. Therefore, periodic applications with fix schedules are statically administrated, such as in data-flow driven communication protocols. As MPSoCs become subject to unpredictable dynamic behavior (see Table I), static administration is not suited anymore. Hence, dynamic administration increases design complexity which underlines the importance of the administration view. Table I lists limitations of static administration for dynamic behavior. It shows that components in the computation, communication and administration view can hardly adapt to unpredictable situations like failure/resiliency, hotspots, system/user changes, data dependencies or application concurrency. If components, such as a PE, memory, router, link or administrative unit, fail or require a certain fault tolerance, static administration is poorly able to react to this situations. Other examples are hotspots, such as thermal issues and bottlenecks, which can temporarily occur and would require dynamic management. Referring to Table I, high-performance or energy-saving modes are examples for unpredictable user and system changes. This can only be effectively handled via dynamic administration. Furthermore, static administration shows weaknesses in non-deterministic operation which occurs due to data dependent control structures and concurrent applications. Moreover, non-deterministic tasks dependencies can only be resolved via dynamic administration.

V. SYSTEM-LEVEL DESIGN PROCESS

The λ-chart defines five steps unifying the design process to: modeling and partitioning, provisioning, scheduling, allocation and validation. Prior to that, the design goal must be defined, e.g., low-power or high-performance. Figure 7 shows that the process steps are independently followed in each design view but the views have to be synchronized via inter-view links. This section describes all five process steps which have been further characterized by structural, behavioral and physical domain descriptions, as illustrated in Figure 8. Therein, the design views and design process are detailed by components, behavior, properties and design goals which relate to a domain description. Modeling and partitioning are limited to behavioral and structural descriptions because the process step does not intend a physical realization. Hence, provisioning, scheduling and allocation increasingly contain physical descriptions due to the elapsed design progress. Finally, validation must be restricted to physical descriptions since it represents the last design process step. Unfortunately, terminology is not consistently used amongst the researchers, e.g. the term allocation might also include provisioning for some people. Because we do not intend to confuse the reader by inventing new names, our model considers the most accepted naming to the authors best knowledge.

A. Modeling and Partitioning

Modeling and partitioning describe the first step in the design process which is to build formal representations of the system structure and behavior. In addition, partitioning underlines the importance of modeling hierarchical structures and concurrent behavior. Representations of the application and architecture should be adequate for an automated design flow. Therefore, graph representations are widely used and applied in the administration, computation and communication views. Referring to Figure 8, our model accounts for structural descriptions of system-level design by describing the topology (architecture). This also includes the modeling of hardware and software supported application functionality, known as HW/SW codesign. In addition, application and administrative algorithm represent behavioral descriptions.

B. Provisioning

Provisioning is the second step in the design process and basically means to select the type and number of components and behavior necessary to fulfill the purpose of the system under design. Hence, provisioning chooses structural and behavioral descriptions, e.g., cores, routers, application tasks and administrative algorithms, to increasingly create the physical realizations. For the administration, computation and communication view, different provisioning decisions are needed. These decisions include the reuse of existing IP, the design from scratch or an intermediate solution. Provisioning considers also different design hierarchies, such as number and type of subsystems (clusters). Usually, the decisions are determined by the availability and experience of designers, the cost of IP and the reuse of existing designs. Nevertheless, other criteria such as tool support or user and system requirements might influence the choice.

C. Scheduling

The scheduling step represents the temporal planning of the application and component behavior, such as execution, communication, monitoring and power mode. Referring to Figure 8, structural and behavioral descriptions include components, behavior and properties related to scheduling. A physical description is the realization either in software or hardware. Administration, computation and communication design could be closely coupled because administrative units are responsible for the scheduling of computation and communication behavior. For example, a scheduling algorithm jointly aims at improved computation performance and low communication latency. In that case, additional monitoring and control functionality is required in all three design views. Furthermore, priority based scheduling represents a widely used technique to account for real-time behavior. In that case, priorities must be considered as behavioral description.

D. Allocation

The allocation step focusses on spatial planning of the application, architecture and administrative algorithm. Allocation is considered in all design views. It requires structural and behavioral descriptions, such as units and algorithms for component binding and application balancing. A physical description is the assignment of application code to components, such as tasks to cores. Furthermore, balancing tries to level component usage to improve performance and reliability. Referring to Figure 8, physical description is also represented by the geometric placement of components, such as of cores, memories, routers or administrative units. Allocation also accounts for techniques aiming at power reduction, such as power/clock gating or frequency scaling. In general, scheduling and allocation must be closely coupled to improve the design goals.

E. Validation

Validation represents the last process step and it proves whether the system fulfills the previously defined purpose or not. As prerequisite of the validation, design execution is either based on an analytical, simulative or hybrid approach. During execution, adequate evaluation data is aggregated for further analysis. Finally, design validation is performed according to the given objectives and constraints. In case the current system misses the design goal, relevant representations and design decisions are adapted by means of an additional design iteration. This can be separately done for each design view. Validation is applied to the physical realizations. Hence, structural and behavioral descriptions can only be examined by means of their physical realization. For example, insufficient computation performance leads to changes in prior design steps and relate to one or more design views.

VI. System-Level Design Exploration

The model of system-level design abstraction is also suited to describe system-level design exploration. We introduce the following exploration types to be able to operate both with separated and integrated design views:

- single view,
- view-to-view and
- all views integrated.

The term design exploration denotes to systematically alternate design parameters with the goal of finding systems that fulfill the intended purpose. In Figure 9-11, we present the design exploration types based on three design examples.

Figure 9 illustrates single view exploration which is not necessarily limited to information of the explored view. In our case, administration delay and transfer time has been additionally provided after modeling and partitioning. Referring to Figure 9, all design process steps are followed in the computation view. In contrast, only “modeling and provisioning” is rudimentarily considered in the remaining views. In general, focusing to a single view simplifies exploration at the cost of limited coverage of the overall design space.

In Figure 10, an example of view-to-view exploration is shown. Therein, the computation view provides a fixed configuration which serves as basis for the exploration of the communication. The administration view delivers only basic information, such as strategy/behavior of scheduling and routing. View-to-view exploration realizes limited interaction between the views during the design process. This allows the exploration to improve design space coverage within a specific design view. Nevertheless, observation of design space in the residual views remains limited.

Single view and view-to-view exploration are either suited for designs dedicated to a specific view or to designs with a small

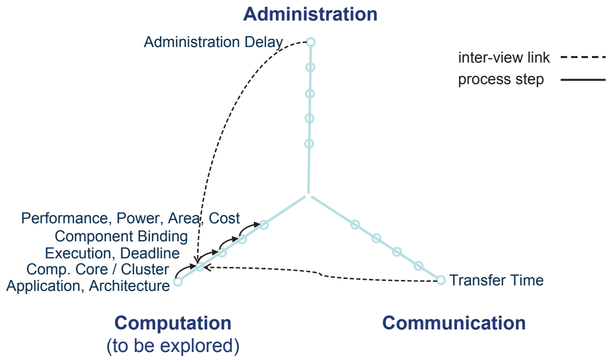


Fig. 9. System-Level Design Exploration - Single view.

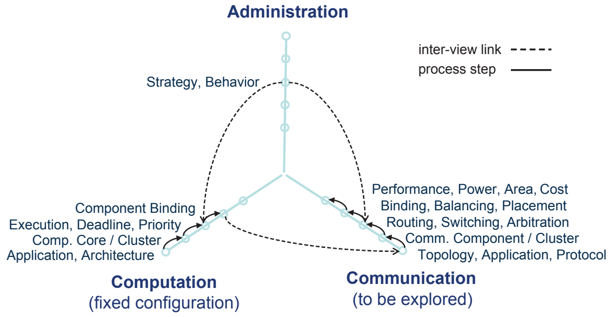

Administration
inter-view link
process step
Strategy, Behavior
Component Binding
Execution, Deadline, Priority
Comp. Core / Cluster
Application, Architecture
Performance, Power, Area, Cost
Binding, Balancing, Placement
Routing, Switching, Arbitration
Comm. Component / Cluster
Topology, Application, Protocol
Computation
(fixed configuration)
Communication
(to be explored)

Fig. 10. System-Level Design Exploration - View-to-view.

amount of interacting components and system functionality. An example would be to limit exploration to the performance analysis of IP cores assuming a single shared bus architecture. The design of complex systems, such as MPSoC, imply wider functionality, more interaction and dependencies between the three design views. A MPSoC is based on an on-chip network and different memory types. It has often several administration components, such as scheduler and performance monitor. Moreover, computation is enabled by an arbitrary number of cores. In addition, an MPSoC structure could be composed of several hierarchies, such as subsystems (clusters) for computation, communication and administration. Therefore, exploration integrating all views, as shown in Figure 11, is needed to create feasible MPSoC designs. Therein, the three design views are closely coupled. Several inter-view links allow the synchronization of results in each view and process step. Referring to Figure 11, exploration starts in the computation view which serves the communication view. In the example, the administration view requires input from the other views. But it also provides design decisions from the scheduling step towards the other views, such as execution schedule, routing strategy etc.

The exploration types are combined to form different exploration strategies. For example, exploration starts with the integrated type to initially create feasible design solutions. Afterwards, single-view or view-to-view exploration allow the optimization towards certain design goals within a design view and process step. Hence, the refinement should be guided with

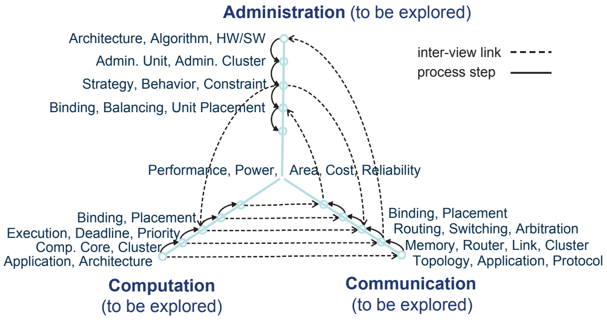

Administration (to be explored)
Architecture, Algorithm, HW/SW
Admin. Unit, Admin. Cluster
Strategy, Behavior, Constraint
Binding, Balancing, Unit Placement
inter-view link
process step
Performance, Power, Area, Cost, Reliability
Binding, Placement
Execution, Deadline, Priority
Comp. Core, Cluster
Application, Architecture
Binding, Placement
Routing, Switching, Arbitration
Memory, Router, Link, Cluster
Topology, Application, Protocol
Computation
(to be explored)
Communication
(to be explored)

Fig. 11. System-Level Design Exploration - All views integrated.

the validation results.

VII. Conclusion and Future Work

Increasingly complex chip design, such as MPSoC design, motivates us to contribute to the better understanding of design and exploration at system-level. Therefore, we developed a new model of system-level design abstraction. It reduces design complexity via abstraction to three design views and relationships between views have been considered. Furthermore, a unified system-level design process was introduced which is applicable to the three design views. As exploration becomes more important, the authors show how the model is used to formalize design exploration at system-level. This paper provides conceptual foundations to more holistic system-level design and introduces formalization of system-level exploration by giving examples. In future work, we will use our model for automatic system-level exploration. Ideally, the model will be proven within a real MPSoC design project.

References

[1] ITRS. (2011, Apr.) International technology roadmap for semiconductors 2010 edition. [Online]. Available: http://www.itrs.net

[2] D. D. Gajski and R. H. Kuhn, "New vlsi tools," *Computer*, vol. 16, pp. 11–14, Dec. 1983.

[3] R. A. Walker and D. E. Thomas, "A model of design representation and synthesis," in *Proc. of DAC*, 1985.

[4] D. Thomas, J. Adams, and H. Schmit, "A model and methodology for hardware-software codesign," *Design Test of Computers, IEEE*, vol. 10, no. 3, pp. 6 –15, Sep. 1993.

[5] J. A. Rowson and A. Sangiovanni-Vincentelli, "Interface-based design," in *Proc. of DAC*, 1997.

[6] T. Blickle, J. Teich, and L. Thiele, "System-level synthesis using evolutionary algorithms," *Design Automation for Embedded Systems*, vol. 3, pp. 23–58, 1998.

[7] K. Keutzer, A. Newton, J. Rabaey, and A. Sangiovanni-Vincentelli, "System-level design: orthogonalization of concerns and platform-based design," *CAD of ICs and Systems*, vol. 19, no. 12, pp. 1523 –1543, Dec. 2000.

[8] A. Sangiovanni-Vincentelli, "Is a unified methodology for system-level design possible?" *Design Test of Computers, IEEE*, vol. 25, no. 4, pp. 346 –357, Jul.-Aug. 2008.

[9] A. Gerstlauer and D. Gajski, "System-level abstraction semantics," in *Proc. of System Synthesis*, 2002.

[10] B. Kienhuis, E. F. Deprettere, P. v. d. Wolf, and K. A. Vissers, "A methodology to design programmable embedded systems - the y-chart approach," in *Proc. of SAMOS*, 2002.

Increase of Robustness on Pre-optimized Production Plans Through Simulation-based Analysis and Evaluation

Christoph Laroque
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany
laro@hni.uni-paderborn.de

Robin Delius
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany
robin.delius@hni.upb.de

Jan-Hendrik Fischer
University of Paderborn
Paderborn, Germany
jhfischer@gmail.com

Dennis Horstkämper
University of Paderborn
Paderborn, Germany
dhorstkemper@gmail.com

***Abstract*—We propose the use of the material flow simulation to evaluate the robustness of a production plan, which was created and optimized with no respect to unforeseen derivations. Since the necessary probabilities for machine failures and similar operational events on the floor can easily be integrated in the simulation model, in order to analyze, how initial plan performs in these situations. The influence of unforeseen events in daily production cannot be modeled within mathematical optimization without consuming large amounts of computation time. We show a possible way to use simulation to evaluate and enhance a production plan. We illustrate the developed process using a real-world use-case of medium complexity and can show, that simulation is able to evaluate the robustness of a given pro-optimizes production plan.**

Keywords: material flow simulation; robustness; production planning; mathematical optimization

I. MOTIVATION

Even after overcoming the global economic crisis tremendous requirements exist within the daily operation of a production facility and its supply chain. Fluctuating demands are leading to less adaequate forecast data and the need to lower capital commitment is leading to the necessarily of designing robust production planning models [1],[5],[6]. It is always the intention to be able to serve all demands in due time while causing minimal costs.

Several uncertainties exist within the production planning process. On the one hand, many unforeseen events can take place: machine failures, missing materials, changed sales demands or ill employees are only a small subset of possible examples. On the other hand, it is simply impossible to include all factors that might occur into the planning process in the first place. Therefore, planning methods are always based on different models of a production structure, which are an abstraction of reality themselves. It is the responsibility of the production planner to decide which factors he wants to take into the account when creating his models. He always has to find a compromise between the detail level of the model (and therefore its significance) and the solvability of the optimization problem which is created on its basis. The lot sizing and scheduling problems that are used within production planning are usually already np-complete even in their simplest form [15]. Therefore, one cannot guarantee to be able to find acceptable solutions in a timely manner while using modern operation research techniques. Thus, we have to find a solution to include the aforementioned uncertainties within the production planning process without limiting its solvability significantly. We connect a mathematical optimization model with a down streamed material flow simulation for this purpose. While we always assume optimal conditions within the mathematical optimization model, we are including the uncertainties in the simulation process. This allows us to analyze whether a production plan is able to perform well creating an acceptable monetary solution under these changed conditions or not. We create a sensible scheduling using rule-based machine controls within the simulation. In addition, we are able to create automatic or manual modifications of the plan and can evaluate these as well using additional simulations. It is easily possible to develop a more robust production plan with these tools.

Simulations usually are used to verify the solutions of an optimization problem. However, the aim of our research is to replace parts of the optimization process with simulation methods to receive solutions with an acceptable quality on a timely matter. First, we solve a mathematical optimization problem with standard solver software like IBM ILOG CPLEX [17]. Figure 1 shows the general optimization and simulation process.

After regarding the necessary State-of-the-Art in Section II, we describe the production model and the corresponding optimization models in Section III. It is possible to include uncertainties in the planning phase within the mathematical optimization process. We briefly discuss these methods in Section IV. To generate a more robust production plan based upon a given near optimal plan we propose a procedure which generates and evaluates a number of scenarios with the help of off-line simulations to create a new plan. We explain the transfer of the optimization solutions into the simulation process in Section V. To cover a broad spectrum of stochastically possible scenarios; several replications of the stochastic simulation based upon the production structure are performed. This way we are able to cover a wide field of possible scenarios for machine failures and other events.

ISBN: 978-1-61208-169-4

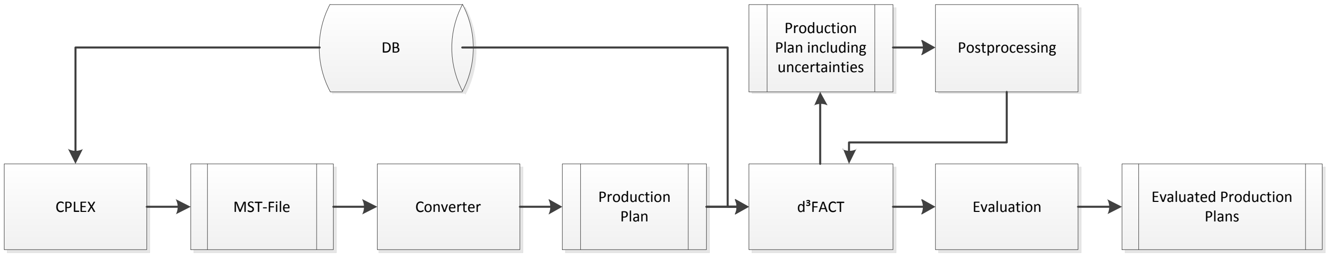


Figure 1: General Structure of presented concept

The production schedules are logged and afterwards evaluated on the base of costs and robustness. A rule-based machine control is used, to try to reduce possible production losses when intermediate products were not assembled in due time. An additional post-processing can be used to maintain further robustness increasing actions. The effect of these actions can be evaluated using further simulations. We present these processes in Section VI. We finally evaluate the outcome of our work using a case study. Additionally, we give a conclusion (Section VII) and an outlook towards further possibilities and improvements for this approach.

II. State of the Art

An ideal environment, free from external influences as used in most scheduling approaches is normally not given when processing a production plan. Production settings are subject to influences from human and machine failures. Additional resources and materials might not be available in due time and new demands often have to be taken into account on a short-term notice. A comprehensive overview about the execution of production plans under uncertainties is given by Aytug et al. [2]. They develop a taxonomy to classify uncertainties, to be able to classify numerous facets of disturbances within operational procedures. These are characterized by four dimensions:

- Cause (e.g., machine failure)
- Context (e.g., materials have not been delivered)
- Effect (postponed starting times)
- Inclusion (reaction upon interruptions, either predictive or reactive) [2]

These aspects illustrate uncertainties within the production planning process. The effect of disturbances and interruptions depends upon the robustness of the scheduling. Schneeweiß [15] gives a basic definition of a robust plan: A plan is robust, when it is insensitive against random environmental influences. Based on this expression one cannot find any quantitative measurements however. Scholl [16] expanded upon this definition. We mainly consider two of the criteria he developed: If a plan is always valid, no matter what environmental influences may effect it, it is called “total validity robust”. One cannot assume to reach this level in practical applications though. Therefore, one is able to analyze the validity robustness in greater detail instead of using a binary value. One could analyze the amount of broken model restrictions or also weight them after their importance. Within production planning, it is especially important to stay within the machine capacities and to adhere to given deadlines. We can consider the objective function of the planning models as the result of a production planning process. Therefore, one can define the criteria of result robustness: A plan is result robust, when its result only differs in a minimal way from the original plan when random environmental influences occur. However, a good result for one scenario may often lead towards a bad result for another scenario. Additionally result and validity robustness conflict with each other: a higher validity often causes higher costs.

Simulations can fulfill two roles within robust production planning: on the one hand, one can use a simulation to simply assess and evaluate the robustness of a plan to confirm the validity of other approaches to create robust production plans. On the other hand they can be used to create robust production plans to include uncertainties.

Aytug et. al [2] identified three main approaches in prior literature to create robust production plans: completely reactive procedures, robust scheduling and predictive-reactive scheduling. Completely reactive procedures only take action when disturbances in the production process already occurred. They sort and filter all jobs given to the current machine and continue with the job that appears to be the best based on this evaluation.

Robust scheduling approaches instead are creating plans, which minimize the effects of disturbances within the production procedure. Therefore, a plan for a worst-case scenario is created. Such a plan aims to be able to be processed in many different scenarios without greater difficulties. Both of these approaches share the issue, that available capacities will not be used to their full extend.

A large amount of research happens within the area of predictive-reactive scheduling. First, a plan for the whole planning horizon is created. This plan will be adapted later on. This can happen in a periodic fashion, on the occurrence of new events or in combination of both methods. In practice, these hybrid approaches are mostly used [12], [7].

Simulations are a standard tool to evaluate the robustness of production plans. This can be done based upon different target measures. Honkomp et al. [10] compare a basic deterministic simulation with multiple stochastic replications. To measure the robustness they use metrics that either compute the relation between the average objctive function of the stochastic simulations and the deterministic objective function or calculate the standard deviation of the stochastic simulations towards the best deterministic objective function. Apart from cost analysis, Pfeiffer et al. [13] also consider the plan efficiency and stability. This is also done in the overview about rescheduling approaches. Usually one obtains simple efficiency measurements (e.g., delays, backlogging amounts and production times). One can also evaluate these values visually [8]. Plan changes caused by stochastic events are processed to optimize the efficiency values. However, effects of changes within the scheduling are not taken into account within these approaches. Instead of optimizing the efficiency values one might also aim to create plans that only differ minimal from the original plan. A framework to evaluate different techniques to generate robust production plans has been developed by Rasconi et al. [14].

III. Production Model

To receive meaningful results we base our work on a close to reality production model with a corresponding complexity. Leaned upon a company in the supply industry of average size the model contains 21 machines with a general production structure, meaning that converging, diverging and linear substructures appear. Some of the 44 products can be produced on several machines in a parallel matter. This may possibly lead to different production and setup times as well as costs. 11 products with external consumer demands exist in total. Based on this assumption, a high degree of freedom exists, when a concrete production plan shall be created. Figure 2 shows the overall machine plan and material flow of the production model.

Typically, two different optimization models are used to create a production plan. Initially we calculate the lot sizes using a Multilevel Capacitated Lotsizing Problems (MLCLSP) based upon macro periods. Subsequently one creates a plan based upon micro periods using a Discrete Lotsizing and Scheduling Problem (DLSP) to determine exact production timings. As a result, the order in which the machines process their corresponding lots is decided.

A. *Lotsizing*

To determine the production amounts for each given period we use a MLCLSP in this paper. The basic version of the MLCLSP, as described by Tempelmeier and Helber [9] develops a cost optimal multiperiod production plan based on given demands, production costs, setup costs, inventory costs and machine capacities. For this purpose the optimization problem tries to take advantage of possible synergy effects that occur when production lots for several demands are combined, creating less need for setup processes. In contrast, this might create capital commitment and inventory costs when products are created in an earlier period. Therefore, a compromise between these factors has to be found. The model considers machine capacities in particular. Each machine can only be operated for a limited amount of time per period, for example for one or several working shifts. This does force an inventory increase.

The MLCLSP is a model based on macro periods. Therefor it only determines which amounts of which products are produced on which machine in every given period. The model explicitly does not determine a lot scheduling. To reproduce dependencies between different products lead times are used. If a product needs another product from an earlier production level as an input, it has to be produced in an earlier period. A production of intermediate products is triggered whenever a final product is created. A bill of materials is used to determine the needed amounts.

The MLCLSP we are using contains several enhancements over the basic models used in most literature. Several additional constraints are used to comply with the complexity of real production planning. Additionally to the standard model, we allow backlogging for products that have a direct external demand. Products can be manufactured on several machines in a parallel matter. We include transport lots and the machine capacities are determined upon a flexible work shift model. The mathematical formulation of the used model is as follows:

Model MLCLSP:

Minimize O = $\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{T}(s_{kj} * \gamma_{ktj} + h_k * y_{kt} + p_{kt} * q_{ktj} + bo_{kt} * i_k + b_{jt} * pc_{jt})$

Under the Constraints:

$$y_{k,t-1} + q_{k,t-1} - \sum_{i\in N_k} a_{ki} * q_{it} - y_{kt} + bo_{k,t+1} - bo_{kt} = d_{kt} \qquad \forall k \in K, \forall t \in T \qquad (4.1.1)$$

$$\sum_{k\in K_j}(tb_{kj} * q_{kt} + tr_{kj} * \gamma_{ktj}) \le b_{jt} \qquad \forall j \in J, \forall t \in T \qquad (4.1.2)$$

$$q_{kt} - M * \gamma_{kt} \le 0 \qquad \forall k \in K, \forall t \in T \qquad (4.1.3)$$

$$q_{kt}, y_{kt}, c_{ktj} \ge 0 \qquad \forall k \in K, \forall t \in T \qquad (4.1.4)$$

$$y_{k0} = 0;\ y_{kT} = 0 \qquad \forall k \in K \qquad (4.1.5)$$

$$\gamma_{ktj} \le avail_{kj} \qquad \forall k \in K, \forall t \in T, \forall j \in J \qquad (4.1.6)$$

$$\gamma_{ktj}, s_{jt}^{0}, s_{jt}^{1}. s_{jt}^{2} \in 0,1 \qquad \forall k \in K, \forall t \in T, \forall j \in J \qquad (4.1.7)$$

$$q_{ktj} = c_{ktj} * cont_k \qquad \forall k \in K, \forall t \in T, \forall j \in J \qquad (4.1.8)$$

$$bo_{kt} \le bomax_k \qquad \forall k \in K, \forall t \in T \qquad (4.1.9)$$

$$b_{jt} = 0 + s_{jt}^{0} * 480 + s_{jt}^{1} * 960 + s_{jt}^{2} * 1440 \qquad \forall j \in J, \forall t \in T \qquad (4.1.10)$$

$$s_{jt}^{0} + s_{jt}^{1} + s_{jt}^{2} = 1 \qquad \forall j \in J, \forall t \in T \qquad (4.1.11)$$

In the objective function the sum of setup-, stock-, production-, backlog and personal costs are minimized. The following constraints enforce the creation of a valid production plan which fulfills external demands in due time whenever possible.

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-169-4

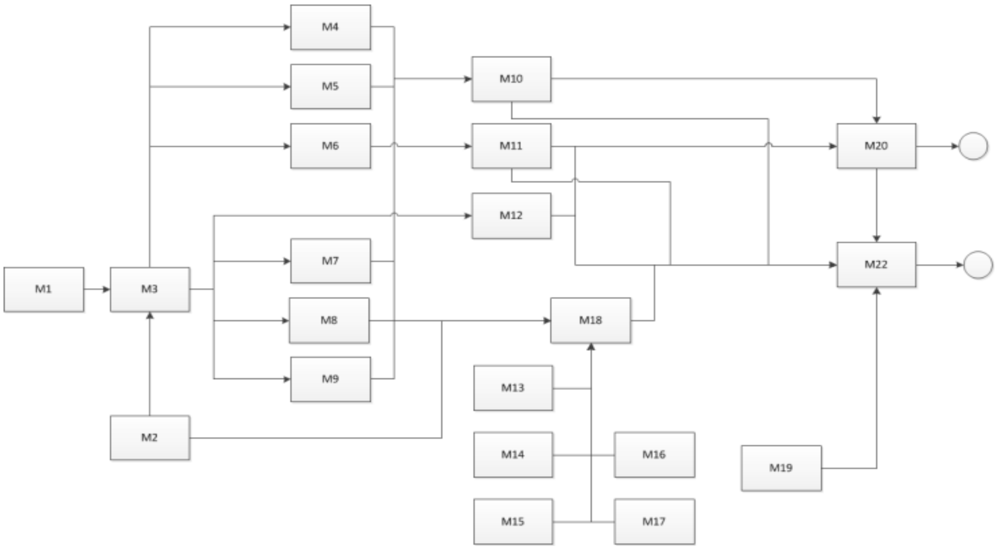


Figure 2: Machine Plan

Constraint 4.1.1. creates a balance between external demands on one side and production- stock and backlog amounts as well as secondary demands on the other side. To be sure that intermediate products are assembled before the final product is created, products must be created a day before the secondary demand takes place. Machine capacities are taken into account in constraint 4.1.2. It is only possible to perform a limited amount of production and setup activities within a single period. Using constraint 4.1.3. ensures that one can only produce a product on a machine when a machine is was set up for that product.

Additionally constraint 4.1.6. ensures that machines can only produce products that they can be set up for. Constraint 4.1.8 expresses that production lots always have to be a multiple of transport lots. Within constraint 4.1.9, maximum backlog amounts for each product are defined. This way we can ensure that demands for intermediate products cannot be backlogged. The constraint 4.1.10 and 4.1.11 determine the amount of working shifts used for a machine in a certain period. The other constraints are used to design meaningful bounds to the variables, for example, stock amounts always have to have a positive value.

Variables and constants meanings:

a_{ki}	Direct demand coefficient of products k and i
b_{jt}	Available capacity of resource j in period t
d_{kt}	Primary demand for product k in period t
pc_{jt}	Personal costs for resource j in period t
h_k	Stock expense ratio for product k
i_k	Penalty costs for backlogging of product k
J	Amount of Resources (j= 1,2,…,J)
K_j	Index set of operations performed by resource j
M	Big number
N_k	Index set of followers of product k
p_{kt}	production costs of product k in period t
q_{ktj}	Production amount of product k on resource j in period t
c_{ktj}	Amount of containers of product k processed by resource j in period t
$cont_k$	Container size/Transport lot size for product k
s_{kj}	Setup costs for product k on resource j
T	Length of planning horizon measured in periods (t=1,2,…,T)
tb_{kj}	Production time for product k on resource j
tr_{kj}	Setup time for product k on resource j
y_{kt}	Stock for product k at the end of period t
γ_{ktj}	Binary setup variable for product k on resource j in period t
bo_{kt}	Backlog variable for product k in period t
$bomax_k$	Maximal backlog amount for product k (always 0 for intermediate products)
$s_{jt}^0, s_{jt}^1, s_{jt}^2$	Binary variables used to calculate the amount of used working shifts

B. *Scheduling*

Using a DLSP one can assess a plan based upon micro periods to determine exact production timings. The solutions of the MLCLSP can be used as parameters for the DLSP. This way one can create a complete machine scheduling plan. A basic version of the DLSP can be found at Fleischmann [11]. The production amounts within a period that have been determined using the MLCLSP can be used as external demands for the DLSP. Periods within the DLSP are chosen as the smallest meaningful unit, for example the smallest common denominator of setup- and production times. The MLCLSP includes lead times; therefore, it is not needed to take dependencies between production levels into account. Hence, we can solve the DLSP for each machine individually. This means that the solution times are rather short. The problem complexity is appropriately low. We do however not include a DLSP within our work, as we use a rule-based machine control to create a scheduling plan within the simulation in an even shorter amount of time.

IV. FUZZY PARAMETERS IN THE MODEL CREATION

Fuzzy parameters and uncertain information can be reproduced using stochastic methods inside the model classes we described earlier. Ideally, we already know exact probabilities for possible events in advance. Where applicable we can use appropriate prognosis methods to estimate this probabilities. Otherwise, we can only use a normal or similar distribution.

The stochastic optimization tries to find a solution that is the best for all possible combinations of parameters. Finding a solution for these models already is an np-hard problem for sharp levels of information. Finding a solution for a stochastic problem is an extremely time consuming task. Fuzzy parameters might even lead to a state explosion, meaning that an exponentially rising amounts of possible parameter combinations exist. The overwhelming amount of combinations cannot be used to create a valid solution. This situation gets even more complicated, as we use a multiperiod, multilevel production structure. A problem in early periods or on a low level can lead to even more problems in later periods or levels. In many situations, one cannot find a solution that is applicable for all possible situations. Therefore, one cannot assume that that it is practical to include uncertainties in the planning process using stochastic optimization methods. Even when such a solution exists, it is unlikely that it can be found within a reasonable amount of time.

Most stochastic optimization approaches are based on three different methods. Multilevel stochastic models with compensation are based upon Dantzig [6]. Decisions on one level are made at an early point of time and fixed for all following levels. We consider a huge amount of possible events; therefore, we would have to model a corresponding amount of model levels. Stochastic programs with probabilistic constraints date back to Charles and Cooper [4]. Within these models, the breach of constraints is permitted for certain parameter combinations. One can only find proper solutions for this type of models when it is possible to transform the models into an equal deterministic model. Additionally, the expressive value of the model can be reduced due to the loosened constraints. Bellman [3] introduced stochastic dynamic programming. Based upon a decision tree a backward chaining is used to conclude the ideal choice at the decision situation. All this approaches share the issue that they can only be solved efficiently, if the amount of possible scenarios can be reduced to a certain amount. However, when looking at a real production problem many decisions are possible. Therefor we have to find different methods to include uncertainties within the production planning process.

V. INTERFACES TO THE OPTIMIZATION SOFTWARE

To be able to simulate the results of an optimization, the solution data has to be preprocessed in order to prepare the data for the simulation model. CPLEX can export a XML-based file-format, which contains the mathematical programming solution for all variables of the problem. The Converter module reads the file line by line, whereas each line represents a variable. We mainly need two decision variables to be able to simulate the plan: The production variable q_{ktj} determines the products that are produced on a certain machine in a given time period. Additional data like production- and setup times as well as costs can be read from the database based on this production lots. Because a work shift model has been included in the mathematical optimization, every machine can have a different capacity in each period. Therefore, we also have to take the variable b_{jt} into account, which describes these capacities. As we included lead times within the MLCLSP, all needed intermediate products should be available at the beginning of a new period. This means that there are no special requirements for the machine scheduling. We are able to schedule the lots in the same order as they appear within the exported XML file. The real scheduling and date safeguarding will be done within the simulation process. Based upon the given data we are able to calculate all needed information in a deterministic fashion. For example, we are able to calculate the stock or backlog amounts via a difference of production amounts, demands and secondary demands. Thus, we have all information needed to control the simulation procedure. These calculations are also needed to evaluate the simulation results. Therefore, it is a sensible approach to calculate these values for the original plan instead of importing every information from the mathematical model.

C. Simulation

The simulation model is implemented using the discrete event simulator d³FACT developed by our workgroup, Business Computing, esp. CIM. The extensible Java API provides a high-performance, petri-net-based material flow component [1].

The production plan information is first transferred towards the simulation logic. During the initialization of the simulation model, all machines are loading their fixed schedules for the complete planning period. It holds for each machine, which products in what amount have to be produced in each period. Furthermore, it holds the planned durations for the maintenance, production and setup processes. The lot release order is fixed and stays so, even in the case of blocked lots due, to late secondary demands. All released lots are stored in a FIFO-Queue, to be processed in their incoming order. At the beginning of each new period, all planned lots are enqueued and the production cycle starts. Prior to nearly any lot, a setup is intended for rigging the machine. If planned, a routine maintenance of the unit is performed after a given amount of work pieces.

If multiple products or machines demand the same intermediate product, a Fork is needed to control the material flow. It stores and routes the tokens as needed towards the point of consumption. The built-in buffer stores the tokens until a machine starts a job and signals its demand. The fork uses a FIFO-Queue to handle the incoming requests and to minimize the mean waiting time for supply. The machine uses a strict FIFO-Queue for lots to dispatch. In this naïve version, even a blocked lot with unfulfilled secondary demands waits until its demands are met. If all lots for a period are finished, the shift ends and the next jobs are dispatched in the next period.

Under certain circumstances, it is possible that in case of unmet secondary demands and fully loaded periods, lots are pushed into the following period. In this case, the moved lots are scheduled prior to the regular lots to dispatch the longer waiting jobs first. Because the planning methods calculates with one day lead-time it is easily possible that delayed lots are blocking further following demands.

D. Uncertainties in the production planning process

The production schedule execution is typically affected by unforeseen interruptions und disturbances. In the simulation model, maintenance, cycle, and setup times are considered stochastically influenced, due to their high influence on the overall flow shop production process and their deterministic usage in the production-planning model. Material shortages, which arise from supplier unreliableness, are not taken in account and all materials are assumed of as supplied in time.

The maintenance, cycle and setup times that are incorporated in the formulation of the production-planning problem, are forming the lower bound for the process execution and are modeled in the simulation.

The stochastic influences are modeled with two parameters. On the one, hand the likeliness of an increased process time and on the other hand the amount of the deviation. The probability that the planned process time varies, is modeled with a uniform distribution, whereas for the duration a normal distribution is used. Ideally, one is able to use historical data to determine the probabilities for each machine individually; however, this is not possible in a hypothetical model.

E. Rule-based machine control

To be able to improve the production plan within the simulation we are using a rule-based machine control. We are allowing a machine to change its own scheduling plan. As a day of lead-time is included in our planning process, this should not have a negative effect on later production levels. One possible rule that we also implemented appears, when a machine is unable to produce a lot because the secondary demands cannot be met. In this case, the machine logic tries to find other lots for this period, which do not need the missing intermediate products. When such a lot exists, it is processed first while the original planned lot will be processed later. This way, we are able to ensure an even utilization of the given machine capacities. Additionally we reduce the danger of possible backlog amounts. This way we increase the validity robustness of the production plan. Another possible decision rule concerns setup carryovers. If production lots of the same product exist in successive periods, it is sensible to change the scheduling in a way, which allows this product to be produced in the end of the first period and in the beginning of the second period. Therefore, the need to setup the machines for both production lots is not applicable anymore. If one introduces a setup, carryover into the mathematical optimization highly increased solution times may occur. The discussed rule-based mechanisms however only lead towards a small increase in processing time within the simulation process. Additional rules can always be applied in a model specific fashion.

F. Evaluation

The evaluation calculates performance figures for the validity and result robustness. For measuring the validity robustness, we compare the objective value of the simulated plans with the objective value of the original plan from the mathematical optimization. A comparison of single cost values is also possible, like evaluating the influence of capital commitment costs. A plan is considered validity robust, when it does not violate any of the optimization models restrictions. The model we use does allow backlogging however. Backlogging always incurs penalty costs, which also influence the result robustness. However, one cannot assess the influence of delivery dates that could not be met, as it might lead to the loss of a customer in the extreme cases. Therefore, it is sensible to protocol every appearance of backlog amounts.

Important information considers the machine load factors. It can happen that the planned or even the maximum capacity of a machine is not sufficient to produce all lots allocated to it. These events are protocolled and evaluated separately as well. This allows for the search of admissible alternatives.

G. Post-Processing

Within the post-processing component, we are able to use additional simulation external methods to generate an improved production plan with an increased robustness based upon the simulated production plan. An increase of validity however usually creates increased costs. Therefore, we cannot assume that increased validity robustness also correlates with high result robustness.

The simplest way to increase the robustness of a plan is to extend the given capacities where possible. Our model is based on a possible three-shift production. Generally, one tries to avoid using all three shifts to avoid high personal costs during nighttime. By courtesy of the simulation we can however estimate the increase of robustness when considering the introduction of additional shifts. This allows the production planner to decide whether the additional costs are justified or not. One possible way to do this automatically is to calculate the average of the production timings after a higher number of simulations. Afterwards we can determine the average machine load factor and decide upon the amount of needed work shifts.

Another possible way to increase the robustness of the production plan is to move several lots into an earlier period, when this period contains larger capacity reserves. This process is considerably more complicated, as secondary demands also have to be fulfilled in due time. Therefor one cannot simply review available capacities for the final product. One also has to check whether available capacities for the production of all needed intermediate products exist, which often is not the case when the overall machine load factor is constantly high. Additionally an earlier production causes further inventory and capital commitment costs. Thus, this way often is not an opportune choice. In general, it lies in the responsibility of the production planner to decide which amounts of cost increases he accepts to increase the validity robustness of his production plans. All production plans that are created within the post-processing can be simulated and evaluated again. The production planer consequently can access all information he needs to come to a corresponding decision.

VI. RESULTS

We executed several simulation runs based upon the production plan created by the mathematical optimization, using a planning horizon of 56 periods with a dynamic demand structure. We assumed a failure rate of 10% for each machine. The corresponding processes were prolonged by a standard deviation of 15% and 30%. Table 1 shows several performance indicators in a comparison of simulations with a naïve and rule-based machine control, in particular focusing delays for final products... We calculated the average values of 100 simulation runs. The rule-based machine controls objective function costs are considerably lower than the costs caused by the naïve machine control. It is noticeable that less final parts get delayed when using the rule-based machine control. Therefore, the ability to supply is increased and lower delay penalty costs occur. These also explain the lower objective cost values.

TABLE 1: COMPARISON BETWEEN DETERMINISTIC (D), RULE-BASED (RB) AND NAIVE (N)MACHINE CONTROL

Standard Deviation	Sim-Type	Objective Function	Delayed Final Products (Absolute)	Delayed Final Products (Relative)	Delay Penalty Costs	Stock Costs
	D	2.769.282,95 €	2.769.282,95 €	2.769.282,95 €	2.769.282,95 €	2.769.282,95 €
15%	RB	3.944.976,12 €	3.944.976,12 €	3.944.976,12 €	3.944.976,12 €	3.944.976,12 €
	N	4.211.949,84 €	4.211.949,84 €	4.211.949,84 €	4.211.949,84 €	4.211.949,84 €
30%	RB	4.355.206,90 €	4.355.206,90 €	4.355.206,90 €	4.355.206,90 €	4.355.206,90 €
	N	4.670.432,31 €	4.670.432,31 €	4.670.432,31 €	4.670.432,31 €	4.670.432,31 €

However, a deterministic simulation of the production plan without stochastic influences shows that no penalty costs occur. The deterministic objective function value is correspondingly low. The rule-based machine control causes an improvement in result robustness as well as validity robustness. Table 2 shows the corresponding evaluation metrics by Honkomp et al. [10].

TABLE 2: METRICS BY HONKOMP

Standard Deviation	Sim-Type	$S.D./OF_{DB}$	$\overline{OF}/OF_{DB}$
15%	Rule-Based	40,00%	1,42
	Naive	40,09%	1,52
30%	Rule-Based	42,90%	1,57
	Naive	44,40%	1,69

The first column represents the relations between the standard deviation of all objective function values of all stochastic simulation runs and the objective function value of the deterministic simulation. A lower value indicates that disturbances and environmental influences have less impact on the ability to supply. The second column represents the relations between the objective function values of stochastic and deterministic simulations. The value shows the cost increase caused by the disturbances and directly shows the result robustness. Normally, a higher robustness is gained by increased costs. However, the inclusion of penalty costs into the objective function value causes lower cost for the more robust plan.

Another reason for increased costs are the personal costs. The simulation showed that more working shifts have to be introduced to be able to satisfy customer demands. The original plan was using working shift per day. The resulting plans when using either simulation method mostly used two or three shifts. The rule-based machine control delays 13% less products beyond the planned capacity restrictions, therefore needing less working shifts and causing less personal costs as well. When analyzing the problems within the production process one needs to find out where a possible bottleneck occurs. During the simulation we protocol all occurrences of backlog amounts and the connected machines, products and periods. For further analysis we can determine which products are delayed most as shown in figure 3.

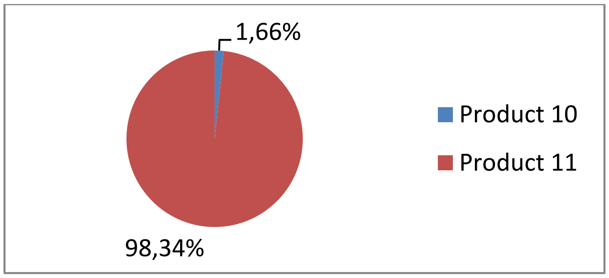


Figure 3: Delayed Final Parts according to products

Surprisingly, most delays are caused by one final part. This is an obvious sign that the production capacity for this part might not be sufficient. Alternatively, production capacities for needed intermediate products might be insufficient. This can be found out by analyzing internal delays for the intermediate products. Table 3 shows the absolute and relative internal delays for both simulation types averaged over 100 simulations. We define internal delays as the amount of intermediate products that couldn't be produced in the planned period.

The usage of the rule-based machine control also shows an improvement when considering the internal demands. Despite not leading to direct revenue losses due to unmet demands, internal delays can cause costs when changes in the production plan have to be made. These costs aren't implicitly included into our production model, but it is in the interest of the production planner to reduce these costs as well. When considering the internal delays per product we are able to find out that product 10 and product 11 are based on the same intermediate product. This product possesses several internal delays, which influence the production of the final products. We were able to find the bottleneck in our production model and can take action to reduce the impact of this issue.

TABLE 3: ANALYSIS: ACCUMULATION OF INTERNAL DELAYS

Standard Deviation	**Sim-Type**	**Internal Delays (Absolute)**	**Internal Delays (Relative)**
15%	Rule-Based	10194,38	1,81%
	Naive	11172,92	1,99%
30%	Rule-Based	16172,68	2,88%
	Naive	17266,10	3,07%

VII. CONCLUSIONS

We have shown in this paper that a material flow simulation can be used to analyze a production plan created in a mathematical optimization and to evaluate its robustness. It is easily possible to read the results of an optimization process, to transfer this data into our simulation framework. We are able to simulate the plan including probabilities for unforeseen events and fuzzy information. The results of the simulations can be used to find possible weak spots in the given plan. In several cases, we might be able to fix these weak spots through automatic post-processing or with manual changes. The effect of these changes can also be evaluated using additional simulation runs. Therefore, a production planner can decide whether he wants to implement these changes or not. Performing a large number of simulations is substantially faster than running another instance of the optimization problem. In the end, we recommend this approach for practical and economic usage.

REFERENCES

[1] Almeder, C.; Preusser, M.; Hartl, Richard F.: Simulation and optimization of supply chains: alternative or complementary approaches? In OR Spectrum 31, S.95–119, Springer Verlag, 2009

[2] Aufenanger, M., Dangelmaier, W., Laroque, C., Rügener, N.: "Knowledge-based Event Control For Flow-Shops Using Simulation and Rules". In Proceedings of the 2008 Winter Simulation Conference, Edited by S. J. Mason, R. R. Hill, O. Rose, T. Jefferson, J. W. Fowler, p 1952-1958, 2008

[3] Aytug, H., Lawley,M.A., McKay, K., Mohan, S. Uzsoy, R.: "Executing production schedules in the face of uncertainties: A review and some future directions." European Journal of Operational Research 161: p68-110, 2005.

[4] Bellman R.: "Dynamic Programming", Princeton University Press,Princeton, New Jersey, 1957

[5] Bihlmaier, R.; Koberstein, A.; · Obst, R.: Modeling and optimizing of strategic and tactical production planning in the automotive industry under uncertainty In OR Spectrum 31, S. 311-336, Springer Verlag, 2009

[6] Biswas S, Narahari Y (2004) Object oriented modeling and decision support for supply chains. Eur J Oper, Res 153:704–726

[7] Charnes, A.; Cooper, W.W.: Chance-constrained programming: Management Science 5:73-79, 1959.

[8] Chong, C. S., Sivakumar, A. I., Gay, R.: "Simulation-based scheduling for dynamic discrete manufacturing." In Proceedings of the 2003 Winter Simulation Conference, Edited by S. Chick, P.J. Sánchez, D. Ferrin, and D. J. Morrice, 2003

[9] Dantzig, G.B.: "Linear Programming under uncertainty", Management Science 1:197-206, 1955.

[10] Frantzén, M., Ng, A. H. C. , Moore, P.: "Asimulation-based scheduling system for real-time optimization and decision making support". Robotics and Computer-Integrated Manufacturing, 2011

[11] Ghezail, F., Pierreval, F., Hajri-Gabouj, S.: "Analysis of robustness in proactive scheduling: A graphical approach". In Computers & Industrial Engineering 58 (2010) 193-198, 2009

[12] Helber, S., Tempelmeier, H.: "A heuristic for dynamic multi-item multi-level capacitated lotsizing for general product structures" in European Journal of Operational Research 75, 296-311, North-Holland,1994

[13] Honkomp, S., Mockus, J.L., Reklaitis, G.V.: „A framework for schedule evaluation with processing uncertainty". Computers and Chemical Engineering 23:595-609., 1999

[14] Fleischmann, B.: "The discrete lot-sizing and scheduling problem" in European Journal of Operational Research 44, 337-348, North-Holland, 1990

[15] Maes, J.; Van Wassenhove, L.: Multi-Item Single-Level Capacitated Dynamic Lot-Sizing Heuristics-A General Review: The Journal of the Operational Research Society, Vol. 39, No. 11 (1988), S.991-1004

[16] Peiffer, A., Kádár,., Monostori,L., Karnok, D.: „Simulation as one of the core technologies for digital enterprises: assessment of hybrid rescheduling methods", International Journal of Computer Integrated Manufacturing, 21:2, 206 – 214, 2007

[17] Peiffer, A., Kádár,., Monostori,L.: „Stability-oriented evaluation of rescheduling strategies, by using simulation." In Computers in Industrie 58 (2007). 630-643, 2007

[18] Rasconi, R., Cesta, A., Policella, N..: "Validating scheduling approaches against executional uncertainty". In Journal of Intelligent Manufacturing (2010). 49-64, 2008

[19] Schneeweiß. C.: "Planung 2: Konzepte der Prozeß- und Modellgestaltung", Springer, 1992

[20] Scholl, A.,:„Robuste Planung und Optimierung", Physica-Verlags, 2004

[21] ILOG CPLEX :High-Performance Software for Mathematical Programming and Optimization: http://www.ilog.com/products/cplex

Fault-tolerant Distributed Discrete Event Simulator Based on a P2P Architecture

Jorge Luis Ramírez Ortiz
Dept. of Electrical Engineering
Autonomous Metropolitan University
Iztapalapa, México
Email: jramirezort@gmail.com

Ricardo Marcelín Jiménez
Dept. of Electrical Engineering
Autonomous Metropolitan University
Iztapalapa, México
Email: calu@xanum.uam.mx

***Abstract*—We describe the construction and performance evaluation of a new distributed discrete events simulation (DDES) tool, based on the Peer-to-Peer (P2P) paradigm. This approach allows the utilization of redundant resources in order to withstand failures on the very processing entities in charge of the simulation work. Our results show that the mechanisms supporting dependability are expensive and are only recommended for long-lasting and very processing-demanding simulations.**

***Keywords*-DDES; P2P; redundancy; snapshot; fault-tolerance.**

I. Introduction

Distributed Discrete Events Simulation (DDES) has found applications to study those systems made up from a massive number of components interacting over a time scale, where it is also necessary to reproduce their complex behavior under controlled conditions and with very fine granularity, i.e., with a high degree of realism. This is the case, for instance, of modern telecommunications systems, very large scale of integration (VLSI) circuits or even some biological models.

The basic operation of DDES consists in dividing the model that represents the system under study, in such a way that each of the resulting parts is simulated using a different computer. This also implies that the supporting computers should be connected by means of a communications network, in order to simulate the possible interactions among the parts of the model.

It is considered that the fundamentals of DDES were settled by Misra [1] with his seminal paper from 1986. Nevertheless, the major breakthroughs on the subject were achieved with the development of high speed networks over the last fifteen years about. Despite of the fact that DDES has evolved to be part of the ordinary toolbox of many research teams, there are open issues on the subject representing a challenging area of opportunity. This is the case of dependability. Suppose that in the middle of a long-lasting simulation, a given computer crashes. The ongoing simulation should be restarted from zero unless a fault-tolerant mechanism is implemented.

In the meantime, new paradigms have been developed in the fields of parallel and distributed computing. These new approaches foster the cooperative work among the components of the very system, in order to tackle complex problems. P2P systems, for instance, have found applications in situations where it is required to split up a big task to render smaller problems that can be assigned to a given number of appointed peers. Projects like seti@home [22], or einstein@home [23], and more recently folding@home [21] or rosetta@home [24], are representatives of this new trend.

In this work, we introduce the construction and performance assessment of a new prototype DDES tool based on the P2P paradigm. Our proposal supports crash failures as well as peer departures. If necessary, a missing component can be replaced by a spare peer and the ongoing simulation can be restored to a previously recorded global state, instead of starting over again. Our prototype is built using JXSE [20], the Java platform for P2P applications development.

The rest of this paper includes the following parts: Section II introduces the basic concepts of discrete event simulation, Section III gives a general view about related work, Section IV describes the operations of our prototype, Section V presents the results of the tool's performance evaluation, and Section VI shows our conclusion and future work.

II. Basic Concepts

A discrete event simulation can be understood as a collection of logical processes. The interaction between any couple of processes is modelled by the exchange of time-stamped messages. This exchange is said to follow the restriction of local causality, if and only if each logical process dispatches all its schedulled events, including the messages that receives, according to their timestamps.

Notice that a logical process should stop any activity until it receives a message from each of the processes that interact with it, to be sure that it chooses the message with the smallest timestamp. Nevertheless, this approach poses the risk of creating a deadlock among a set of entities interacting in a circular way.

There exist two types of methods that deal with the risk of blocking in distributed simulations. On one side, we found the *optimistic procedure*. In the other side, we have the *conservative procedure*. In the first case, messages can be exchanged as they are produced. This decision may lead to a potential violation on the causal order of events, which is detected when a given entity receives a straggler message and it finds out that its corresponding timestamp is smaller than the timestamps of a number of messages already processed. This means that the straggler should have been dispatched before any of them. To fix this condition, the receiver starts a local procedure called *rollback*, that restores its local state to a previous one where the new set of messages can be processed accordingly. If necessary, the entity in charge of rollback must send the corresponding *anti-message(s)* to cancel the effect of any message that could have been issued out of order. In the other side, the conservative procedure avoids the possibility of executing actions out of chronological order. This time, entities exchange the so-called *null-messages*, which do not carry any physical meaning. The transmitter of a null-message sets a lower bound on the timestamp for the next meaningful message that could be issued. Each entity knowing the least timestamp that can receive from any of its incoming channels, will be able to dispatch any message in the right order.

The use of an optimistic synchronization mechanism also implies that the involved participants should take the responsibility of saving their local states to support rollback. Therefore this solution implies the utilization of bounded storage capacities. To prevent storage overflow, it is necessary to implement a *fossil recollection mechanism*. This is, a mechanism to dismiss previous recorded states that can not be reached, by no means, during rollback. It is known that no rollback can restore the state of any processing unit beyond the so-called *global virtual time* (GVT) [1][11][14][15].

Let p_i be a process that takes part in a distributed execution, such that σ_i^0 is the initial state of p_i, and let σ_i^k be the state of the process right after executing its k-th local event e_i^k. The *global state* of the distributed execution will be the n-tuple $\Sigma = (\sigma_1 \ldots \sigma_n)$, made up with the local states of the participating processes. We now define a *cut* $C = h_1^{c_1} \cup \ldots \cup h_n^{c_n}$, where $h_i^{c_k}$ is the ordered set of events dispatched at process i, up to its last local event c_k. Indeed, the cut can also be described in terms of the tuple $(c_1 \ldots c_n)$. The set $(e_1^{c_1} \ldots e_n^{c_n})$, including the last event on each process, is called the border of the cut. Evidently, each cut $(c_1 \ldots c_n)$ defines a global state $(\sigma_1^{c_1} \ldots \sigma_n^{c_n})$. Based on the “happened before” relation $(\rightarrow)$, it is possible to define a consistent cut if, for any to events e y e':

$$(e \in C) \wedge (e' \rightarrow e) \Rightarrow e' \in C. \qquad (1)$$

Otherwise, we said that we deal with an inconsistent cut. Therefore, a *consistent global state* will be a global state corresponding to a consistent cut. There exist a well-known collection of distributed mechanisms that can be employed to produce the global state of an ongoing distributed execution [18][10].

III. Related work

In this section, we present a collection of discrete events simulators, whose functionalities address many of the features of our proposal. By no means we intend to present an exhaustive description, but a sample of the systems that we consider to be closely related to ours.

The list of tools that we decided to consider includes the following simulators: Parsimony [2], GPDES [3], Omnet++ [4], POSE [6], μsik [7], and Aurora [8]. We also present Table I, where we describe the features that led our search. These are general purpose tools. The label “C-C” stands from “Client-to-Client”, which means that the tool supports the communications between any processing unit. In contrast, “C-S” means that each client, or processing unit, is only aware of the existence of the server from which receives workload and to which sends the results of its local processing. The rest of the columns are self-explanatory.

It is apparent that only Aurora supports either fault-tolerance, or changes on the underlying communications network. Nevertheless, it is also important to point out that this system can not be considered a real DDES tool, as it does not partition the instance of the model under study to allocate each of the resulting parts to a different processing unit. Instead, it allocates a completely different instance of the model to each of the available processing components, pretty much in the spirit of systems like seti@home, folding@home, among others. In contrast, distributed DES techniques take for granted that the processing units are required to exchange information among themselves in order to produce the trace that represents the behavior of the system under study.

Fault-tolerance is a pending issue do not addressed on any of the tools of our list. The difficulties of building a DDES supporting fault-tolerance lie on the fact that it is required a thorough design including three types of redundancy: i) space or component redundancy offering spare units to replace any possible active unit that may go out of service, ii) information redundancy to regularly record a snapshot or global state of the ongoing distributed execution, and iii) time redundancy to repeat a given distributed execution from a previously recorded snapshot, when a faulty component was still active (before the last failure ocurred). For this purpose the missing unit is previously replaced by a spare unit, which now starts from the corresponding local state of the unit that replaces. Previous works [9][19] have recognized that the toughest problem comes from the construction of a global state of the underlying asynchronous execution, specially when it can not be granted the existence of FIFO channels.

IV. Architecture

The design of our proposal is based on two types of entities, in charge of a simulation: the coordinator, implemented by a rendezvous type peer, and the workers, implemented by full-featured edge peers [13]. A worker contacts the coordinator to offer its processing capacities. From the coordinator’s perspective these cooperative peers are regarded to be either as idle, or active workers (see Figure 1).

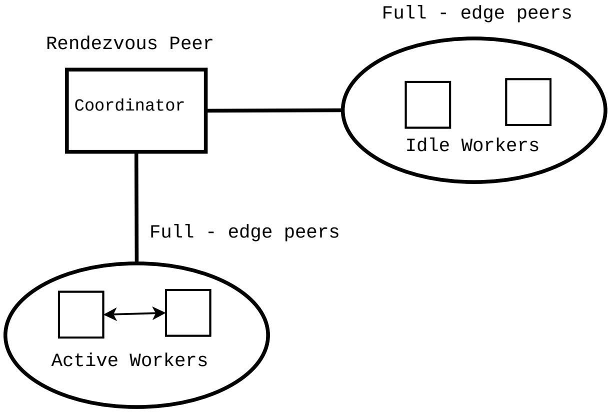


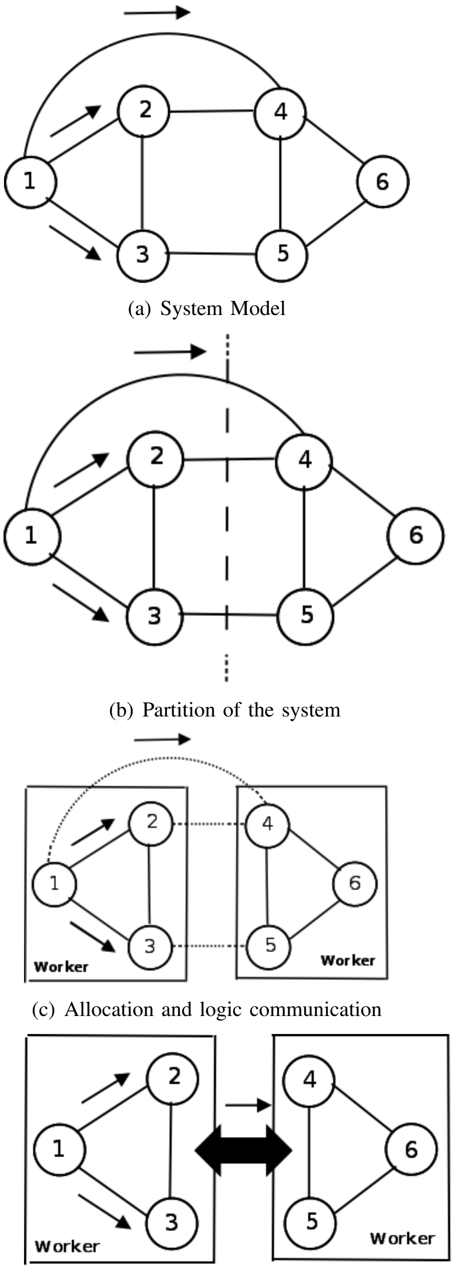
Figure 1. Architecture of our P2P-based simulator.

At the starting stage, the coordinator considers that any available worker is idle. The coordinator knows the graph representing the system about to be simulated (see Figure 2(a)). Then, it splits up (partitions) the graph into a fixed number o subgraphs (see Figure 2(b)) and allocates each of the resulting parts to an idle worker, which now is consider to be active (see Figure 2(c)) and (see Figure 2(d)).

Table I
SOME GENERAL PURPOSE DDES

Project	Comms.	Synch.	Network	Fault-tolerance
Parsimony	C-C	both	static	N
GPDES	C-S	—	static	N
OMNeT++	C-C	both	static	N
POSE	C-C	optimistic	static	N
μsik	C-C	both	static	N
Aurora	C-S	—	dynamic	Y

The coordinator triggers the simulation sending a message to the active worker(s) in charge of the node(s) which is (are) supposed to receive the starting event(s). On each active worker there is a single execution thread that processes all possible messages sent to the given peer.



(a) System Model

(b) Partition of the system

(c) Allocation and logic communication

(d) Real communication

Figure 2. System's graph partition and allocation.

Our simulation tool supports two different restoration mechanisms, the so-called local rollback or restoration of type 1, which is trigered when some straggler message is received out of order and threatens the causal delivery of events. The restoration of type 2, or global rollback, happens when an active worker leaves the system or crashes, and the whole system must be restarted from a previously recorded global state. In both cases the evolution of any logical process is recorded by 3 complementary queues: *the input message queue, the output message queue, and the former states queue*.

In the case of local rollback, only the logical processes directly involved are restored to a previous state in order to guarantee the causal delivery of the late message. If necesary, this procedure may imply the transmission of some anti-messages that start a similar procedure at the receiving peers. Also, each peer is required to store by itself the states of the local processes that allocates. In contrast, global rollback happens when the coordinator detects that an active peer is missing. In this condition, it stops the overall ejecution, then it enables an idle peer to replace the missing one. Finally, the coordinator "rewinds" the whole system to a previously recorded snapshot. This also means that the coordinator is responsible for storing and updating the snapshots that regularly takes. By updating we mean that when the last state has been recorded, the coordinator eliminates the previous one, in order to maintain a limited amount of storage to support this mechanism.

It is apparent that the coordinator is also responsible for recording the global state or snapshot (SN) of the ongoing distributed execution. To accomplish this task, *it regularly stops the overall execution and sends a snapshot message to an appointed worker which starts the Chandy-Lamport protocol among the active workers*. When each worker recognizes that it has finished the protocol, it sends its local records to the coordinator which collects these results to put the pieces together and store the resulting snapshot. This procedure is the key to support the global rollback mechanism.

Similarly, the coordinator is also responsible for triggering the distributed procedure that determines the GVT. When each worker has finished its local procedure, it sends its local proposal to the coordinator, which collects this value from every worker and picks up the smallest result, now considered the new GVT. Finally, the coordinator broadcast this new value to every active peer. The GVT enables the active workers to proceed with their fossil recollection process which, in turn, complements the local restauration process. SN and GVT messages have the first and second highest priority above any possible message that can be received at any worker.

Last, the coordinator collects the traces generated by the active workers, containing the partial results of the simulation. It is able to recognize when the GVT reaches a predefined value $+inf$, which implies that the running simulation is finished.

V. Performance assessments

We devised a comprehensive set of experiments to evaluate our tool's performance under different conditions. We also considered that the selection of the type of system to be simulated was a key aspect for this experimental assessment. We looked for a simple system with a deterministic behavior, where the complexity of its operations comes from the size of the system and the connection among its elements. We also looked for a type of system ranging from very small to massive instances.

Each instance of the system to be studied is represented by a connected graph, called the communications graph. The simulation to be run on this graph consists in the execution of the PIF algorithm (which stands from Propagation of Information with Feedback)[17]. An appointed node in the communications graph, from now on called the root, starts sending an information unit to each of the nodes that share an edge with it, i.e., to its neighbors. A node is said to be "woken" (see Figure 2(a)), when it receives this information unit for the first time, it also considers that its "father" is the node that woke it. On due time, each of the receiving nodes forwards the same unit to its corresponding neighbors but to its father. Only when the node has received this unit from each incoming link, it is able to send this information back to its father. The global effect of this simple algorithm is observed in two consecutive phases. During phase one (propagation or broadcast) we can imagine an expanding wave that goes from the root to the rest of the graph. When this wave reaches the boundaries of the graph it starts the second phase (convergecast). This time, information travels back to the root on top of the three induced on the graph during phase one.

We selected this simple algorithm to simulate, for we know in advance the place where action starts and finishes. Indeed, we fix this point. Also, we know the total number of information units to be exchanged. Besides, the simulation time depends only on the underlying graph diameter.

We stressed our tool with three different types of experiments:

a. In the first group of experiments, we tested a fixed set of systems under 3 different conditions, using a centralized simulation, using a distributed simulation with 2 active workers in charge and finally, a distributed simulation with 3 active workers. We compared the time required to finish the simulation under each of these cases.

b. In the second group of experiments, we measured the global added cost required to support a fault-tolerant system. We considered that this feature is mainly based on the capability of recording snapshots and the evaluation of the GVT. Therefore, we ran new system instances, but this time with the snapshot and GVT procedures switched on and off, respectively. Notice that, we did not inject faults, but we evaluated the price of the "insurance" mechanisms, although these mechanisms were never invoked.

c. In the third group of experiments, we wanted to evaluate the tool's resilience. In order to trigger the global restoration procedure, we dismissed an active worker immediately after the first snapshot was already stored at the coordinator. We used the same system instances from the previous study. We compared the time required to finish the simulation under each of the cases, with and without failure.

About the graph representing the model of the system under study, we tested 10 different graph orders, from 100 up to 1000 nodes. For each order we created 10 different graph instances. Each instance is a graph randomly created.

A. First group of experiments: centralized vs distributed

In this experimental design we simulated the PIF algorithm, with two types of configurations: centralized (1 active worker) and distributed (2 and 3 active workers). Communication channels are assumed to have constant delay. Distributed simulations include snapshot and GVT mechanisms.

Results show that (Figure 3) the graph order is a key element to decide the best configuration that supports the fastest simulation, i.e., a distributed simulation is not necessarily the fastest, specially when we deal with small graph orders.

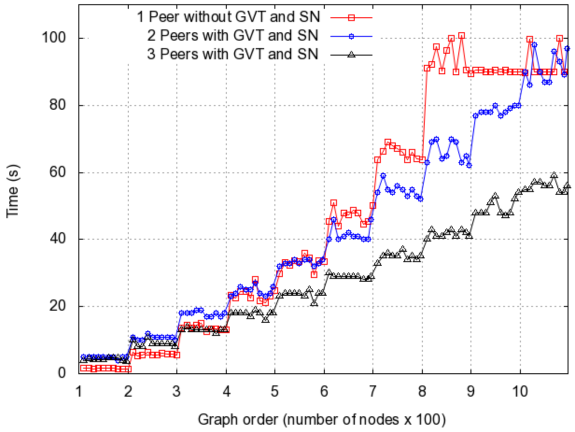
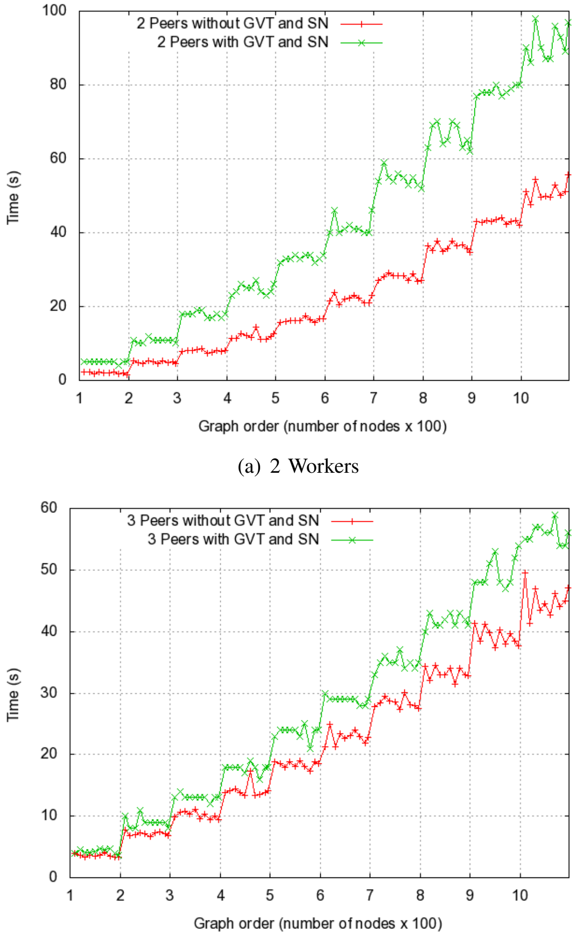


Figure 3. Centralized simulation vs distributed simulation.

B. Second group of experiments: the added cost of fault-tolerance mechanisms

In this new set of experiments we evaluated the involved cost, measured in simulation time, required to deploy the fault-tolerance mechanisms. We consider that this capability basically depends on the utilization of the snapshot and GVT procedures. For this purpose we tested 2 distributed configurations, with 2 and 3 active workers, respectively. On each case, we measured the elapsed time required to finish a given simulation, with and without fault-tolerance mechanisms included. It is very important to quote that not any worker was dismiss, but we simply switched on and off these mechanisms to measure how much the processing time is lengthened, when these "insurance policies" are included. Also, it is important to notice that, for those cases where mechanisms are included, they are only executed once, during the overall elapsed time.

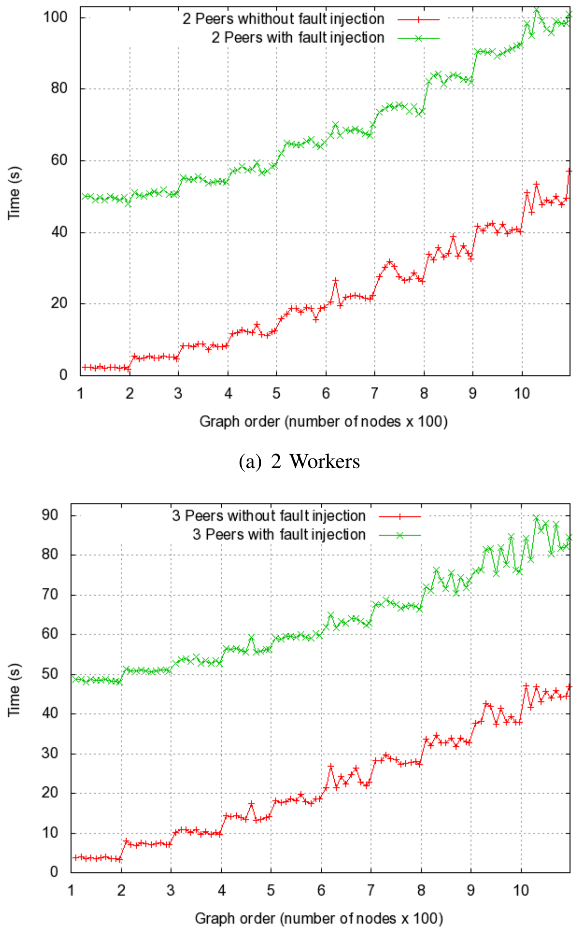
Results show that (see Figures 4(a) y 4(b)) the extra resources, required to support fault-tolerance, depend on the number of involved peers that share these mechanisms. In these particular cases, 3 peers apparently have a smaller impact on



(a) 2 Workers

(b) 3 Workers

Figure 4. Simulation with/without GVT and SN.



(a) 2 Workers

(b) 3 Workers

Figure 5. The effect of fault injection

the simulation added cost, compared to the impact observed on the configuration made up with 2 peers only.

C. Third group of experiments: fault injection

Finally, in this set of experiments we wanted to know whether it is worth using fault-tolerance mechanisms or it is better to start over a simulation from the very beginnig. In order to compare these two possibilities, we assume that an active worker may go out of service an instant before the finalization of the underlying simulation. We tested 2 distributed configurations with 2 and 3 active workers. In both cases we compared the elapsed simulation time with and without fault injection. A fault is produced by dismissing an active worker immediately after the snapshot and the GVT mechanisms have been executed. Results show that (see figs. 5(a) y 5(b)) the graph order is the key to answer this question.

VI. Conclusion and Future Work

In this work, we described the construction of a prototype that demonstrates the viability of a DDES tool based on P2P entities. The design of our proposal is based on two types of peers: the coordinator, implemented by a rendezvous type peer, and the workers, implemented by full-featured edge peers. This solution supports workers failures, as well as departures. We focused our work on measuring the costs associated to dependability. Fault-tolerance mechanisms are expensive and it is worth their utilization provided that we deal with a long lasting simulation. New open issues are pending, such as the optimal snapshot recording frequency, the efficient storage of the global state, the possibility of load rebalancing on the fly and the possibility of supporting a failure at the very coordinator. In contrast to the “@home” type applications, DDES requires a strong interaction between participants. Therefore, we consider that this new approach will turn into a feasible solution only when final users, behind the altruistic peers, will be connected by high-speed channels.

References

[1] J. Misra, “Distributed discrete event simulation,” Computing Surveys, vol. 18 Issue 1, pp. 39-65, Mar. 1986.

[2] B. Preiss, “The parsimony project: a distributed simulation Testbed in Java,” Proc. 1999 International Conference On Web-Based Modelling and Simulation, vol.31 of Simulation Series, pp. 89-94, San Francisco, CA, January 1999. Society for Computer Simulation.

[3] L.M. Campos. “A general-purpose discrete event simulator,” Symp. on Performance Evaluation of Computer and Telecomunication Systems, Orlando, USA, Jul. 2001, pp. 1-12.

[4] OMNET++ User Manual. Internet: http://www.omnetpp.org/doc/manual/usman.html#sec378, 28.07.2011.

[5] B. Preiss and I. MacIntyre. "Yaddes yet another distributed discrete event simulator," User Manual. Internet: http://www.brpreiss.com/reports/ccng/E-197/report.pdf, 28.07.2011.

[6] T.L. Wilmarth and L.V. Kale, (2004). "POSE: getting over grainsize in parallel discrete event simulation," Proc. International Conference on Parallel Processing. Los Alamitos, CA: IEEE Computer Society, vol. 1, Aug. 2004, pp. 12-19, doi:10.1109/ICPP.2004.1327899.

[7] K.S. Perumalla, (2005). "μsik a microkernel for parallel distributed simulation systems," Proc. 19th Workshop on Principles of Advanced and Distributed Simulation, Los Alamitos, CA: IEEE Computer Society, Jun. 2005, pp.1-12, doi:10.1109/PADS.2005.1.

[8] A. Park and R.M. Fujimoto, " Aurora: an approach to high throughput parallel simulation," Proc. 20th Workshop on Principles of Advanced and Distributed Simulation, Los Alamitos, CA: IEEE Computer Society, Jun. 2006, pp. 3-10, doi:10.1109/PADS.2006.11.

[9] O. P. Damani and V. K. Garg "Fault-Tolerant Distributed Simulation," Proc. Twelfth Workshop on Parallel and distributed simulation , May. 1998, pp.38-45.

[10] K. M.Chandy and L. Lamport, "Distributed Snapshots: determining global states of distributed systems," ACM Transactions on Computer Systems, Feb. 1985, 3(1), pp. 63-75.

[11] D.R Jefferson, "Virtual time," ACM Transactions on Programming Languages and Systems, Jul. 1985, 7(3), pp. 404-425.

[12] R. Koo and S. Toueg, "Checkpointing and rollback-recovery for distributed systems," IEEE Transactions on Software Engineering, Jan. 1987, pp. 23 - 31.

[13] J. Verstrynge, Practical JXTA cracking the puzzle, Dawning Streams, Inc.,Lulu Enterprises, Inc., Netherlands, 2008.

[14] Y. Lin and P.A. Fishwick, "Asynchronous parallel discrete event simulation," IEEE Trans. on Systs., Man and Cibernetics, Part A: Systems and Humans, Jul. 1996, pp. 397-412.

[15] A. Ferscha, Parallel and Distributed Simulation of Discrete Event Systems, Handbook of Parallel and Distributed Computing, McGraw Hill, 1995.

[16] R.M. Fujimoto, "Parallel and Distributed Simulation," Winter Simulation Conference, P.A. Farrington, H.B. Nembhard, D.T. Sturrock, and G.W. Evans (eds.), Dec. 1999, pp. 122-131.

[17] A. Segall. "Distributed network protocol," IEEE Trans. on Information Theory, Jan. 1983, 29(1): pp. 23-35.

[18] L. Lamport. "Time, clocks and the ordering of events in a distributed system," Communications of the ACM, Jul. 1978, 21(7):pp. 558-564 .

[19] Y. Lin, "Design Issues for Optimistic Distributed Discrete Event Simulation," Journal of information science and engineering, 2000, Vol. 16, pp. 243-269.

[20] JXSE. Internet: http://jxse.kenai.com/, 28.07.2011.

[21] folding@home. Internet: http://folding.stanford.edu/, 23.05.2011.

[22] seti@home. Internet: http://setiathome.berkeley.edu/, 23.05.2011.

[23] einstein@home. Internet: http://einstein.phys.uwm.edu/, 23.05.2011.

[24] rosetta@home. Internet: http://boinc.bakerlab.org/rosetta/, 23.05.2011.

ISBN: 978-1-61208-169-4

Distributed Simulation of Dense Crowds

Sergei Gorlatch, Christoph Hemker, and Dominique Meilaender
University of Muenster, Germany
Email: {gorlatch,hemkerc,d.meil}@uni-muenster.de

***Abstract*—By extending previous approaches, we develop an agent-based model for the simulation of large, dense groups (crowds) of individuals. The model reflects behavioral complexity by assigning psychological and physiological attributes to the agents. In order to cope with the computational complexity, we design and implement a distributed, multi-server simulation framework in which the user can flexibly change both the simulation environment and parameters at runtime. We implement the simulation system using our Real-Time Framework (RTF) and demonstrate its scalability and speedup over multiple servers.**

***Index Terms*—Crowd simulation, Agent-based simulation, Parallel and distributed simulation, Real-Time Framework (RTF).**

I. Introduction and State of the Art

The simulation of the behavior of large and dense human crowds is a socially important and technologically challenging task. To represent the behavior of a crowd, three different kinds of models have been proposed in the literature: flow-based, entity-based, and agent-based models.

This paper develops an agent-based model for computer-based simulations that reproduces the motion of a crowd of individuals by a combination of psychological and geometrical rules with social and physical forces. We also design and implement a new approach to parallelize the simulation across several servers, using the Real-Time Framework (RTF) [2], developed at the University of Muenster.

In developing our model for crowds, we start with the HiDAC (High-Density Autonomous Crowds) model [5] based on the previous work [3] and improving the models suggested in [1], [4]. The resulting agent-based model has no central controlling unit; each agent corresponds to a simulated person with its own individual behavior.

We aim at a model for the challenging case with dense crowds, complex indoor scenarios with many rooms and large, unobstructed areas, with a possibility of panic situations. Our agents are designed to react dynamically to changes in their environment (e.g., if a door is interactively closed during simulation) and can select alternative routes. Agents pursue a global goal, e.g., leaving the building by following a sequence of waypoints at those doors that lead to the exit.

The high density of crowds and the complicated scenarios lead to highly intensive calculations. Therefore, we develop a distributed implementation of the simulation system that can run on multiple servers. We address the critical problem of scalability, which should allow for significantly higher numbers of agents than the sequential version, and we demonstrate the achieved speedup of simulation by conducting experiments with real-world scenarios.

II. The Model for Crowds

We implement several extensions and optimizations to HiDAC [5], in particular: a) the opportunity for dividing big rooms in several smaller rooms in order to balance the computations; b) adaptable creation of rooms, with nearly arbitrary number and arrangement of walls; and c) the introduction of local waypoints which are used for avoiding collisions and for moving around walls standing in the way.

Virtual World Representation. A map of the virtual world, e.g., a building, consists of several rooms which are augmented with walls and obstacles. We construct maps under the condition that walls and doors geometrically form polygons. This quite realistic assumption allows us to apply Jordan's curve theorem and to use the so-called ”Ray casting algorithm” for deciding about the viewfield of the agents.

Collision recognition. Figure 1 shows how an agent recognizes a wall standing in its way (for that, test calculations are performed which we omit here because of lack of space) and runs around the wall. The values used for the corresponding calculations are as follows: the normalized vector n_w , the orientation vector o_w, the startpoint S and enpoint E, the current (global or local) goal T of the agent, the position P and its shortest distance to the wall d_w, and L is a potential local waypoint on one of its ends.

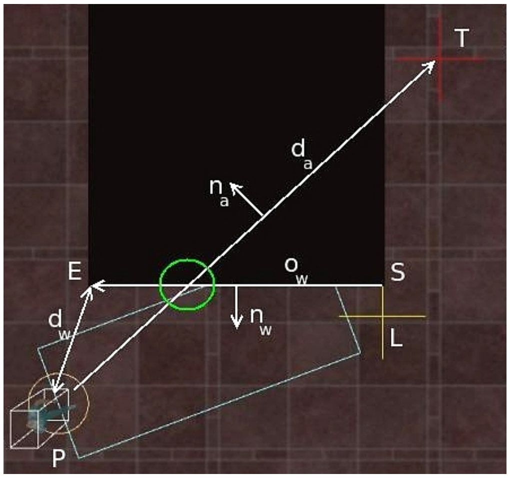


Fig. 1: Recognizing a wall standing in the way

Pathfinding. Agent's movement is simulated by first assigning the agent a start room and a target room. Pathfinding reduces to the problem of finding the shortest path along the

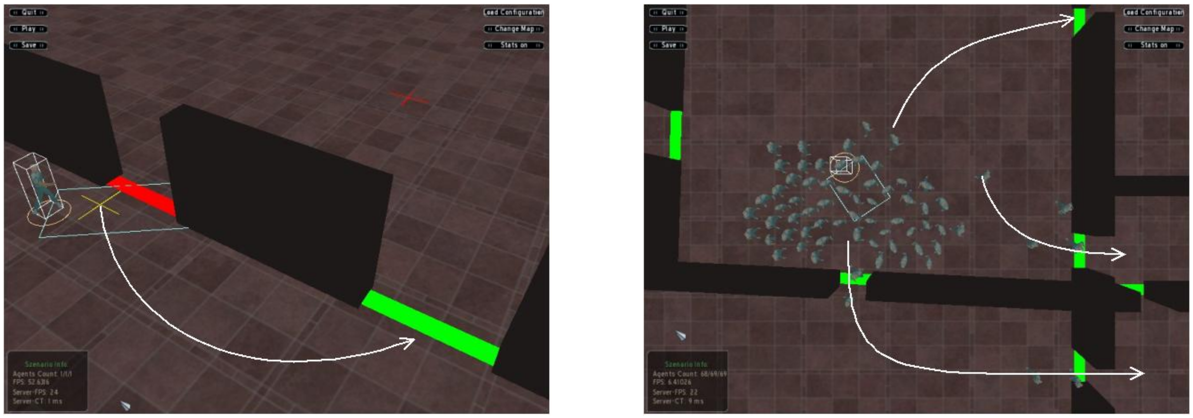


Fig. 2: Alternative routes for closed (left) and open (right) doors

edges of the graph that represents the simulated rooms. The weight of the graph edges plays an important role: the more agents stand before a door, the greater is door's weight: every single agent contributes to it by its diameter. The agent is excluded from the door weighting, as soon as it crosses the door or chooses another door. By using Dijkstra's algorithm for finding the shortest path, we determine the list of rooms which the agent should cross. During simulation, every agent selects the way on which it would bump into as few other agents as possible. Using door weights, the agent learns about the blocked and free passageways beyond its current room.

Alternative routes. An agent decides from one path to another in one of two cases: either a door on agent's way turns out to be closed or this door is blocked by other agents.

Closed doors which are located in the direct viewfield of an agent (see Figure 2, left) trigger a re-calculation of the shortest path; doors which are known to be closed/blocked are removed from this calculation by removing the corresponding edges from the graph of the building. For the doors which are blocked but not closed (Figure 2, right), the patience attribute of the agent is used for deciding either to wait or to aim at another door. The patience attribute is implemented as a counter: our model avoids too long detours by comparing the advantage of an alternative route with its overhead.

Agent's perception. Agents are designed to be similar to people in their perception of the environment. Every agent has its viewfield, which we also call its influence rectangle.

Within its viewfield, the agent tries to avoid collisions, see Figure 3. A collision is interpreted geometrically as an overlapping of an agent with another agent, an obstacle or a wall. The recognition and the consequent handling of the collisions is based on recognizing this overlappings and restoring an overlapping-free situation.

In the model, we express also pushing behaviour which means that an agent is actively pushing onto other agents, in order to reach its goal faster, in spite of possibly bringing the others to falling down. An opposite behavior to pushing is the agent waiting patiently: every agent owns a circular area within its viewfield, so-called circle of influence, such that if another agent appears within this circle then the agent waits till the other leaves. In order to avoid the situation that two agents block each other, each agent has a timeout value, which limits its waiting and which depends on its level of patience. Similarly, we model the effects of hectic running forwards and backwards in very dense scenarios.

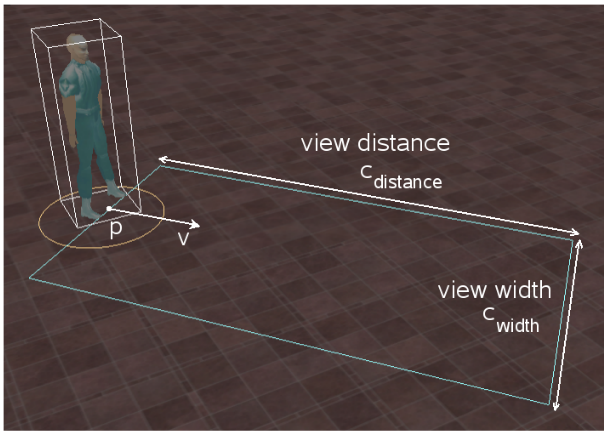


Fig. 3: The viewfield of an agent

Intelligent collision avoidance. In the model, we must avoid the situation that if after the collision of an agent with an object, the priorities and the external influences on the object do not change, then the agent will collide with the same object again and again. Therefore, we implement two additional features: a) intelligent collision avoidance changes the priorities of the agent for a short period of time after the collision; b) modified collision avoidance changes the external influence onto the agent after the collision, such that the same collision becomes very improbable.

Agents' falling down. An agent can be brought to falling down when other agents are pushing it too hard. In order to ensure that the simulation is near to reality, the other agents try to avoid the fallen agents considering them as obstacles;

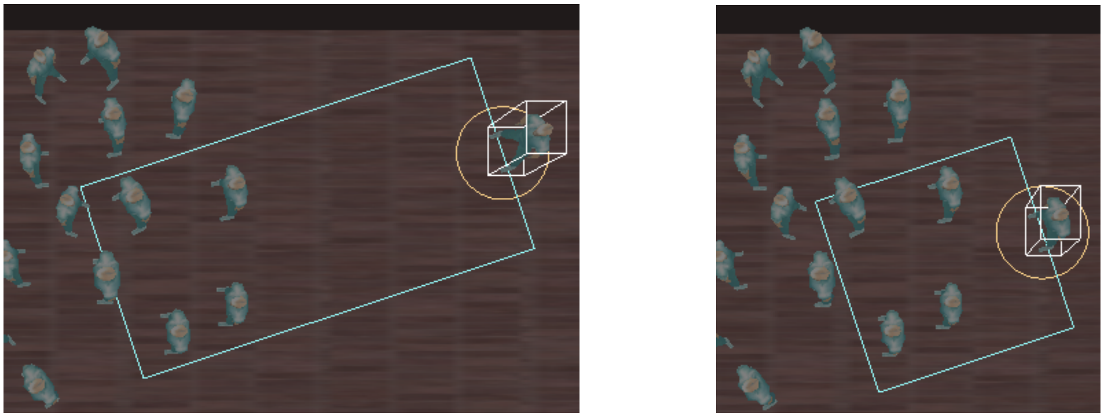


Fig. 4: Dynamic change of viewfield

however, if the pushing is too hard then it may happen that the agents are running directly on the fallen agents. This is done by assigning parameter values in the equations which compute agents' movements.

Panic propagation. For modeling panic behaviour, we modify physiological and psychological parameters of an agent. The panic factor of an agent, if increased, allows for a higher maximum speed and acceleration rate. In addition, panicing agents do not wait for others, and their personal circle is decreased, thus making such agents more active at pushing. Panic propagation is modeled by means of the panic level parameter: this level gets increased by each contact with a panicing agent, and as soon as it is greater than the patience factor, the agent itself becomes panicing.

Right move preference. In order to realistically model the collision prevention among two agents, we provide agents with a preference to move rather to the right than to the left.

Dynamic sight adaptation. We adapt the viewfield of agents depending on the density of the crowd: in a very dense environment, the agent takes care of its near neighborhood and ignores the agents which are far from it, see Figure 4.

III. Distributed simulation system

In order to organize an efficient simulation process for our crowd model, we address two issues: 1) we reduce the amount of information which is updated in each simulation step by means of the intelligent interest management, and 2) we parallelize/distribute the simulation computations by employing multiple servers and thus accelerating the computations.

Interest management. Interest management stands generally for the differentiation between important and not important information. With it, each client receives only those information updates which are relevant of its simulation state. The AoI (Area of Interest) management deals with the updates of not only agents but also other entities, e.g. obstacles. Figure 5 shows the effect of applying the AoI management.

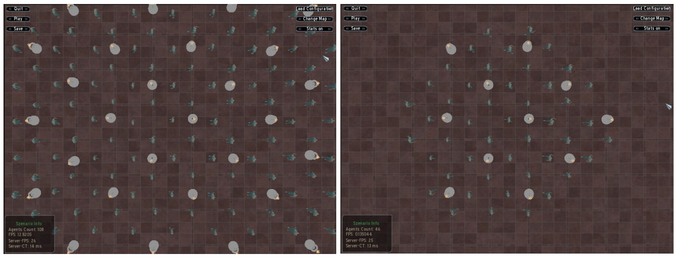


Fig. 5: Area of Interest: off (left) and on (right)

Distribution of simulation. For distributing the simulation among several servers, we use the Real-Time Framework (RTF). The intuitive technique traditionally used in many distributed applications is 'zoning': the environment is split into disjoint zones, in which computations are handled by different servers. For crowd simulations, the 'zoning' approach has several drawbacks. First, agent interaction over zone borders is prevented, since information is exclusively available only to one responsible server. Thus, an agent cannot make a decision based on observing other remote agents, which is often necessary in practical scenarios. Moreover, when simulating dense crowds, we cannot distribute the computational workload where it is especially needed: zone borders can only be placed in sparsely populated areas, thus eventually leaving the simulation of a very densely populated area to one server. Finally, strict separation of data among servers requires the client, responsible for visualization, to communicate frequently with every single server in order to render a complete picture of the simulation state.

The novelty of our system is the use of 'replication' rather than 'zoning' for computation distribution. Replication means that each server holds the complete simulation data, see Fig. 6. Each server is computing updates only for its so-called active agents; all other agents are called shadowed on this server, and their updates are computed by other servers (every agent is active on exactly one server) and received from them. This allows us to distribute the workload evenly between servers, even in densely crowded scenarios, without hindering agent interaction as with 'zoning'. Additionally, a client now only

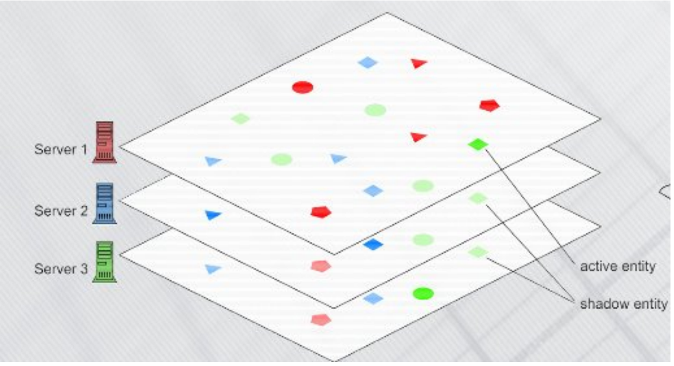


Fig. 6: Replication: active and shadow agents

needs to connect to one server to receive a complete picture of the simulation state for visualization. Replication in our system is implemented using RTF which supports both replication and zoning and advanced combinations of both. The simulation environment is described on a high level of abstraction in an RTF-specific 'map' which determines the distribution of geometrical space on available servers. Our current system employs a single area replicated over the network: each server comes with its own HiDAC unit. Using mechanisms offered by RTF, agents can be added to a unit, removed from it, and migrated to a different unit at runtime.

IV. Event Management

An event in our simulation system is a short-lived, serializable object which is used within the simulation process to send messages between the hosts (clients and servers). The event management subsystem is usually initiated by the client, e.g., to add or remove a particular obstacle. Another kind of interactions happen among servers in order to reflect changes of the simulation state.

The transparent distribution of simulation is supported by event management: interactions concerning the whole simulation are implemented as atomic multicasts; interactions concerning different servers are forwarded automatically; interactions must not bring the simulation into a non-consistent state.

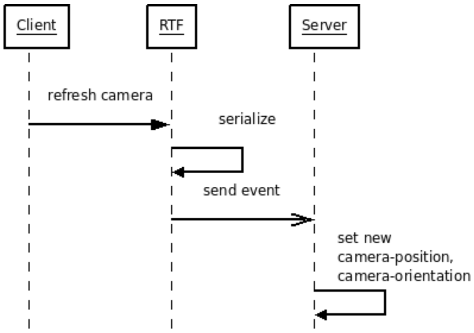


Fig. 7: Event ClientRefreshCameraPosition

As an example, we consider the situation when a client changes its camera focus (viewfield) during the simulation process. In this case, the new position and the orientation of the camera are read from the data of the virtual camera. As soon as these values' changes are greater than a predefined threshold, the client sends an event with the actual data to the server to which this client is connected. The server updates the viewfield of the client and uses it for the AoI management. The event is implemented as a message which is sent only to the connected server, see Figure 7.

Another example is the `ClientToggleDoorState` event for opening/closing a particular door. As a rule, doors are managed by a single server, such that a client which is not connected to this server should forward the event to it, see Figure 8.

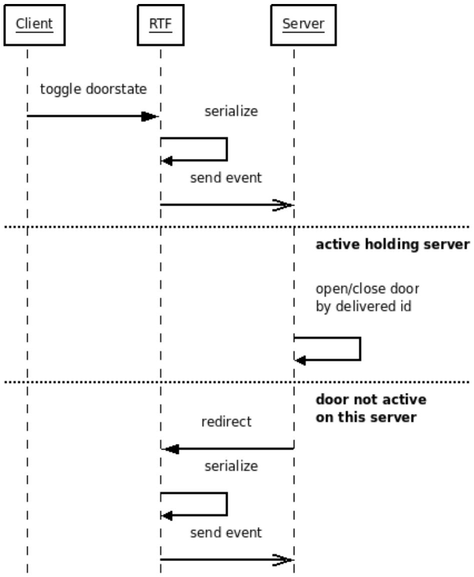


Fig. 8: Event ClientToggleDoorState

A special case is the event `ClientAddAgent`, with a possibility for a server to forward the event to another server. This happens if this other server manages fewer agents than the server to which the event was initially sent. Forwarding the message allows to balance the load between servers.

The only exception in the event management is the event `ServerSendDoorWeights` which is not sent from a client but rather from each replicated server to the activating server. This server manages a global object which contains the actual door weights of the simulation. The event sends the difference vector which reflects the change during the previous simulation step on the server. This vector is then used for computation on the server which manages the global object.

V. Experimental Results

In order to assess the performance of our simulation system, we conducted a series of tests in a high-load setup which emphasized those elements of a simulation scenario that

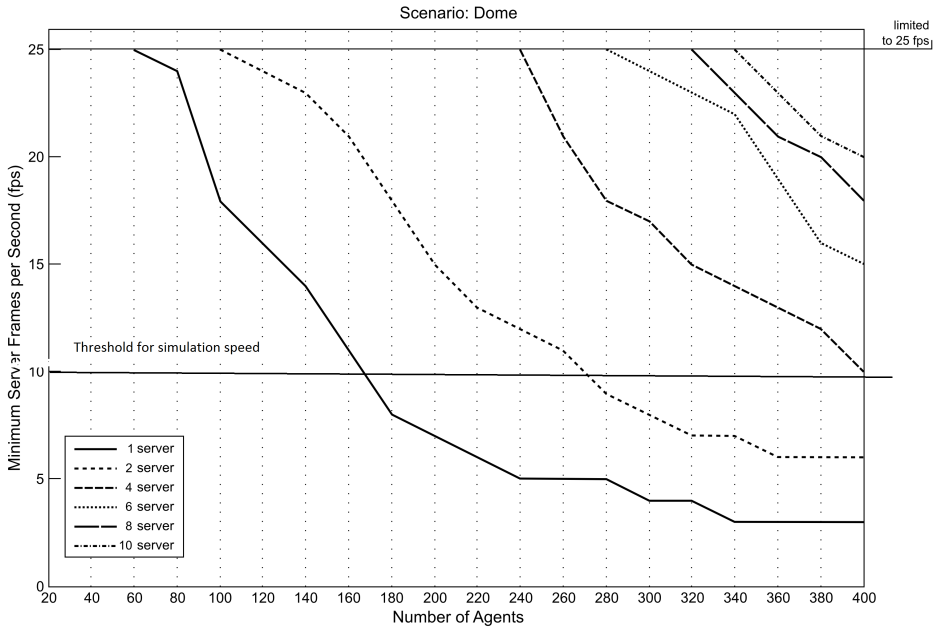


Fig. 9: Simulation speed on 1 to 12 servers.

lead to bottlenecks in the system's performance. We studied a complex indoor environment with many rooms and one large, unobstructed area, which is much more challenging than the simpler scenarios which are usually studied in the literature. While other agents hidden from agent's sight can be disregarded in many calculations, open space takes away this potential performance gain. Also, our testing setup ensures permanent agent movement because this induces additional computational workload. E.g., a scenario with 400 stationary agents may require less computing power than a scenario with 200 moving agents. Measurements were conducted on a local network of common desktop PCs (servers).

The measured value in the experiments is the rate of simulation in *frames per second* (fps) on the weakest system server. Measurements were done as follows: First, the server environment was prepared, comprising 1, 2, 4, 6, 8, 10, or 12 servers. Then, the test scenario was populated with 20 agents. After 1 minute runtime, the server simulation speed was measured. The simulation then again was populated with 40 to 400 agents, with a step of 20 agents, and measurements were taken again after 1 minute.

Our series of tests with the evacuation scenario for the St. Paulus Cathedral in Muenster (a medieval building of about 5000 sqm with a complex system of doors) produce the results shown in Figure 9. We observe that an increase in the number of servers allows for the simulation of more agents, or, at a fixed number of agents, increases the rate of simulation in *fps*. A value of 10 fps is an empirically found threshold to ensure correct calculations in our implementation: rate $\ll 10$ fps may lead to calculation errors, e.g., agents passing through walls. As shown in Figure 9, four servers already suffice to achieve this threshold for up to 395 simulated agents. Regarding scalability, one server can simulate 170 agents at 10 fps, whereas two servers manage 280 agents at the same frame rate (an increase of 64%), and four servers can increase this number further to 395 (132%).

VI. Conclusion

In this paper, we extended and modified the HiDAC approach to crowd simulation [5]. The advantages of our system include: flexibility (interactive changes at runtime), extensibility (accommodating new behavioral factors) and efficiency (real-time response) and, most importantly, the scalability over the number of servers used for the simulation of especially large, dense crowds. We also demonstrated that our Real-Time Framework (RTF) [2], originally created for applications like multi-player online games, supports high-performance distributed implementation of agent-based simulations at runtime and ensures their high scalability.

References

[1] Michael Batty. Polynucleated Urban Landscapes. *Urban Stud*, 38(4):635–655, 2001.

[2] Frank Glinka, Alexander Ploss, Sergei Gorlatch, and Jens Müller-Iden. High-level development of multiserver online games. *Int. Journal of Computer Games Technology*, 2008(5):1–16, 2008.

[3] D. Helbing, L. Buzna, A. Johansson, and T. Werner. Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions. *Transportation Science*, 39(1):1–24, 2005.

[4] R. L. Hughes. The Flow of Human Crowds. *Ann. Rev. of Fluid Mechanics*, 35:169–182, 2003.

[5] Nuria Pelechano, Jan M. Allbeck, and Norman I. Badler. Virtual crowds: Methods, simulation, and control. *Synthesis Lectures on Computer Graphics and Animation*, 3(1):1–176, 2008.

Distributed Simulation on a Many-Core Processor

Karthik Vadambacheri Manian and Philip A. Wilsey
Experimental Computing Laboratory,
School of Electronic and Computing Systems,
PO Box 210030, Cincinnati, OH 45221–0030
vadambkk@mail.uc.edu, and philip.wilsey@uc.edu

***Abstract*—Parallel Discrete Event Simulation (PDES) using distributed synchronization supports the concurrent execution of discrete event simulation models on parallel processing hardware platforms. The multi-core/many-core era has provided a low latency "cluster on a chip" architecture for high-performance simulation and modeling of complex systems. A research many-core processor named the Single-Chip Cloud Computer (SCC) has been created by Intel Labs that contains some interesting opportunities for PDES research and development. The features of most interest in the SCC system are: low-latency messaging hardware, software managed cache coherence, and (user controllable) core independent dynamic frequency and voltage regulation capability. Ideally, each of these features provide interesting opportunities that can be exploited for improving the performance of PDES. This paper reports some preliminary efforts to migrate an optimistically synchronized parallel simulation kernel called WARPED to an SCC emulation system called *Rock Creek Communication Environment* (RCCE). The WARPED simulation kernel has been ported to the RCCE environment and several test simulation models have also been ported to the RCCE environment. Based on initial efforts, some preliminary insights on how to exploit some of the exotic features of SCC for increasing the performance of PDES applications is noted.**

***Index Terms*—Parallel and Distributed Simulation, Time Warp, Many-core processors.**

I. Introduction

Discrete Event Simulation (DES) is widely used for performance evaluation across many disciplines, including: computer systems, computer networks, wired and wireless networks, emergency evacuation management, wargaming, and others [1], [2]. Often the simulation models grow very large and can easily exceed the capabilities of large single-processor compute platform. Thus, many simulation analysis activities are based on fairly small simulation models whose behavior is projected into the larger reality that the simulation is attempting to model. The results from these projections can be highly inaccurate [1]. PDES propose to help solve this problem by running the simulation on a cluster of computers, but it has thus far failed to deliver a promise of reliable speedup across many applications. This failure is largely due to the huge mismatch in speeds of execution vs communication of classic parallel platforms. The integrated solution and more uniform performance between communication and computation on Intel's SCC [3], [4] many-core solution will provide a key opportunity for parallel simulation to begin having a dramatic role in the cloud services community.

In PDES, the concurrently executed simulations communicate by exchanging time-stamped event messages. Unfortunately, the event processing step in most simulation applications are fine-grained computations that generate one or more (but only a few) new events per event execution. One of the key problems in optimizing parallel simulation is finding adequate event processing work during the higher latency event communications. Intel's SCC many-core processor chip provides a low latency on-chip communication medium that should substantially reduce the time disparity between event processing costs and event communication costs.

The Intel SCC chip also has other features that present exciting and unique opportunities for optimizing parallel software codes. For example, the current SCC chip has cache memory that does not enforce coherence; it also has program controlled power and frequency islands allowing independent frequency/voltage settings among subsets of the processor cores. From the perspective of optimistically synchronized parallel simulation (e.g., Time Warp [1], [5]), the ability to use software to modulate power and frequency islands provides an opportunity for the simulation kernel to slow the processing that occur off the critical path and accelerate the processing on the critical path (within the bounds of satisfying the total processor thermal envelope). Thus, balancing the load and potentially accelerating the critical path of the total parallel simulation (similar to Intel's dynamic overclocking). The optimistic nature of Time Warp synchronized simulations may also allow one to exploit the incoherent caches by allowing continued execution for some time without forcing unnecessary coherency checks. However, suitable algorithms to successfully exploit this are not yet known to the authors.

This work focuses on the potential opportunities between Time Warp parallel simulation and the Intel SCC many-core platform. The principle objective of this work is to develop techniques to affect ultra-high performance parallel simulation on many-core processors and ultimately on cloud services provided by clusters of many-core processors. To pursue these investigations, the WARPED simulation kernel [6] is used. WARPED was developed at the University of Cincinnati for supporting large scale simulations (millions of concurrently

Support for this work was provided in part by the National Science Foundation under grant CNS–0915337. The Intel c++ compiler used for this project is provided by Ohio Super Computing Center.

ISBN: 978-1-61208-169-4

executed simulation objects) on smaller (32-64 node) Beowulf clusters. WARPED is an excellent starting point for this project because it is a modular design setup with threaded objects setup for execution on a heterogeneous Beowulf platform that contains local (shared memory) and remote (Message Passing Interface, MPI, based messaging) communication capabilities.

This paper reports on some preliminary explorations to effectively utilize the SCC many-core processor for efficient parallel simulation modeling and analysis. This work throws light on further research for running PDES on many-core Beowulf clusters. The rest of the paper is organized as follows: Section II presents some background and related work. Section III provides a brief introduction to parallel simulation and WARPED. Section IV describes the challenges faced while porting WARPED to the RCCE environment. Section V describes the experimental setup and presents some preliminary performance results. Section VI presents some future research directions that we hope to follow with SCC. Finally Section VII concludes the paper.

II. Background and Related Work

Cloud computing will play a significant role in the future for providing general purpose and high performance computing capabilities. Furthermore, the cloud computing platform will almost certainly be composed primarily of multi-core and many-core processing nodes. Hence the issues of efficiently running PDES on multi-core and many-core processors in cloud infrastructure is an important area of research worth exploring. Fujimoto *et al* [7] has analyzed this area and studied the different issues encountered while running PDES on cloud. One of the issues is the processing delays of the PDES simulation due to the load sharing of the node with other tasks in the cloud. Malik *et al* [8] has analyzed this issue and come up with a modified version of the Time Warp protocol to mitigate this issue.

PDES researchers have also worked to enhance the performance of PDES execution on multi-core processors [9], [10]. When PDES runs in a cluster of multi-core nodes, communications between the cores will have substantially lower latencies than communications between nodes. Bahulkar *et al* [9] has studied the behavior of communication between the cores and between the nodes and their overall impact in the performance of the simulation. They found that if the frequently communicating cores are present in the same chip, it greatly enhances the performance of the PDES. This has lead them to focus their studies on partitioning and load balancing.

Intel's SCC chip is an experimental many-core processor created by Intel Labs mainly for the purpose of many-core research efforts. SCC is the first Intel chip with x86 compliant cores on a single die. The die has 48 cores organized into 24 Tiles with 2 x86 cores per Tile (Figure 1). The principle features of the SCC platform are: (i) hardware support for message passing between cores implemented by a 2-D on-chip mesh interconnection network, (ii) an absence of hardware cache coherency on the Tile caches, and (iii) a fine grained, software controllable, dynamic power and frequency management capability.

Each SCC Tile contains a hardware router, 256KB of L2 cache for each core (2) on the Tile, a 16KB shared Message Passing Buffer (MPB), and 16KB of L1 caches in each core. The MPB provides high-performance on-chip message passing capabilities. The message bandwidth is around 1 GB/s and on-die 2D mesh bisection bandwidth is 2 Tb/s. The MPB memory is cached only in L1 cache of the core and hardware coherence is not enforced. Hence care must be taken while accessing the MPB memory. Typically the programmer will invalidate the cache entry before accessing the MPB memory. Since the caches are incoherent, a shared memory application running across multiple cores must use software managed coherence to ensure correct memory accesses.

One more interesting feature of SCC is that the software control of the operating frequency and voltage of the processing cores. Specifically the operating frequency of the cores and the 2D communication network can be controlled by the executing software. As illustrated in Figure 1, the frequency and voltage adjustment occurs in groups of cores (called *islands*) on the chip. Each Tile forms a *frequency island* and 2x2 groups of Tiles form *voltage islands*. Thus there are a total of 28 frequency domains (24 for the processing cores; one each for the system interface, the voltage regulator controller, and the 2D communication network and memory controller) and 7 voltage domains (6 domains for the cores and 1 domain for the 2D communication network). All of the cores in a frequency island will share the same frequency and all cores in a voltage island will share the same voltage. Frequency changes take only a few cycles whereas voltage changes occur on the order of a million cycles. Hence in addition to voltage change instructions, additional instructions are also provided to check whether the voltage change is complete. The inter-core latency within the SCC chip over a 2D-message network is directly proportional to the number of hops taken by the packet. As Figure 1 shows, by default 12 cores in each quadrant are mapped to a specific memory controller. External memory requests are serviced by these memory controllers.

III. Parallel Simulation and WARPED

Research in parallel and distributed simulation focuses primarily on distributed synchronization mechanisms and the methods to optimize them [1]. A distributed simulation will organize a sequential simulation into concurrently executing parts that are called *Logical Processes* (LPs). The LP will concurrently process events (following some synchronization protocol) and exchange timestamped messages to communicate event information designated for another LP.

There are two main categories of synchronization protocols for distributed simulation, namely: (i) *conservative* [11], and (ii) *optimistic* [5], [12]. Conservative techniques implement a strict enforcement of the "happens-before" relationship between events [13] to synchronize the LP event processing activities. In contrast, optimistic techniques do not strictly enforce the event causality relations. Instead optimistically

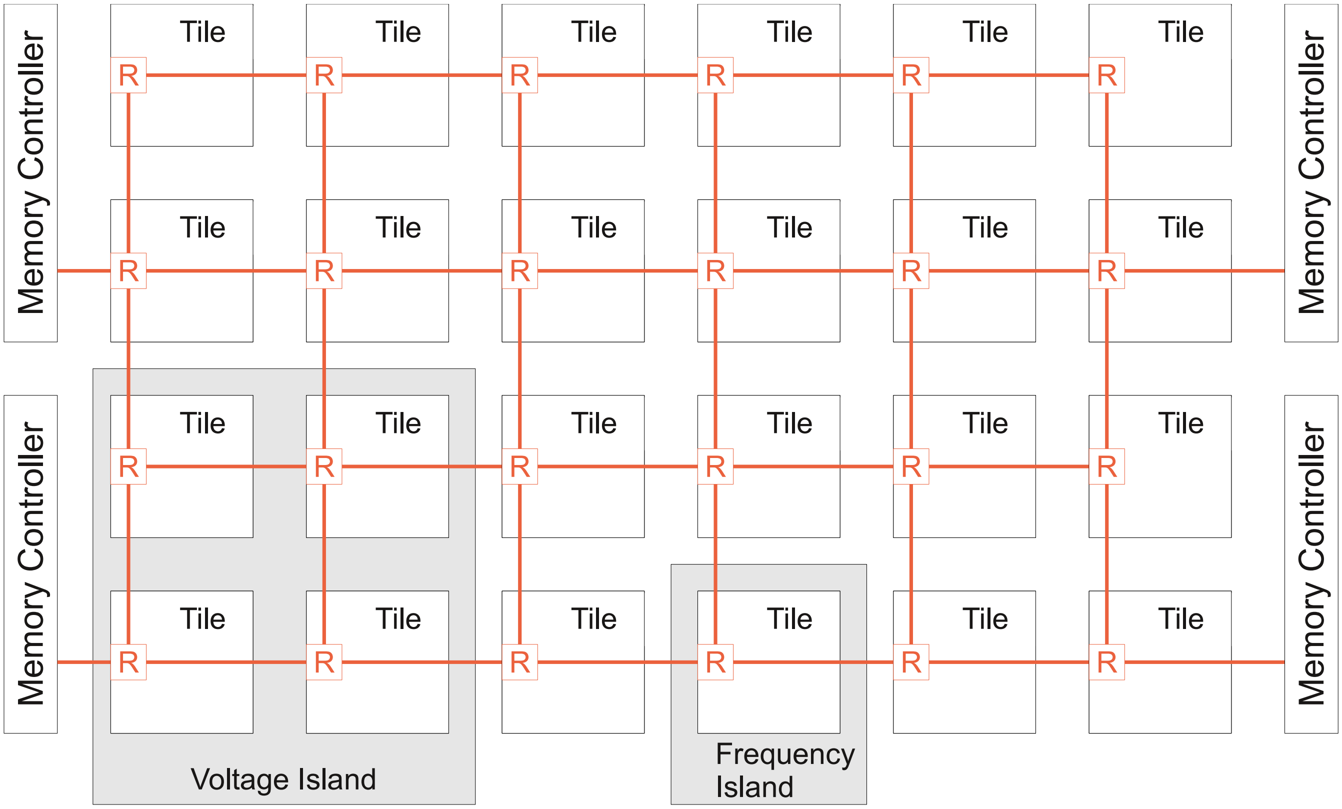


Fig. 1. Architecture of Intel's SCC Processor

synchronized simulations will have some mechanism to detect and recover from an event causality error. This permits optimistic techniques to aggressively process the distributed events and permit greater amounts of parallelism. Of course this comes at the cost of also potentially triggering causality violations that must be repaired.

This paper studies parallel simulation on many-core processors using a simulation kernel called WARPED [6], [10], [14]. WARPED is both a general purpose discrete event API for building simulation models and an implementation of a discrete event simulation kernel (implementing the aforementioned API). The WARPED simulation kernel is highly configurable and has optimized implementations of a sequential and a parallel execution mode. The parallel version implements the Time Warp protocol [1], [5] for synchronization. The design goals of WARPED are to support exploratory research in PDES and to simplify the construction of simulation models for parallel execution. More precisely, the WARPED API hides the implementation details from the simulation model developer. The WARPED code also includes several test simulation models, namely: (i) the classic PHOLD model used by many parallel simulation researchers [1], (ii) a configurable simulation model of a RAID-5 storage array, and (iii) a generic shared memory multi-processor (SMMP). Parallel execution of VHDL models using the WARPED kernel is also possible using the SAVANT/TyVIS tools [15].

Originally WARPED was configured as a collection of highly optimized heavy-weight processes designed to run on Beowulf clusters containing (single or multiple) single-core processors using primarily distributed memory and message passing for communication [6]. More recently the kernel has been expanded and tuned for multi-core processing [10]. While the work to optimize WARPED for execution on multi-core processors is underway, the architecture of many-core processors contain exotic features such as on die network interconnect, no hardware cache coherency, dynamic voltage and frequency regulation, and so on that are not found on conventional multi-core processors. Hence the results obtained for running PDES on multi-core processors cannot necessarily be generalized to many-core processors.

IV. Porting Issues

The experiments with the SCC many-core platform were performed in a software emulation environment that executes on a conventional x86 platform. The emulation environment is called *RCCE*. The RCCE environment provides a framework for software development that closely emulates the SCC communication environment. The WARPED kernel and simulation models are written in C++ and must be migrated to the constrained, shared memory environment and support libraries available for the SCC platform.

The primary language environment for SCC is C and the message passing environment is primarily designed to support a SPMD/synchronous communication application programming environment. Fortunately interfacing the WARPED C++ code with the RCCE API is fairly straightforward. However, there were several challenges to be overcome before the WARPED simulation models could be successfully executed in the RCCE emulator, namely: (i) the use of complex static variables in WARPED, (ii) the asynchronous communication patterns used in the WARPED simulation models, (iii) WARPED has variable length messages sent between LPs, and (iv) issues with the RCCE emulator while executing PDES on more than 30 cores (While this is not a problem that is overcome in these experiments, this section explains why experiments were limited to 30 cores in the emulator). Each of these issues is discussed more fully in the sections below.

A. Static variables issue

The RCCE emulator uses OpenMP to emulate the SCC message passing environment. If the program running in RCCE emulator contains static variables, then they will be initialized only once and shared between all threads. Unfortunately WARPED code contains a significant amount of static variables that are designed to be static to a specific thread, not to all threads. To overcome this issue the RCCE manual recommends the usage of the `#pragma openmp threadprivate` directive on static variables whenever they are encountered and the code is compiled. However, in the current versions of g++, the `threadprivate` directive only works for Plaid Old Data-types (POD) such as `int`, `float`, *etc* and do not work for non POD types such as class objects. Fortunately Intel's icpc C++ compiler can process complex data types in the `threadprivate` directive. Thus, a licensed version of the Intel compiler had to be obtained from the Ohio Supercomputing Center and the WARPED code was modified to build correctly with the icpc compiler.

B. Asynchronous Communication Pattern

The RCCE communication system is designed to support synchronous communication, where a message *send* request must wait for the corresponding message *receive* request. This is not a good match for the asynchronous message passing scenario programmed into WARPED. That is, a WARPED LP sends the message asynchronously and continues with other work. It then periodically polls back to check whether new messages incoming have arrived. Fortunately there exists an asynchronous communication library for the RCCE platform called *immediate RCCE* (iRCCE) [16]. Unfortunately, the ping-ping example pattern in the iRCCE manual and other sample codes in the Many-core Applications Research Community (MARC) [17] forums have all used both non-blocking send or receive in the same function of the application and then they use a blocking call to poll and check whether the requests have completed. Even this communication pattern is not useful for WARPED. WARPED completely decouples the send and receive primitives into separate functions and no blocking call can be used after the send or receive.

The solution that was ultimately successful is to put the asynchronous send and receive requests in a per thread global wait-list. Whenever the application needs to check for messages, it simply checks this global wait-list for completed tasks. This test can be achieved in a non-blocking manner. This method was programmed into the WARPED code for execution with RCCE.

C. Arbitrary length messages

The LPs in a WARPED simulation can exchange multiple message types of varying lengths. Examples of these message types are: initialization, event message, Global Virtual Time (GVT) estimation messages, tests for termination, and so on. As explained in the previous subsection, when messages are obtained from the global waitlist, any type of message can be received from LPs on any other core. Therefore, the message type and size for the next message cannot be known. Unfortunately, the RCCE/iRCCE platform requires that the message size in the send and receive operations match. Thus, the ported WARPED messaging subsystem was modified to send each message in two parts, the first part is a message header containing the length of the actual message and the second part is the actual message. The receive operation is likewise broken into two receives: the first receive reads the message length information and uses that information to trigger the specific command to receive the actual message. Since the order of the messages is guaranteed, this is a workable solution.

D. #pragma omp flush issue

The ported version of WARPED runs in parallel on the RCCE emulator up to 30 cores. However when core count is increased beyond 30, the message headers become polluted, with payload data from the previous message. Problems similar to this are reported in the MARC forums. This may be due to a `#pragma omp flush` issue reported in the MARC forum where the MPB does not reflect the latest content after being written by a thread. As a result, no experimental results are shown in this paper with SCC node counts above 30.

V. EXPERIMENTAL SETUP AND RESULTS

The simulation experiments were run on two machines. The first is an Intel Core i7-920 with 4 hyper-threaded cores supporting 8 threads and operating at 2.67 GHz. The second is a dual core Intel Core2Duo supporting 2 threads operating at 2.00GHz. Both machines have 3 Gb of RAM and are running Linux (version 2.6.x).

Four simulation models are packaged with the WARPED simulation kernel, namely: PHOLD, RAID, SMMP, PING-PONG. PHOLD is a synthetic simulation widely used by the parallel simulation community for showing performance results. The PHOLD configuration used in these experiments contains 4 LPs with an event density of 4 and with an exponential distribution and a seed of 1.0. The RAID simulation simulates a RAID 5 disk array composed of 4 disks and with total of 100 I/O requests issued by two LPs. SMMP is a simulation model that simulates a symmetric multiprocessing environment containing 8 processors, with cache speed 10

Model	Runtime (secs)
PHOLD	835.50
RAID	72.40
SMMP	229.50
PINGPONG	49.50

TABLE I
SIMULATION ON 30 CORES WITH THE INTEL I7

MPB Size	i7 Runtime (secs)			
(bytes)	PHOLD	RAID	SMMP	PINGPONG
100	1.03	0.16	5.32	4.08
150	1.01	0.15	5.32	2.10
200	1.00	0.15	5.27	2.53
8K	0.98	0.14	5.27	1.93

TABLE II
MPB ANALYSIS WITH 8 EMULATED SCC CORES ON THE I7

times that of main memory and with cache hit ratio of 0.85. During the simulation 1,000 memory requests are made to the memory space by each of the 8 simulated processors. Finally, the PINGPONG simulation contains a fixed set of balls that are circulated among a fixed set of players (LPs). A subset of players start the simulation by circulating the balls to other players. The simulation ends when all the balls are received back at the originating LP.

A summary of the simulation runtimes for the emulated 30 core SCC platform is shown in Table I. These results were run on the Intel i7 platform and they simply show the completion of all of the simulation models on an emulated configuration of 30 cores for the SCC platform. In the next two sections, studies to evaluate the impact of message size and to show the potential impact that voltage and frequency adjustments might have are described.

A. Analysis of MPB size

To show the impact of the message passing buffer size on simulation performance, the above simulation models were run in the SCC emulator for varying sizes of the MPB (Table II). By default, the MPB for each core is 8K. However, the maximum message size used by the simulation models is 235 bytes. Hence the default 8K is more than sufficient for these simulation models. The simulations were run on the Intel i7 and results are presented in Table II. The Table clearly shows that the MPB size affects mainly the PINGPONG simulation and other simulations are not significantly affected. Thus the performance impacts depends not only on computation load of the processors but may also be due to their communication pattern. This is because, of the 4 simulations, SMMP and PINGPONG have simulation objects executing on all the 8 cores and even SMMP have more simulation objects than PINGPONG. But interestingly PINGPONG is more affected by MPB variation than SMMP. This may be due to more inter-core communications in PINGPONG than in SMMP. But this needs to be verified further by a detailed investigation on this subject.

Model	# SCC cores	Core id	Time (sec)	Rollbacks
PHOLD	4	0	346.87	2462
		1	352.06	3794
		2	352.57	2911
		3	352.76	2640
RAID	4	0	26.36	381
		1	26.53	175
		2	26.52	1344
		3	26.54	836
SMMP	4	0	71.92	20052
		1	71.91	2359
		2	71.94	1201
		3	71.92	4531

TABLE III
SIMULATION RESULTS FROM CORE2DUO

VI. PDES RESEARCH DIRECTIONS WITH SCC

With the changes outlined in Section IV, all of the WARPED example simulation models (except VHDL, which was not yet attempted) were run on the RCCE simulator. The work is still embryonic and just barely scratched the surface of possibilities and opportunities with the SCC platform. However, even in this preliminary state, one can draw some interesting insights. These are described below.

A. Harnessing voltage and frequency control of SCC

The dynamic voltage and frequency control features of SCC could be highly useful for balancing and optimizing the performance of Time Warp synchronized PDES simulations. In particular, the concurrent LPs of a Time Warp simulation process events aggressively without regard for all event causalities. Thus, some LPs may have frequent rollbacks while others (on the critical path) may have minimal rollbacks. For example, the above simulation models were run for a configuration of 4 emulated SCC cores and detailed runtime and rollback numbers were collected. These results are shown in Table III. The Table shows that the simulation results for RAID show that the LP on core #1 has only 175 rollbacks while the LP on core #2 has 1344 rollback. Likewise SMMP shows widely varying rollback performance among the various cores. By decreasing the frequency of the cores having excessive rollbacks and increasing the frequency of cores having minimal rollbacks, the simulation may actually be able to accelerate the critical path of execution for faster overall simulation throughput. Thus, on many-core processors with suitable thermal monitoring and frequency control capabilities, application specific dynamic overclocking can function to maximally increase overall throughput. Unfortunately there is no way known to test this hypothesis with the RCCE software emulator.

B. Impact of communication on simulation performance

Another interesting area would be to study the communication between the cores of SCC within a single chip and communication between cores of SCC on different nodes and their impact on the overall performance of PDES. This is similar to the study of Bahulkar *et al* [9] but now on many-core processors.

C. Combining shared memory and distributed memory Time Warp protocols

In addition to the per-core private memory and message passing buffer, SCC has a significant amount of off die memory shared between the cores. The amount of shared memory is configurable. Time Warp protocol is conventionally used on either shared memory or distributed memory architectures and the design of each system varies significantly [6], [18]. The SCC provides a unique opportunity to take the best of both worlds and to come up with an efficient combined solution. A similar work is done by Sharma *et al* [19] on clusters of multiprocessors. The emphasis of their work is on exploiting the parallelism of Symmetric MultiProcessing (SMP) node rather than integrating both shared and distributed memory time warp designs. One important hurdle to cross in this direction is maintaining the cache coherency. In SCC no hardware cache coherency is present for want of scalability of the cores. Hence cache coherency in SCC has to be maintained in software which by itself is an interesting research direction.

D. Multilevel Time Warp

Traditional Time Warp optimizations are designed to hide the high network latency by performing useful work during the network communications. However, the high speed on chip network on the SCC processors supports a relatively low network latency. Hence it may be time to revisit the classical Time Warp optimizations such as lazy cancellation to see whether their overhead outweighs their merit in many-core chips. Finding optimizations for PDES on many-core chips is a new avenue for research. Further, in the cluster of many-core processors, the events can be obtained from a local or remote cores. Hence classical Time Warp optimizations can applied to remote events and switch to many-core specific optimizations for events from local core. Hence multilevel Time Warp protocol can be used to efficiently handle both the local events and remote events. More study needs to be done in this area to see the extent of practical usefulness.

VII. Conclusion

This initial work with parallel simulation on the RCCE emulator has provided a few insights on programming needs for future many-core processors. This is a first step in analyzing the potential perform of the many-core SCC platform for efficiently supporting Time Warp synchronized parallel simulation. The possibility to adjust frequency and voltage settings to optimize critical path performance (while maintaining safety under the processor's thermal limits) is an interesting prospect for study. Likewise, the incoherent caches on the SCC platform present opportunities. The dynamic state saving in Time Warp and the opportunity to repair damage from incorrect or premature computations may allow for the development of algorithms to exploit the incoherent caches in interesting ways to increase performance. In any event, the features of many-core processors present numerous interesting opportunities and challenges for the parallel simulation community.

References

[1] R. M. Fujimoto, "Parallel discrete event simulation," *Commun. ACM*, vol. 33, pp. 30–53, October 1990.

[2] A. M. Law and W. Kelton, *Simulation Modeling and Analysis*, 3rd ed. Mc Graw Hill, 2001.

[3] Intel Press Release, Intel Corporation, "Futuristic intel chip could reshape how computers are built, consumers interact with their pcs and personal devices," Intel Press Release, Intel Corporation, Tech. Rep., Dec. 2009. [Online]. Available: http://www.intel.com/pressroom/archive/releases/20091202comp_sm.htm

[4] J. Howard *et al.*, "A 48-core IA-32 message-passing processor with DVFS in 45nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, 7-11 2010, pp. 108 –109.

[5] D. Jefferson, "Virtual time," *ACM Transactions on Programming Languages and Systems*, vol. 7, no. 3, pp. 405–425, Jul. 1985.

[6] D. E. Martin, P. A. Wilsey, R. J. Hoekstra, E. R. Keiter, S. A. Hutchinson, T. V. Russo, and L. J. Waters, "Redesigning the warped simulation kernel for analysis and application development," in *Proceedings of the 36th annual symposium on Simulation*, ser. ANSS '03, 2003, pp. 216–223.

[7] R. Fujimoto, A. Malik, and A. Park, "Parallel and distributed simulation in the cloud," *SCS M&S Magazine*, 2010.

[8] A. Malik, A. Park, and R. Fujimoto, "Optimistic synchronization of parallel simulations in cloud computing environments," in *Proceedings of the 2009 IEEE International Conference on Cloud Computing*, ser. CLOUD '09, 2009, pp. 49–56.

[9] K. Bahulkar, N. Hofmann, D. Jagtap, N. Abu-Ghazaleh, and D. Ponomarev, "Performance evaluation of pdes on multi-core clusters," in *Proceedings of the 2010 IEEE/ACM 14th International Symposium on Distributed Simulation and Real Time Applications*, ser. DS-RT '10, 2010, pp. 131–140.

[10] R. Miller, "Optimistic parallel discrete event simulation on a beowulf cluster of multi-core machines," Master's thesis, University of Cincinnati, Cincinnati, OH, 2010.

[11] J. Misra, "Distributed discrete-event simulation," *Computing Surveys*, vol. 18, no. 1, pp. 39–65, Mar. 1986.

[12] K. M. Chandy and R. Sherman, "Space-time and simulation," in *Distributed Simulation*. Society for Computer Simulation, 1989, pp. 53–57.

[13] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Communications of ACM*, vol. 21, no. 7, pp. 558–565, Jul. 1978.

[14] R. King, "WARPED redesigned: An api and implementation for discrete event simulation analysis and application development," Master's thesis, University of Cincinnati, Cincinnati, OH, 2011.

[15] P. A. Wilsey, D. E. Martin, and K. Subramani, "SAVANT/TyVIS/WARPED: Components for the analysis and simulation of VHDL," in *VHDL Users' Group Spring 1998 Conference*, Mar. 1998, pp. 195–201.

[16] C. Clauss, S. Lankes, J. Galowicz, and T. Bemmerl, "ircce: A non-blocking communication extension to the rcce communication library for the intel single-chip cloud computer," RWTH Aachen University, Tech. Rep., Feb. 2011. [Online]. Available: http://communities.intel.com/message/110482#110482

[17] "Marc - manycore application research community." [Online]. Available: http://communities.intel.com/community/marc

[18] S. Das, R. Fujimoto, K. Panesar, D. Allison, and M. Hybinette, "Gtw: a time warp system for shared memory multiprocessors," in *Proceedings of the 26th conference on Winter simulation*, ser. WSC '94, 1994, pp. 1332–1339.

[19] G. D. Sharma, R. Radhakrishnan, U. K. V. Rajasekaran, N. Abu-Ghazaleh, and P. A. Wilsey, "Time warp simulation on clumps," in *Proceedings of the thirteenth workshop on Parallel and distributed simulation*, ser. PADS '99, 1999, pp. 174–181.

ISBN: 978-1-61208-169-4

Simulation of Particle Deposition in an Airplane Cabin Mockup

Miao Wang
School of Mechanical Engineering
Purdue University
West Lafayette, IN 47906, USA
wang283@purdue.edu

Chao-Hsin Lin
Environmental Control Systems
Boeing Commercial Airplanes
Everett, WA 98124, USA
chao-hsin.lin@boeing.com

Qingyan Chen
School of Mechanical Engineering
Purdue University
West Lafayette, IN 47906, USA
yanchen@purdue.edu
and
School of Environmental Science and Technology
Tianjin University
Tianjin 300072, China
yanchen@tju.edu.cn

Abstract— **Accurate prediction of particle deposition in airliner cabin is important to estimate exposure risk of occupants to infectious diseases. This investigation simulated airflow field, particle dispersion and deposition in a half-occupied four-row cabin mockup using a Detached-Eddy Simulation (DES) model with a modified Lagrangian method. Three types of particles with diameters of 0.7, 10 and 100 μm were studied that represent different particle dispersion and deposition processes. This study tested two flow scenarios: one is a breathing case in which particles were released from an index occupant with very small inertial force; and the other is a coughing case in which the particles were released by a high momentum jet flow. This study found that the DES model with the modified Lagrangian method can predict reasonably good results for air velocity, particle concentration and deposition in the cabin environment. The particle deposited depended on particle size and inertial forces. For the breathing case, the deposition rate on the cabin surfaces was 35% for the small (0.7 μm) particles, 55% for the medium (10 μm) particle and 100% for the large (100 μm) particles. In the coughing case, the particle deposition was enhanced due to the high initial velocity. The particle deposition rate was 48%, 69%, and 100% for the small, medium and large particles, respectively.**

Keywords-CFD; experiment; particle; deposition; indoor

I. INTRODUCTION

Over four billion people arrive at and depart from airports all over the world every year. This figure will double by 2025, according to a long term traffic forecast [1]. Commercial airplane passengers travel in an enclosed cabin environment at close proximity [2]. During the long time of air travel, the exposure risk to infectious diseases can be very high. Mangili and Gendreau [3] evaluated the risk of infectious disease transmission in commercial airplane cabins and concluded that air travel was an important factor in the worldwide spread of infectious diseases.

Infectious disease transmission in airplane cabins can occur in many ways, such as direct contact with contagious particles generated from an infected person, inhaling pathogenic airborne agents or droplets, or touching contaminated surfaces. These different disease transmission paths are all closely related to the deposition and transport of contaminant particles or droplets. For example, saliva droplets generated by an index person through coughing or sneezing can deposit directly on the mouth or eyes of another person. The dose of airborne infectious agents and droplets is associated with their deposition rate and transport path, and a surface in an airplane cabin can be contaminated by the trapping of contaminant particles. As the commercial airplane cabins are crowded and packed with different solid surfaces, their influence on particle deposition and transport can be significant. Therefore, it is essential to evaluate the level and distribution of particle deposition in a cabin environment.

The rapid growth of computer power makes CFD a promising tool for predicting airflows, particle transportation, and deposition in enclosed environments [4,5]. For cabin airflow and contaminant transport simulation, Baker et al. [6] validated their CFD prediction of air velocity and mass transport inside an aircraft cabin using measurement data. Zhang et al. [7] measured and simulated gaseous and particulate contaminant transport in a four-row cabin mockup. Poussou et al. [8] simulated transient flow and contaminant concentration field in a small-scale cabin mockup with a moving body. These studies explored complicated airflow and contamination concentration fields inside a cabin environment. However, particle deposition on cabin surfaces was neglected in these cases, which could be significant for a crowded cabin environment.

Particle deposition has been studied by many researchers, however, for other enclosed environments. Lai and Nazaroff [9] applied an analogous model for particle deposition to smooth indoor surfaces and predicted a reasonable result for simple geometry. Lai and Chen [10] conducted a Lagrangian simulation for aerosol particle transport and deposition in a chamber and found good agreement between their CFD result and the empirical estimation. Zhao et al. [11] simulated particle deposition in ventilated rooms. Their deposition results agreed with the measured data at low turbulence level, but failed to match the experimental data when the turbulence was high. Zhang and Chen [12] simulated particle deposition on differently oriented surfaces inside a cavity using a modified Lagrangian method and predicted improved results. Although reasonable prediction of deposition was reported by many studies, the relatively simple geometry and airflow conditions in these cases may not guarantee a good result in a much more complex environment such as an airplane cabin. A study of the literature showed that particle deposition inside an airplane cabin has not been well investigated by either numerical or experimental studies.

Using numerical simulations, this paper aims to extend the understanding of contagious particle depositions inside an airplane cabin environment. This investigation first evaluated a modified Lagrangian particle deposition model with Detached Eddy Simulation (DES) [13] and applied it to a four-row cabin mockup. The simulation included two flow scenarios, one breathing and talking case, and the other a coughing case. The particle depositions on different cabin surfaces were determined from the simulation results. The study discussed the deposition statistics and identified key factors related to particle depositions in airplane cabins.

Section I of this paper introduces the background information of this study. Section II shows the numerical models used in the simulation. Section III shows the test case and simulation results. Section IV discusses the result. Section V concludes this study.

II. Airflow and Particle Phase Models

Accurate models of airflow and turbulence in an indoor environment are important for predicting the particle transportation and deposition process. This study used the DES Realizable k-ε model [13], which can provide accurate prediction of air velocity and turbulence quantities [14]. Due to limited space available, the formulation of this model was not included in this paper, but can be found from literature [13].

With the airflow information, this study modeled the particle dispersion and deposition with a Lagrangian method, which can be expressed as:

$$\frac{d\vec{u}_p}{dt} = F_D\left(\vec{u}-\vec{u}_p\right) + \frac{\vec{g}\left(\rho_p-\rho\right)}{\rho_p} + \vec{F} \qquad (1)$$

where $\vec{u}_p$ and $\vec{u}$ are the particle and air velocities, respectively; ρ_p and ρ are the densities of particles and air respectively; $\vec{g}$ is the gravitational force; $\vec{F}$ is other forces such as the Thermophoretic force, Saffman lift force, and Brownian force; and is the drag coefficient.

In (1), the term $\vec{u}$ represents the actual airflow velocity, which should be written as:

$$\vec{u} = \bar{u} + u' \qquad (2)$$

where $\bar{u}$ is the velocity solved by the DES model, u' is the turbulence velocity component that should be properly modeled. Although there is no model available for the DES model used in this study, many models have been developed for different RANS models, which may be used by the DES in its near wall region. This study applied a deposition model proposed by Matida et al. [15].

$$u_i' = \begin{cases} \zeta_i A u^* y^{+2} & y^+ \le 4 \\ \zeta_i \sqrt{2k/3} & y^+ > 4 \end{cases} \qquad (3)$$

where $A = 0.008$ is a constant, u^* is the shear velocity, and y^+ is the distance from a particle to the nearest wall in the wall unit.

III. Prediction of Particle Deposition in a Four-row Airplane Cabin

A. Case Description

Fig. 1 depicts the schematic of the four-row twin-aisle cabin mockup. In the experiment [7], the cabin mockup had 28 seats, 14 of which were occupied by human simulators, as shown in red in the figure. The air was supplied from two groups of linear diffusers located near the center of the ceiling. The total airflow rate was 0.23 m^3/s, or 8.2 L/s per passenger seat. Three-dimensional air velocity and air temperature were measured at two planes, as depicted in green in Fig. 1. The air velocity and temperature profiles at the inlet diffuser and the temperature of different surfaces were also measured.

The particle source was located at the center seat of the third row (seat 3D), as shown in Fig. 1. Non-evaporative, monodispersed Di-Ethyl-Hexyl-Sebacat (DEHS) particles were released from the source into the cabin with a small momentum. After the airflow and particle field reached a steady-state, the particle concentration was measured at eight positions, as shown in Fig. 1. The particle used in the experiment had a diameter of 0.7μm. However, in the CFD simulation, three sizes of particles (0.7, 10, and 100μm) were simulated to study the influence of particle size on the particle deposition.

The numerical simulation was conducted based on CFD code ANSYS FLUENT (version 12.1). The study applied the DES Realizable k-ε model with the modified Lagrangian method as discussed before. The simulation used a solution from the RNG k-ε model as the initial field and calculated 10 minutes of flow time to reach the steady-state flow field. Then, the particles were continuously released from the source into the cabin and were mixed with the cabin air. For each particle size, 1000 particles were generated every second. The case was calculated for another 15 minutes of flow time until the particle concentration field reached steady-state, which corresponded to six complete air changes in the cabin. The averaged air velocity, particle concentration, and deposition results were obtained in the next five minutes of flow time.

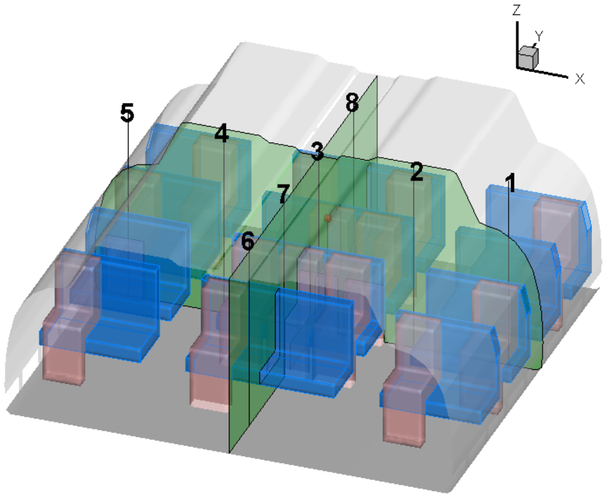
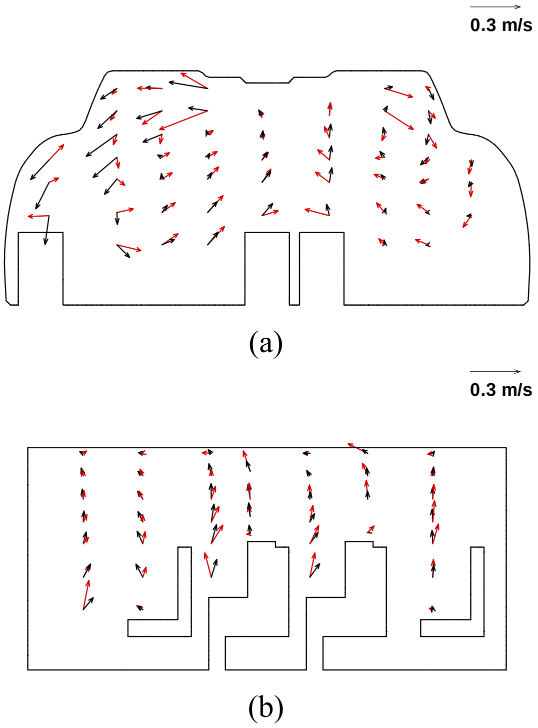


Figure 2. Schematic of the four-row cabin mockup [7].



(a)

(b)

Figure 1. Comparison of simulated (black vectors) and measured (red vectors) airflow field at: (a) the cross-section through the third row, and (b) the mid-section along the longitudinal direction.

B. *Air Velocity Field*

Fig. 2 compares the simulated and measured air velocity vectors at the cross-section through the third row and at the mid-section along the longitudinal direction. In the cross-sectional view (Fig. 2 (a)), the ceiling diffusers and the thermal plume in the middle generated two large circulations at each side of the cabin. The prediction agreed with the measurement in terms of circulation pattern. But significant discrepancies can be found in a quantitative comparison. Similar results were also reported by Zhang et al. [7], who concluded that the simulation was very sensitive to the accuracy of the boundary conditions, which may not have been accurately measured. Note that the airflow field was asymmetrical due to the inlet and wall-boundary conditions.

In the mid-section along the longitudinal direction, the vector field shows an upward motion due to the two circulations and the thermal plume in the middle of the cabin. The CFD result agreed reasonably well with the measured data as shown in Fig. 2 (b), though differences can be found at some positions. For example, at the third row, the CFD model predicted a backward airflow motion, which was not supported by the measurements. At the same location, the CFD results also predicted a smaller upward velocity than did the experiment.

C. *Particle Deposition onto Different Surfaces*

1) *Breathing and Talking*

In the experiment, the particles were released with a very small initial velocity, which could be representative of the particle release from the breathing or talking of a passenger. The distribution of the particle deposition at solid walls and exhaust vents was also calculated for five minutes of flow time in this investigation. The density of the deposition was calculated as:

$$\bar{C}=\frac{N_{dA}}{N_{total}dA} \qquad (4)$$

where N_{dA} was the number of particles deposited on surface area, dA, during a certain amount of time; N_{total} was the total number of particles generated during the same time; and dA was a small surface area, which was the same as the computational mesh.

Fig. 3 shows the normalized particle deposition density of the 0.7, 10, and 100 μm particles. Due to the asymmetrical airflow pattern, the deposition was also asymmetrical. For the small (0.7 μm) particles, a high particle deposition density was observed at the ceiling and side walls along the path of the major circulation (Fig. 3(a)), while the floor and seat had relatively low deposition density (Fig. 3(d)). This is because the particles were small, and mainly followed the airflow pattern. The small 0.7 μm particles were carried by the thermal plume to reach the ceiling, where most particles joined the airflow circulation formed by the supply jets. The particles deposited at the ceiling and side walls along their path to the exhaust.

For the medium (10 μm) particles, Fig. 3(b) shows that their deposition at the ceiling and side walls was similar to that of the small particles, but the deposition rate was much lower. The deposition density at the floor was higher. As the particle size increased, the gravitational force became comparable to the drag force, which changed the deposition distribution.

For the large (100 μm) particles, Fig. 3(c) shows no deposition on the ceiling and side walls. All the particles were deposited at the surfaces of passenger 3D, as shown in Fig. 3(f). For particles of this size, the gravitational force was dominant. The particles had a free fall motion from its source (mouth/nose) and deposited within a very small area on passenger 3D.

TABLE I. STATISTICS OF PARTICLE DEPOSITION ON DIFFERENT TYPES OF SURFACES

Surfaces	**Deposition Percentage**					
	Breathing and talking			***Coughing***		
	0.7 μm	***10 μm***	***100 μm***	***0.7 μm***	***10 μm***	***100 μm***
Exhaust	65%	55%	0	52%	31%	0
Passenger	7%	8%	100%	20%	14%	2%
Floor	3%	21%	0	15%	49%	91%
Ceiling	8%	2%	0	4%	1%	0
Side wall	12%	10%	0	4%	1%	0
Section end	2%	1%	0	2%	1%	0
Seat back	1%	1%	0	1%	1%	4%
Seat front	1%	1%	0	1%	1%	0
Tray table	1%	1%	0	1%	1%	3%

2) Coughing

This study further modified the initial conditions for the particles so as to study the particle deposition with a cough from a passenger. The inlet velocity, flow rate, area of opening, and angle of the jet flow from the cough were chosen according to Gupta et al. [16]. As in the previous case, seven sizes of particles were continuously released from the cough by the passenger at seat 3D. All the models and simulation procedures were the same as in the breathing and talking case.

Fig. 4(a) shows the normalized particle deposition density of the 0.7 μm particles at the ceiling and side walls. Compared with the previous case, the deposition on the ceiling and side walls was significantly reduced. This was because the jet flow that carried the particles could penetrate the thermal plume. Therefore, most of the particles did not enter the major circulation so they could not reach the ceiling. For the deposition on the floor and seats, Fig. 4(d) shows a high deposition density on the seat back of passenger 2D, the surface of passenger 3D, and the floor area close to seat 3D, due to the jet impingement.

For the 10 μm particles, Fig. 4(b) shows a lower deposition density at the ceiling and side walls than that for the breathing and talking case. As shown in Fig. 4(e), a high deposition density was observed in the areas of jet impingement. Unlike the 0.7 μm particles that mostly suspended in the air after entering the air, a majority of the 10 μm particles deposited due to the jet momentum and the gravity.

For the 100 μm particles, Fig. 4(c) shows that no particles deposited on the ceiling and side walls. All the particles deposited on the back surface of seat 2D, the surface of passenger 3D, and the floor close to seat 3D due to direct impingement and gravity because these particles were too heavy to be carried by the airflow.

IV. DISCUSSION

Table I shows the statistics of the particle deposition on different types of surfaces. For the breathing and talking case, 65% of the 0.7 μm particles were removed by air through the exhaust. The side walls and ceiling trapped a large portion of the particles (12% and 8%, respectively). These surfaces may not be frequently contacted by passengers. The passenger surfaces had 7% of the 0.7 μm particles. Despite the large area, the floor only received 3% of the particles. The two section ends trapped 2% of the particles because the airflow along the longitudinal direction was small. The seat front, seat back, and tray tables trapped 3% of the particles that could likely be touched by the passengers. For the 10 μm particles, the number of particles exhausted was reduced to 55%, but was still a majority. The deposition on the ceiling decreased to 2% since gravity became important for this size of particle. For the 100 μm particles, all the particles deposited on the surface of the index passenger, which can be explained by their free fall motion. In general, the gravity force played a major role in the particle deposition.

In the coughing case, the jet could penetrate the thermal plumes and could transport the particles to the lower part of the cabin. The jet impingement enhanced particle deposition on the floor, thus increasing the total particle deposition by 13% and 14% for the 0.7 μm and the 10 μm particles, respectively. For the two particle sizes, the deposition on the passenger also increased. The deposition on the ceiling and side walls decreased slightly. About 91% of the 100 μm particles deposited on the floor, with the rest on the seat back and tray table in front of the index passenger.

V. CONCLUSION

This study applied the DES model with a modified Lagrangian method to predict the particle dispersion in a four-row airplane cabin mockup. By comparing with the experimental data, this investigation found that this new model can predict reasonably good results for air velocity, particle concentration, and particle deposition.

For the cabin case, three sizes of particles were assumed to be released by an index passenger sitting in the middle of the cabin due to breathing with zero velocity and due to coughing with suitable jet velocity. This study found that the distribution of particle deposition onto surfaces depended on particle size, particle release mode, and the airflow pattern in the cabin. In the breathing case, 35% of the small (0.7μm) particles, 55% of the medium (10μm) particles, and 100% of the large (100μm) particles deposited onto the cabin surface and the rest were removed by the cabin ventilation. In the coughing case, the number of small, medium and large particles deposited changed to 48%, 69%, and 100%, respectively.

REFERENCES

[1] ACI. The Global Airport Community, 2007. www.airports.org/aci/aci/file/Annual Report/ACI Annual Report 2006 FINAL.pdf . Last accessed 09/17/2011.

[2] J. D. Spengler and D. G.Wilson, "Air Quality in Aircraft," Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering 217, 2003, pp. 323-335.

[3] A. Mangili and M.A.Gendreau, "Transmission of infectious diseases during commercial air travel," Lancet 365, 2005, pp. 989-996.

[4] P. R. Spalart and D. R. Bogue, "The role of CFD in aerodynamics, off-design," Aeronautical Journal 107(1072), 2003, pp. 323-329.

[5] Q. Chen, "Ventilation performance prediction for buildings: A method overview and recent applications," Building and Environment 44(4), 2008, pp. 848-58.

[6] A. J. Baker, S. C. Ericson J.A. Orzechowski, K.L. Wong and R. P. Garner, "Aircraft passenger cabin ECS-generated ventilation velocity and mass transport CFD simulation: Mass transport validation exercise," Journal of the IEST (Online) 51(1), 2008, pp. 90-113.

[7] Z. Zhang, X. Chen, S. Mazumdar, T. Zhang and Q. Chen, "Experimental and numerical investigation of airflow and contaminant transport in an airliner cabin mockup," Building and Environment 44(1), 2009, pp. 85-94.

[8] S. Poussou, S. Mazumdar, M. W. Plesniak, P. Sojka and Q. Chen, "Flow and contaminant transport in an airliner cabin induced by a moving body: Scale model experiments and CFD predictions," Atmospheric Environment 44(24), 2010, pp. 2830-2839.

[9] A. C. K. Lai and W. W. Nazaroff, "Modeling indoor particle deposition from turbulent flow onto smooth surfaces," Journal of Aerosol Science 31, 2000, pp. 463-476.

[10] A. C. K. Lai and F. Chen, "Modeling of particle deposition and distribution in a chamber with a two-equation Reynolds-averaged Navier–Stokes model," Journal of Aerosol Science 37(12), 2006, pp. 1770-1780.

[11] B. Zhao, C. Yang, X. Yang and S. Liu, "Particle dispersion and deposition in ventilated rooms: Testing and evaluation of different Eulerian and Lagrangian models," Building and Environment 43(4), 2008, pp. 388-397.

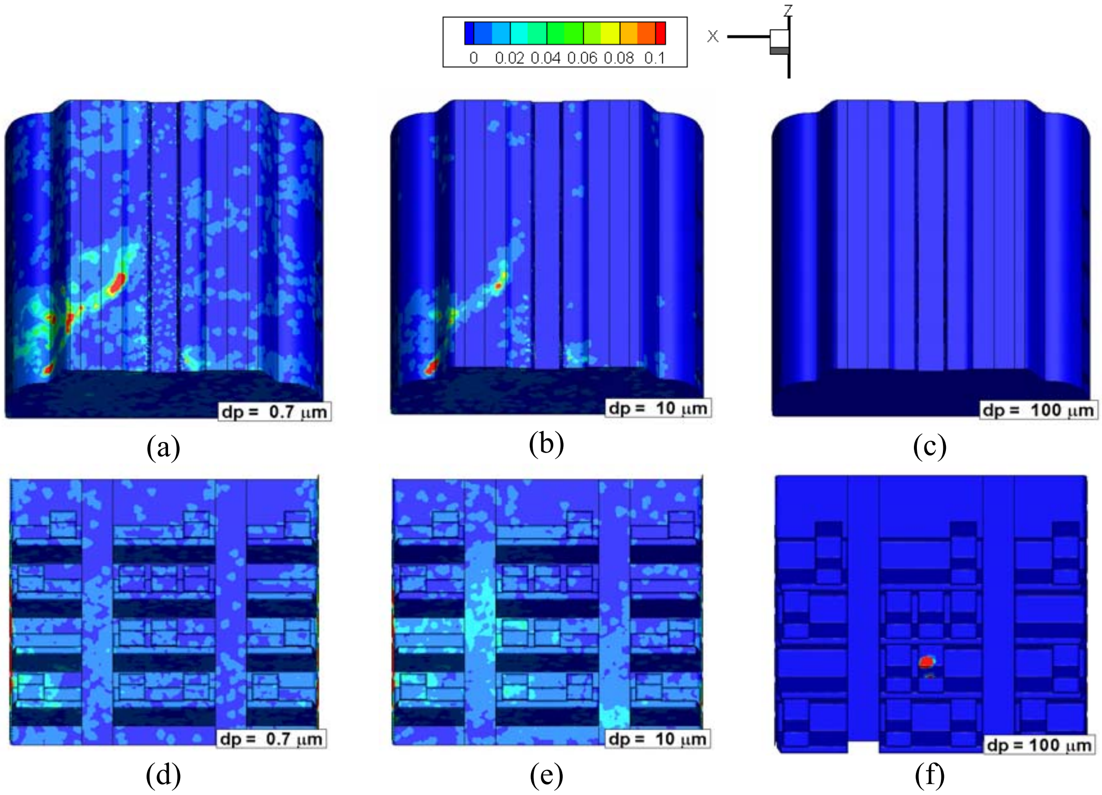
[12] Z. Zhang and Q. Chen, "Prediction of particle deposition onto indoor surfaces by CFD with a modified Lagrangian method," Atmospheric Environment 43(2), 2009, pp. 319-328.

[13] FLUENT, 2005. Fluent 6.2 Documentation. Fluent Inc., Lebanon, NH.

[14] M. Wang, and Q. Chen, "Assessment of various turbulence models for transitional flows in enclosed environment," HVAC&R Research 15(6), 2009, pp. 1099-1119.

[15] E. A. Matida, W. H. Finlay, C. F. Lange and B. Grgic, "Improved numerical simulation of aerosol deposition in an idealized mouth–throat," Journal of Aerosol Science 35, 2004, pp. 1-19.

[16] J. K. Gupta, C.-H. Lin, and Q. Chen, "Flow dynamics and characterization of a cough," Indoor Air 19, 2009, pp. 517-525.



(a) (b) (c)

(d) (e) (f)

Figure 3. Particle depositions at different surfaces for the breathing and talking case: the top row is for the ceiling and side wall surfaces and the bottom row for the floor and seats surfaces (a) and (d) for 0.7 μm particles, (b) and (e) for 10 μm particles, and (c) and (f) for 100 μm particles.

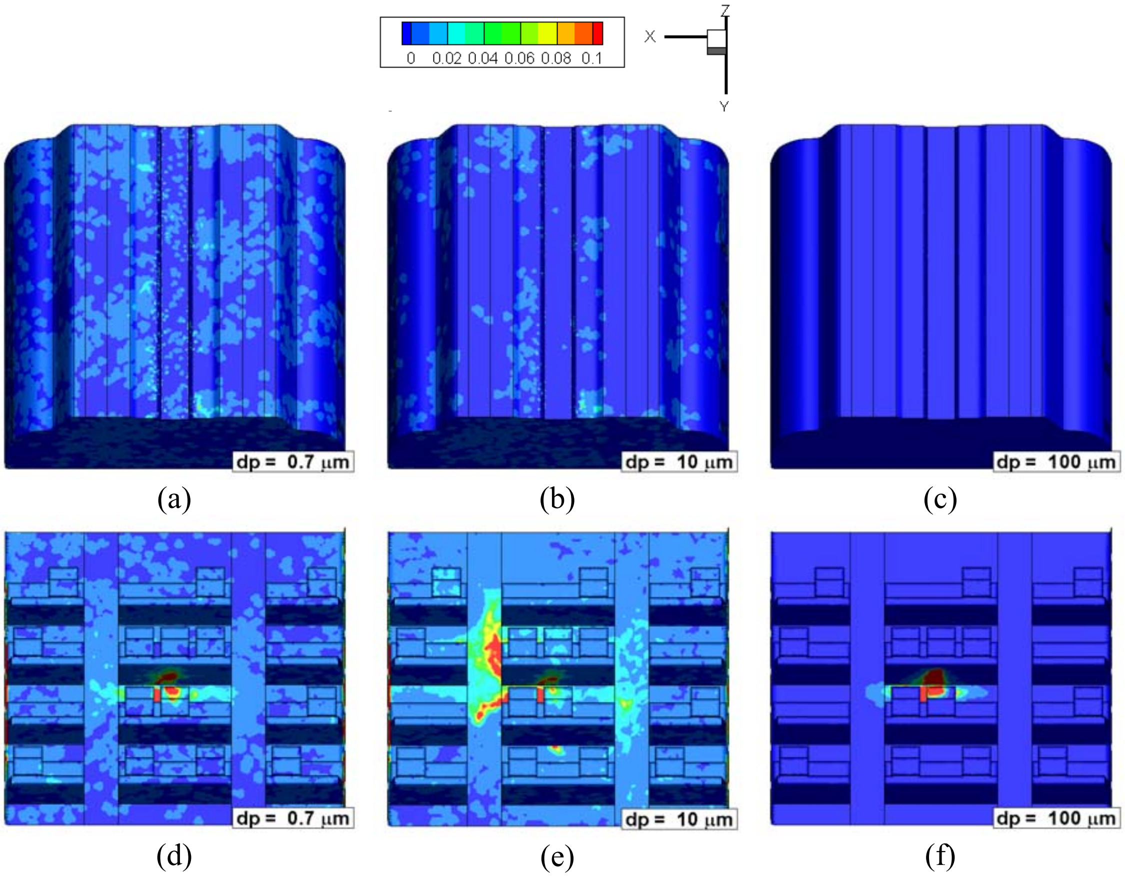


Figure 4. Particle depositions at different surfaces for the coughing case: the top row is for the ceiling and side wall surfaces and the bottom row is for the floor and seats surfaces (a) and (d) for 0.7 µm particles, (b) and (e) for 10 µm particles, and (c) and (f) for 100 µm particles.

ISBN: 978-1-61208-169-4

Traffic and Monotone Random Walk of Particles: Analytical and Simulation Results

Alexander P. Buslaev*, Alexander G. Tatashev† and Andrew M. Yaroshenko‡
* *Moscow Automobile And Road Construction State Technical University*
Moscow, Russia
apal2006@yandex.ru
† *Moscow University of Communications and Informatics*
Moscow, Russia
a-tatashev@rambler.ru
‡ *Moscow Automobile And Road Construction State Technical University*
Moscow, Russia
andreijar@rambler.ru

***Abstract* - An analytical model of random walk of particles on one-dimensional lattices and two-dimensional ones is considered. This model can be used for study of traffic on multi-lane road, traffic on crossroads, etc. We have obtained new exact formulas that allow to calculate the average velocity of particles. The steady state probabilities have been found also. We developed also an appropriate simulation model. The values, obtained with aid of analytical calculations, were compared with simulation results, and the result of this comparison was that the accordance of the models is good correspondence of the values found with aid of the two models.**

Keywords-stochastic models; random walk; multi-lane traffic; optimization.

I. Introduction and formulation of problem

Models of random walk on a lattice, which are used for the traffic study, help to study characteristics of multi-lane traffic, the behaviour of traffic on crossroads, etc. Such models help to solve problems of traffic optimization.

The appearance of cellular automata models in the problem of traffic flows is associated with publications of Nagel et al. [1–2], which were published in mid-nineties. In [1–2], the dependence of the average velocity and intensity of traffic on the model parameters was studied. The following factors, which contribute to the activation of this approach development, can be noted

1) Desire to explain the discrepancy between the solutions of traffic equations and experimental observations, which discovered chaotic behaviour in the so-called unstable mode;

2) Desire to create models discrete in time and space or only by one coordinate, which would be independent of infinite-dimensional models and take into account the presence of individual behaviour of ”particles with motivated behaviour”.

We enumerate papers known to us that contain exact mathematical results and relate to the theme of our paper.

It has been noted that the scheme considered in [1–2] is similar to monotone random walks on a lattice. This theme has its own history. In particular, the papers of Soviet mathematician Belyaev and his students [3, 4] are devoted to traffic flows in the underground and contain exact results for one-dimensional random walk (not only monotone).

There are a lot of experimental results obtained with aid of cellular automata models. That is apparently due to the relative availability of computational tools. A Russian mathematician Blank gave well-defined statements and achieved exact results in a number of important cases, [5–6].

For example, one of problem in this scope is the problem of percolation of fast particles through a lattice with a slow particles on it.

Here we are talking about exact estimates, while simulation results such as the experimental observation about the significant influence of the presence of heavy trucks on traffic intensity certainly exist.

In [7], a model of particles (vehicles) movement on a multi-lane road was considered. In this model the velocity of movement is the sum of determinate and stochastic components, i.e., $v_{model} = v_{determ} + v_{stochast}$. In the model considered in [7] the determinate component of movement corresponds to the background movement on lane and the stochastic component corresponds to individual manoeuvres of particles. Each lane corresponds to the sequence of cells. The size of the cell is determined by the dynamic dimension of the vehicle. The dynamic dimension takes into account the safety requirement and depends on velocity of movement, [8]. Stochastic movement is described by monotone walk on cells of lane and the regular movement is described by uniform movement of the lane, [7], [9].

ISBN: 978-1-61208-169-4

In [10], a model of random walk on one-lane ring has been considered. The formula has been found for the average velocity of particles. This formula is a generalization of formula, obtained in [5], for the model of random walk, where randomness occurs only for initial configuration of particles.

In [11–13], models of random walk on a discrete lattice, similar to models introduced in [7, 9], have been used for the solving some problems of traffic optimization.

The present work considers a stochastic model, which describes movement of particles (vehicles) on the multi-lane road. The steady state probabilities and the average velocity of particles have been obtained. The configurations of particles on the lattice determine the model states. The limiting case is studied, in which the length of lane is infinitely large.

Some simulation models have been developed, which are used for the study of characteristics of single-lane and multi-lane movement.

The developed simulation models were realized using Delphi 7 environment.

The volume of PC computers memory was sufficient for the case of nearly 200 cells of the lattice.

A minute of computer program time corresponds to nearly 500 units of discrete model time.

If the simulation time interval is equal nearly 30000 units of discrete model time, the difference between values of average particles velocities found with aid of simulation and analytical models does not usually exceed 1 %.

II. Single-lane and multi-lane flows

A. Single-lane flows

Let us describe the stochastic model of movement on the circle, which was considered in [10]. The ring contains n cells and $m < n$ particles. If at the current time there is a particle in the cell and the next cell is empty then the particle moves to the next cell with probability p. The steady states probabilities and the average velocity of particles have been found. States of the model are determined by the configurations of particles on the lattice. As shown in [10], the probabilities of states depend on the number of particle clusters only.

Let us consider a mathematical model of movement on the lane.

This model is the limiting case of the model studied in [10]. In this case the number of cells is big. The formula for average velocity v has been obtained:

$$v = \frac{1 - \sqrt{1 - 4rp(1-r)}}{2r}, \qquad (1)$$

where r is the flow density, $r = m/n$.

In [10], a formula has been obtained for average velocity $v(n,m)$ of the stochastic movement on non-moving lane for number of cells equal n, number of particles equal m, and the probability of realization of attempt equal p. In accordance with this formula

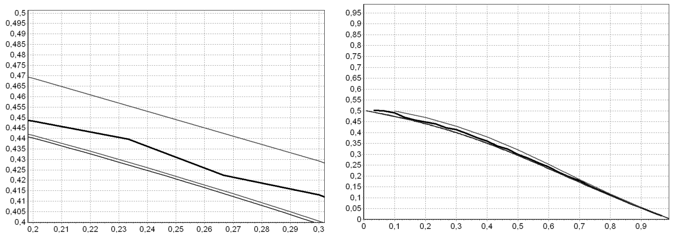


Figure 1. Average velocity (y-axis) of particle movement on the ring for n equal to 10 (the upper curve), equal to 100 (the second curve from the top), 170 (the second curve from the bottom) and ∞ (the lower curve), simulation for $n = 30$ (the second curve from the top). Values of density are indicated on x-axis. The same curves are represented on both the diagrams but the scale of left diagram is larger

$$v(n,m) = \frac{n}{m} \sum_{k=1}^{\min(m,n-m)} \frac{Cp}{(1-p)^{k-1}} C_{m-1}^{k-1} C_{n-m-1}^{k-1},$$

where

$$C = \left(\sum_{k=1}^{\min(m,n-m)} \frac{n}{k} \cdot C_{m-1}^{k-1} C_{n-m-1}^{k-1} \frac{1}{(1-p)^{k-1}} \right)^{-1},$$

C_m^k is the number of combinations of k elements from m ones.

Diagrams of $v(n,m)$ for values of n, equal to 10, 100 and 170, and the diagram of function (1), which corresponds to an infinitely great value of n, is showed in Fig. 1. Value of p is supposed to be equal to 0.5. In this figure the corresponding values of the average velocity of particles, which have been obtained by simulation, are also shown. Interval of simulation was supposed to be equal to 10000 simulation steps.

In the case $p \sim 1$ we have, using formula (1),

$$v \sim \begin{cases} 1, & 0 < r \le \frac{1}{2}, \\ \frac{1}{r} - 1, & \frac{1}{2} < r < 1, \end{cases}$$

that corresponds to the results of work [5], where a similar model has been considered for $p = 1$.

In case $p \sim 0$ we have

$$v \sim p(1-r).$$

Function (1) is convex on the flow density. Due to this fact the problem of the optimal distribution of particles on the lanes has been solved for canalized movement. This problem has been formulated in [11].

B. Multi-lane flows

A model of stochastic movement on torus was studied (Fig. 2). There are several lanes. Each lane is a circle that consists of some number of cells. The sections of the torus are also circles and every section contains cells of different

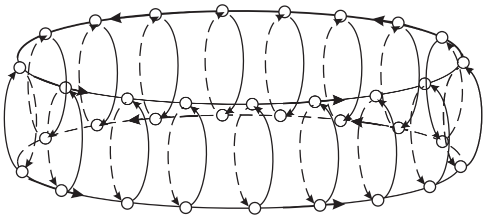


Figure 2. Torus

lanes. The movement of particles is monotone with respect to each coordinate, i.e., the particles can move only in one and the same direction for every coordinate. At the current time with probability $1/2$ the transition of particle in the direction of the first coordinate and with probability $1/2$ the transition of particle in the direction of the second coordinate is planned. The transitions in the given direction are planned simultaneously for all the particles. But every transition is realized with probability that does not depend on behaviour of the other particles. Suppose that at the current time the transition of particle in direction of s-th coordinate is planned, $s = 1, 2$. If the particle is in cell (i, j) and the next cell, in direction of the s-th coordinate, is empty, then with probability $0 < p_s < 1$, which does not depend on results of attempts of the other particles, the particle passes to the next cell in the direction of s-th coordinate, $s = 1, 2$.

The states of the model correspond to configurations of particles. It has been proved that in the case of small values of probabilities p_1 and p_2 probabilities of all the states are asymptotically equal to $1/N$, where N is the number of states. Exact formulas have been found for the average velocity of movement. It is proved that the distribution of particles on the lanes is hypergeometric for the number of lanes equal 2 and a generalization of the hypergeometric distribution if the number of particles is greater than 2.

The behaviour of the model is studied for the case of arbitrary values of p_1 and p_2 and great values of number of cells. It occurs for this case that the particles are distributed on the lanes uniformly. The average velocity can be calculated using formula (1).

III. Simulation model of multi-lane movement. Comparison with the analytical model

A. *The case of non-moving lane*

A multi-lane traffic simulation model has been developed.

We consider the case of non-moving lane, i.e. $v_{det} = 0$. The model contains a two-dimensional array. There are K lanes. Each lane is a sequence of n cells. There are m particles. The transitions of particles occur at integer time steps. Each cell contains no more than one particle. If at the current time the cell ahead of a particle is empty then it passes to the next cell with probability p. If the cell ahead of the particle is occupied and the neighbouring cell of another lane is empty then the particle passes to the neighbouring cell. If the both neighbouring cells on the other lanes are empty then each of them can be chosen with probability $1/2$. If two particles can come to the same cell the priority is given to the particle that is located on the right.

The results of analytical calculations were compared with the results of simulation.

The differences of the models are the following

1) Movement is monotone in direction of two coordinates in the analytical model and movement is monotone in direction of only coordinate in the simulation model.

2) In the analytical model all the lanes are inner.

3) In the simulation model collisions of particles are prevented by restrictions on changing the lane. This collisions are impossible in the analytical model as at each time the transitions in direction of the only coordinate are allowed.

In the analytical model some simplifications are used and analytical results can be obtained. The simulation model describes the real situation more detailed.

In accordance with the rules of movement in the simulation model a particle moves to another lane if there is another particle in the next cell ahead.

Therefore we should expect that the average velocity of particles in this model will be slightly higher than the average velocity of a particle in the mathematical model (Section II), in which transitions to another lane are independent of whether the cell, neighbouring to the cell to be occupied, is empty or occupied.

In Figure 3 the results of analytical calculations and results of simulation are given for the models of two-lane movement. Values of flow density and average velocity are indicated on x-axis and y-axis respectively.

We suppose that the number of cells on each lane is equal to 100. Let p and r be defined as in Section II. Let $v^*(r)$ be the average velocity of a particle that has been found by simulation and $v^{**}(r)$ be the average velocity calculated using formula (1). As it was expected the maximum value of the relative differences

$$\Delta = \frac{u^*(r) - u^{**}(r)}{u^*(r)}$$

is attained at $r = 0.5$ approximately. This can be explained with the following. By small values of the flow density a situation, where particles pass to the other lane, occurs rarely. By great values of flow densities transitions of particles to the other lane are complicated and therefore occur also rarely.

As the number of cell on the torus tends to infinity, the average velocity converges to values calculated with the aid of formula (1). Therefore data, represented in Fig. 3, give an estimation of difference between values of the average velocities in the analytical model of movement on torus and the simulation model of multi-lane movement.

ISBN: 978-1-61208-169-4

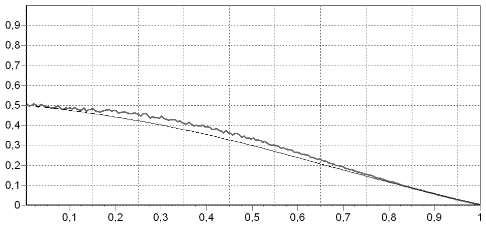


Figure 3. Dependence of average velocity on density for $p = 0.5$. Lower curve corresponds the analytical model. Upper curve corresponds the simulation

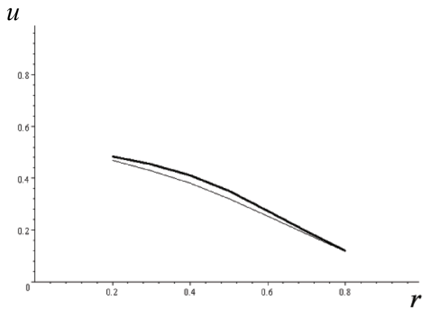


Figure 4. Results of comparing of the two-lane simulation model (the upper curve) and model taking into account a determined component of average velocity (the lower curve). Values of flow density and average velocity are indicated on x-axis and y-axis respectively.

B. Comparison with the analytical model, which takes into account the determinate component of the traffic velocity

In Section IIIa, the comparison of the values of average velocities is represented. We compared also the values of average velocities of particles in simulation and analytical models, developed in [7, 9]. In these models the velocity of movement is represented as sum of the determinate component and the stochastic one. The approach using such representation is mentioned in the Section I.

Results of comparison are represented in Fig. 4. This comparison allows to estimate the differences between the average velocities, which are obtained with the analytical and simulation models. This difference can be explained by that in the simulation model the particles can change lanes and by this reason the average velocity of movement is some higher.

As for models which are considered in Section IIIa the maximum value of the relative differences between values of velocities is attained at $r = 0.5$ approximately.

IV. The "ring with junctions" movement model.

We consider simulation model of traffic with crossroads, that is "figure-of-eight". This model contains the main ring and k rings of smaller size, each of that has one common cell with the main ring, Fig. 5. The flow passing such a cell (crossroads) divides into two parts.

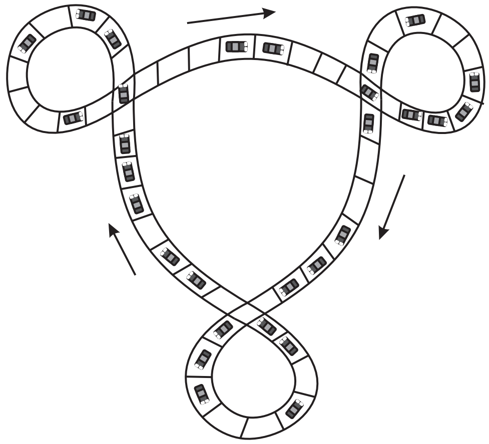


Figure 5. Movement model "ring with junctions"

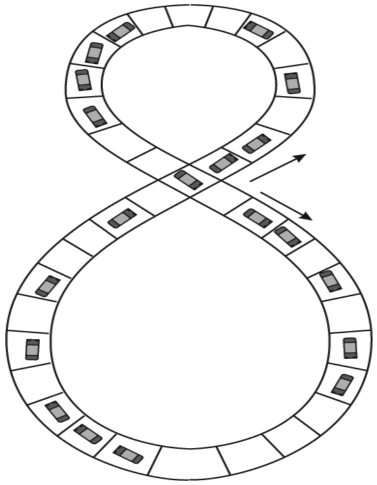


Figure 6. "Figure-of-eight" model

Some of them remain on the main ring and others begin to move on the small ring.

In one of version of the model the particles coming to the crossroads from the main ring have priority and in the second version the particles coming to the crossroads from the small ring have priority.

If $k = 1$ then we have the model of "figure-of-eight". Each ring in the "figure-of-eight" is represented by one-dimensional array, Fig. 6. Two arrays have a common cell, which is a model of cross-road. The common cell is called cell 0. The first ring contains n_1 cells $(1, j)$, $j = 1, 2, \ldots, n_1$. The second cell contains $n_2 < n_1$ cells $(2, j)$, $j = 1, 2, \ldots, n_2$. Coming through the crossroads, particles coming from the first ring have priority over the particles coming from the second ring.

So there are $n = n_1 + n_2 + 1$ cells in the model. There are $m < n_1$ particles. Each of them occupies one cell. There is no more one particle in each cell. Transitions of particles occur at integer time steps. If at integer time k the particle

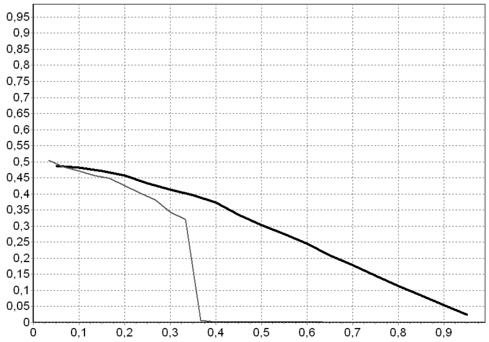


Figure 7. Comparison of diagrams of dependence of average velocity on density for the ”figure-of-eight” (the lower curve) and one-lane ring (the upper curve) models. The values of model parameters are supposed as $n_1 = 20,\ n_2 = 10,\ p = 0,5$. Values of flow density and average velocity are indicated and on x-axis and y-axis respectively.

occupies cell (i, j) and cell $(i, j + 1)$ is empty then with probability $0 < p < 1$ this particle at time $k+1$ will occupy cell $(i, j + 1)$ and with probability $1 - p$ will occupy still cell (i, j), $j = 1, 2, \ldots, n_i - 1$, $i = 1, 2$.

The results of comparison of values of average velocity of particles in the ”figure-of-eight” model and the model of one-lane movement on ring which contains the same number of cells are represented, Fig. 7.

We can see in the figure that for values of m which are sufficiently less than n_1 differences between values, obtained using two models, are small. For values of m, close to n_2, sharp decrease of the average velocity occurs. If $m \geq n_2$ movement stops in some number of simulation steps.

V. Conclusion and future works

An analytical model have been developed where traffic is represented by random walk. Exact results for traffic characteristics have been obtained.

We have developed simulation models taking into account some features of real traffic more detailed. The values, obtained with aid of the analytical calculations, and simulation results was compared. It allows to estimate the difference between average velocities in models with different rules of movement of particles.

We plan to use the simulation model so that it could help to obtain new analytical results.

References

[1] Nagel, K. and Schreckenberg, M. “A cellular automaton model for freeway traffic”, J. Phys. I France 2, 1992, pp. 2221-–2229.

[2] Schreckenberg, M., Schadschneider, A., Nagel, K., and Ito, N. “Discrete stochastic models for traffic flow”, Phys. Rev. E., 1995, vol. 51, pp. 2939–2949.

[3] Belyaev, Yu. and Zele, U. “A simplified model of movement without overtaking”, Izv. AN SSSR, ser. ”Tekhn. kibernet”, 1969, N 3, pp. 17–21.

[4] Zele, U. “Generalizations of movement without overtaking”, Izv. AN SSSR, ser. ”Tekhn. kibernet.”, 1972, N 5, pp. 100–103.

[5] Blank, M. “Exact analysis of dynamical systems arising in models of traffic flow”, Russian Mathematical Surveys, vol. 55, no. 3 (333), 2000, pp. 167–168.

[6] Blank, M. “Dynamics of traffic jams: order and chaos”, Mosc. Math. J., 1:1, 2001, pp. 1-26.

[7] Buslaev, A., Prikhodko, V., Tatashev, A., and Yashina, M. “The deterministic-stochastic flow model”, arXiv: physics/0504139v1[physics/soc.-ph], vol. 20, Apr. 2005, pp. 1–21.

[8] Inose, H. and Hamada, T. ”Road traffic control”, University of Tokio Press, 1975.

[9] Buslaev, A., Novikov, A., Prikhodko, V., Tatashev, A., and Yashina, M. “Stochastic and simulation approaches to optimization of road traffic”, Moscow, Mir, 2003.

[10] Buslaev, A. and Tatashev, A. “Particles flow on the regular polygon”, Journal of Concrete and Applicable Mathematics (JCAAM), vol. 9, no. 4, 2011.

[11] Bugaev, A., Buslaev, A., Tatashev, A., and Yashina, M. “Optimization of partially-connected flows for deterministic-stochastic model”, Trudy MFTI, vol. 2, no. 4(8), 2010. pp. 15–26.

[12] Bugaev, A., Buslaev, A., and Tatashev, A. “Monotone random walk of particles on an integer number lane and LYuMen problem”, Mat. modelirovanie, vol. 18, no. 12, 2006, pp. 19–34.

[13] Bugaev, A., Buslaev, A., and Tatashev, A. “Simulation of segregation of two-lane flow of particles”, Mat. modelirovaniye, vol. 20, no. 9, 2008, pp. 111–119.

Tunnel Simulator for Traffic Video Detection

Sofie Van Hoecke*†, Steven Verstockt*†, Koen Samyn*†, Mike Slembrouck‡, Rik Van de Walle†
*Electronics and Information Technology Lab (ELIT), Howest, Ghent University Association, Belgium
†Multimedia Lab (MMLab), ELIS, IBBT - Ghent University, Belgium
‡Traficon, Trafic Video Detection, Belgium
Email: {sofie.van.hoecke, steven.verstockt, koen.samyn}@howest.be, ms@traficon.be, rik.vandewalle@ugent.be

***Abstract*—Testing and evaluating advanced traffic video detection algorithms and systems requires photorealistic source material to be generated. As recording real life traffic situations has severe limitations, a Tunnel Simulator is developed to create custom test scenarios within tunnels. The Tunnel Simulator enables the creation of custom tunnels by setting properties for the tunnel and individual traffic events. Based on the settings, a photorealistic scene is generated with the specified tunnel, traffic events and ground truth. The scene can then be previewed in real time 3D view and/or rendered in order to test the video detection algorithms on it. The presented Tunnel Simulator is the first high-quality traffic simulator that succeeds in generating photorealistic source material that can be used to evaluate traffic video detection algorithms.**

***Keywords*-simulator, traffic video detection, photorealism.**

I. Introduction

Over the years, traffic volume and complexity have been growing at a steady pace. As a result, traffic managers are faced with an increased demand in automated traffic monitoring systems. These systems do not prevent primary incidents, but once an incident has been positively identified, protocols for reducing secondary incidents are initiated such as tuning green-red cycles of traffic lights, closing the tunnel entrance to prevent collisions, adjusting tunnel ventilation to cut off oxygen supply to fires, or providing paramedics with accurate information and images.

New algorithms and features for automated traffic monitoring using video image processing need extensive testing in order to assess quality and reliability under various conditions. In order to test and evaluate new detection algorithms, accurate video source material is required. This real life recorded video source footage is then fed to the traffic monitoring system and the generated output is compared to a ground truth. In order to do so, a ground truth needs to be manually created for each relevant video feed. As traffic and system complexity grows, developers no longer find the appropriate source material. Despite numerous videos documenting real traffic incidents, many possible scenarios and events remain uncaptured on film. Up till now this has been resolved by creating and recording custom traffic scenarios. However, this method does not allow for dangerous situations to be created, e.g., driving a burning bus through a tunnel filled with regular traffic. Such an event is a good example of potentially catastrophic situations that have to be detected as soon as possible in order to minimize casualties.

As a solution, a Tunnel Simulator was designed and developed that generates user specified video source material and ground truth. The choice for focusing on tunnels was straightforward as a tunnel is potentially the most dangerous place for incidents, illustrated by the tragic disaster in the Gotthard tunnel in 2001 where a collision between two trucks resulted in a fire incident in which eleven persons died and that lasted over 24 hours before fire-fighters were able to bring the situation under control. Ever since, traffic video detection is mandatory in tunnels. Road tunnels must be equipped with the appropriate equipment for detection and monitoring, including sensors for temperature, visibility, CO_2, and smoke, as well as video cameras [1]. Incident detection in tunnels became a key safety aspect for every major traffic manager, and, even today, it remains a very important aspect of automated traffic detection systems.

The remainder of this paper is as follows. Section II outlines the state of the art. Subsequently, Section III defines the advanced features of our tunnel simulator. Section IV covers the internal design details. Next, Section V covers the evaluation results, after which we summarize the most important conclusions of our work in Section VI.

II. State of the art

During the past decades, a considerable amount of research has been done on developing real-time traffic surveillance algorithms based on roadside video aiming to extract reliable traffic state information [2]-[4]. However, the research is limited to detecting events such as stopped vehicles or car collisions in open road. As a result, ground truth material is widely available.

Also different traffic models [5] and simulators have been built [6]-[9]. These simulators are however intelligent transportation systems (ITS) and flow control systems and focus on optimal traffic flow control and calculating the most efficient route to go from A to B.

Closests to our research goal is the tunnel simulator of the Swedish Road Administration (SRA). The SRA has built a Tunnel Simulator [10] in order to provide training for traffic managers, staff, and paramedics. The large variety of objects (e.g., grass, trees, vehicles, billboards, buildings, railings and lights) allows to create accurate descriptions of real tunnels, but the quality of rendered computer-generated imagery (CGI) is low compared to today's standards. To the authors' knowledge, no high-quality traffic simulator has been reported upon yet.

III. Tunnel Simulator

Road tunnel operation depends heavily on traffic control and monitoring, as well as on well prepared and tested emergency and rescue plans [1]. However, the principle risk in road tunnels is the driver, a risk that can never be excluded. Therefore, new algorithms using video image processing are constantly developed to positively identify incidents better and faster.

In order to allow extensive testing of these algorithms to assess quality and reliability, a Tunnel Simulator was designed and developed to generate user specified video source material and ground truth. Figure 1 presents the general concept and Figure 2 a screenshot of the final result.

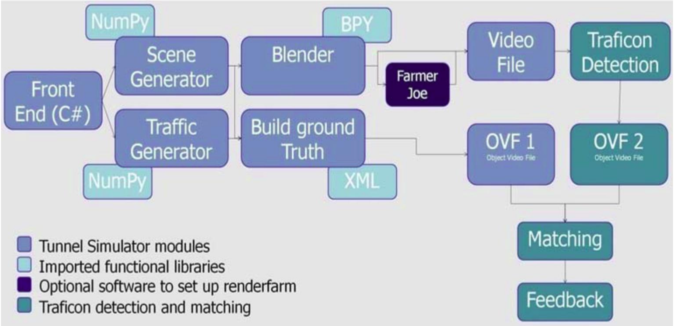


Figure 1. General concept of the Tunnel Simulator

As can be seen on Figure 1, the Tunnel Simulator consists of a scene generating module, a traffic generating module and a module to gather the user's input. The front-end enables users to create a custom tunnel by setting and editing tunnel properties such as height, length, lighting, angle and direction. Similar steps can be followed to create custom traffic events inside the tunnel. The front-end checks if properties are valid before passing them on to the scene and/or traffic generator. Communication between the generators and the 3D application is necessary to create a virtual world with traffic events. The scene generator provides the necessary information to build the tunnel, while the traffic generator adds vehicles and animation paths to the scene, to set up the animations. Each generator provides information to the ground truth module in order to build the based Object Video File (OVF). The 3D application will use its internal render engine to render the CGI. Optional tools (e.g., a render farm) are provided to speed up the rendering process. These loosely coupled modules ensure easy adaptation/replacement to meet future requirements. The Tunnel Simulator will generate two outputs: a video containing the rendered animation and a reference OVF file, containing a description what happens in the video. This video can then be fed to a detection system that will generate a second OVF file. Both OVFs can then be compared in order to assess and evaluate detection accuracy and quality.

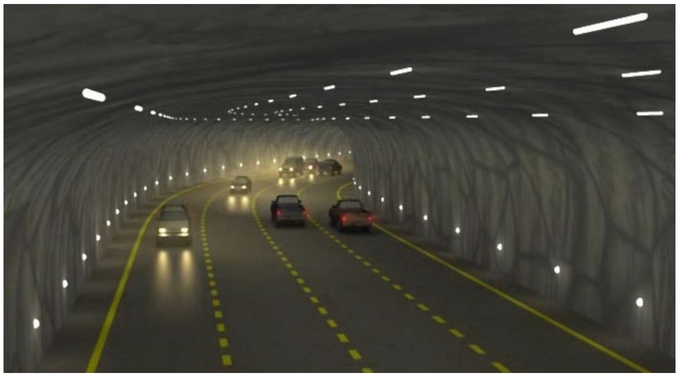
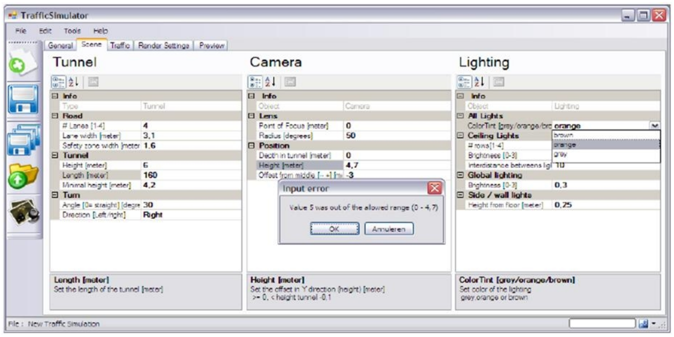


Figure 2. Screenshot of a created tunnel scene by the Tunnel Simulator

The Tunnel Simulator offers advanced features listed below:

1) **User-friendly creation and configuration:** Creating and configuring the tunnel and according scenes is straightforward by using a user-friendly interface so that application training can be minimized. Within this user interface (see Figure 3) all parameters can be set to shape a tunnel and define a traffic situation.



Tunnel

Info	
Type	Tunnel
Road	
# Lanes [1-4]	4
Lane width [meter]	3,1
Safety zone width [meter	1.6
Tunnel	
Height [meter]	6
Length [meter]	160
Minimal height [meter]	4,2
Turn	
Angle [0= straight] [degre	30
Direction [Left/right]	Right

Length [meter]
Set the length of the tunnel [meter]

Camera

Info	
Object	Camera
Lens	
Point of Focus [meter]	0
Radius [degrees]	50
Position	
Depth in tunnel [meter]	0
Height [meter]	4,7
Offset from middle [- +] [m	-3

Input error

Value 5 was out of the allowed range (0 - 4,7)

Height [meter]
Set the offset in Y direction (height) [meter]
>= 0, < height tunnel -0.1

Lighting

Info	
Object	Lighting
All Lights	
ColorTint [grey/orange/brc	orange
Ceiling Lights	
# rows[1-4]	
Brightness [0-3]	
Interdistance betweens lig	10
Global lighting	
Brightness [0-3]	0,3
Side / wall lights	
Height from floor [meter]	0,25

ColorTint [grey/orange/brown]
Set color of the lighting
grey,orange or brown

File : New Traffic Simulation

Figure 3. User interface for tunnel creation and traffic event configuration

2) **Photorealism:** The traffic detection algorithms use specular lightning spots, intensity and reflections to detect vehicles and objects so photorealism is required. In order to achieve this, for example, two light types are used per car: spots to light up the environment, and normal lamps as visible lights on car. Figure 4 shows the result.

3) **Adjustable tunnel shape and driving lanes:** Different shapes of tunnels can be created using the simulator by configuring straight and curved sections, the tunnel shape (e.g., cylinder, rounded rectangle), the height and width, amount of lanes, uni- or bidirectional

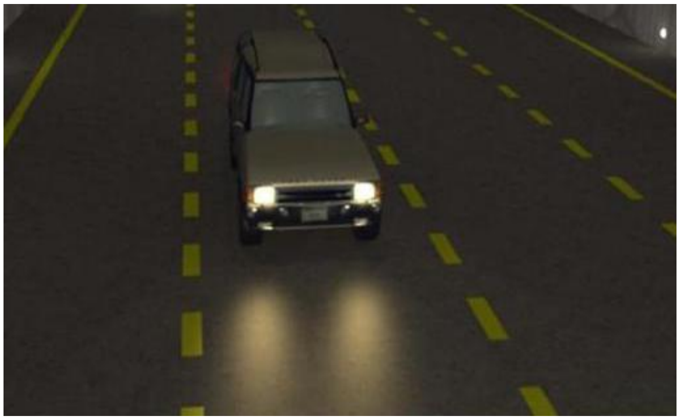


Figure 4. Modeling the car lights using two light types

traffic, and dead zones at the sides. Also markings are added on the road to visualize individual driving lanes. Both markings and lanes are in accordance with road administration guidelines.

4) **Custom traffic events:** Users can create a tunnel with custom traffic events by setting properties for each individual event. As each car gets a speed description in advance, this description will determine the car's behaviour during the animation. This way, any traffic event can be generated. Currently supported traffic events are: regular traffic, cars stopping on specific lane, traffic jam, falling objects (see Figure 5), and ghost driver. More events will be added in the future.

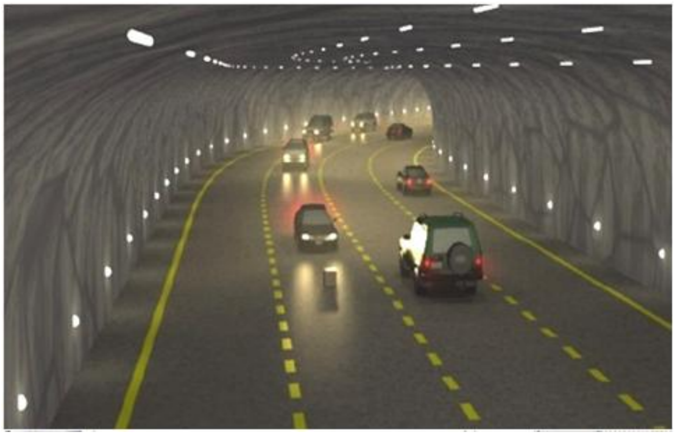


Figure 5. Traffic event in which fallen object hinders regular traffic

5) **Adjustable lighting:** Lighting is another crucial element for the scene generation as it changes perception dramatically. Height, position and amount of light sources can be configured, as well as the color, intensity and range for each light source. Slight variations in color and intensity change the level of perceived realism. This way, physically correct lighting can be achieved in the tunnel. This is important for detection algorithms as changes in intensity (e.g., specular lighting spots) and reflections are means of detecting vehicles and other objects.
6) **Adjustable camera settings:** According to the adjustable lighting, the amount, height and position of cameras can be configured, as well as the viewing angle, focal point and lens.
7) **Modular design:** The simulator is designed in a modular way, with loosely coupled modules so that third parties can create and/or integrate their own modules into the simulator. This way, the Blender render engine can be easily replaced by more performant render engines such as 3ds Max and Maya. Additionally, new scenarios, such as urban settings and highways, can be built and easily integrated into the tunnel simulator.
8) **Material / texture for road and tunnel:** As each material has unique properties and reacts differently to light, the best materials are chosen in order to ensure proper perceived realism. Currently one preset material is used for the road, and another preset material is used for the tunnel. However, due to the modular design, easy integration of new materials and textures is possible, should the need arise.
9) **Preview functionality:** The simulator provides the ability to preview current settings at all times, in order to facilitate fine tuning and check settings before starting the rendering process. A simplified visualization of the generated tunnel and traffic events is herefore constructed and allows real time 3D preview. Once a satisfactory tunnel is constructed, users can save the tunnel settings in an XML-file. Scene and traffic specifications can be saved separately, enabling users to create custom content blocks and linking them.
10) **Performance optimization:** Based on the parameters set in the user interface, a scene is built and the traffic flow is created. The animation is rendered at a chosen quality, resulting in a video file and photorealistic, accurate image suitable for video detection. In order to improve performance, only what can be seen is created. As viewing distance inside tunnels is obstructed by curves, this improves the rendering process by not wasting resources on visualizing invisible objects. Additionally, quality options such as resolution, oversampling, ambient occlusion and motion blur can be turned down to speed up the render process. And finally, a render farm can be used for quicker results. In our implementation, Farmerjoe was used, but thanks to the modular design, this can easily be replaced by another render farm.

IV. Internal design details

The Tunnel Simulator has been implemented and is currently evaluated by Traficon. Below is an overview of the main internal design details.

A. Constructing the 3D tunnel using Blender

Due to the high license costs of 3ds Max and Maya, Blender was chosen to develop the Tunnel Simulator. Blender features an internal render engine capable of features such as ray-tracing, motion blur, oversampling and ambient occlusion, but lacks real global illumination capabilities, apart from using the radiosity solver, which requires enormous amounts of processing power.

Creating a 3D tunnel model using Blender can be done in several ways. One approach is to use basic objects such as triangles and squares, and use them like building blocks by scaling, translating and rotating them in order to create surfaces. Joining multiple surfaces creates objects, called meshes. The process of joining multiple surfaces can be compared to welding two metal sheets together. Another method for constructing a 3D tunnel, and the one we used, is by moving a cross section along a path in order to define the tunnel's shell, similar to extrusion. The first step is creating a guiding path for the cross section, such as a curve. Blender has three main types of curves: Bezier curves, NURBS and paths. Essentially, paths are the same as Bezier curves, but are initially locked to a 2D plane and have a start and ending point, giving them a direction from start to end. Any of these types can be used as a guiding path for the cross section. Figure 6 displays a path, which was used to create the outer shell. The path is constructed by seven control points, called knots. The curve is generated by a Bezier algorithm. All kinds of free form curves can be created by manipulating knots (e.g., translations, insertions and adding weight to knots).

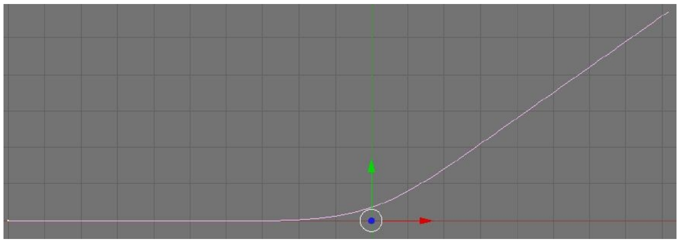


Figure 6. A path created by using Blender

Once the path is created, a cross section needs to be constructed. This cross section will then be extruded along the path in order to create the shell. The cross section is created by using a closed curve such as a Bezier circle or NURBS circle. The actual shape in Figure 7 is generated by four Bezier circles, each one creating a quadrant and joined together. The final step for creating the shell is extrusion, realized by setting the cross section as the curve's bevel shape (see Figure 7). The main advantage of this method is its easy adjustability. By adjusting the knots and handles, the shape, length and curvature of the tunnel can be adjusted.

Finally, a road needs to be constructed and placed inside the shell. This process is similar to creating the shell. A new cross section needs to be constructed. This object is a line with identical width as the tunnel cross section. An identical path is created by duplicating the tunnel path and placing it on top of the tunnel path.

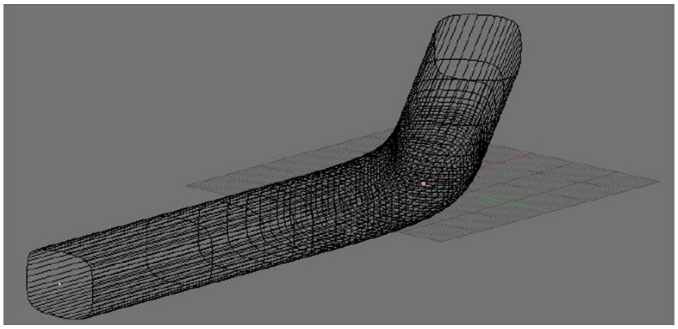


Figure 7. Modeling the 3D tunnel with Blender

Once the tunnel shape is created, materials are assigned to it in order to define its appearance. Each material has unique properties, and reacts differently to light. Therefore, different (fixed) materials are chosen for the tunnel and road to ensure proper perceived realism. As already stated, due to the modular design, easy integration of new materials and textures is possible, should the need arise.

Changing the tunnel's width and cross section shape cannot be directly controlled by users through the graphical user interface in order to prevent mismatches between cross section diameter and road settings. The user is only allowed to change road settings such as number of lanes, driving lane width and dead zone space. Adjusting these settings will automatically cause the tunnel cross section to scale to a suitable size.

B. Creating the vehicle database

The 3D cars and SUVs used by the Tunnel Simulator are provided by Marlin Studios. Each vehicle is constructed by approximately 2100 to 4000 polygons and is made up by four groups of sub objects and a high resolution texture: glass windows, body, interior and wheels. The high resolution texture is improved by implementing normal mapping and ambient occlusion baking. Most of the car's interior was removed since this is not visible when rendering.

As no (free) models of trucks with good quality and level of detail for Blender could be found, we built the truck model ourselves. The model uses 3789 polygons and texture mapping to provide extra detail. Figure 8 presents the three different trucks currently supported in the Tunnel Simulator.

C. Creating the lighting setup

Now that the scene and vehicle database is created, the lighting setup needs to be constructed to mimic lighting conditions inside a real life tunnel. Adding lights to a closed scene can change perception dramatically.

There are five different kinds of lights available in Blender: a normal lamp, sun, spot, hemi and area light. Testing indicated that spots produced the most realistic lighting

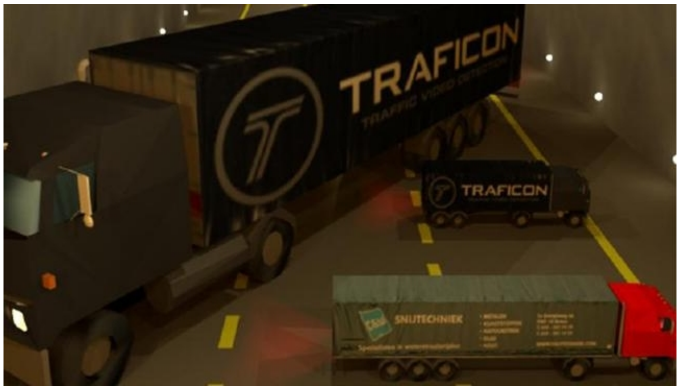


Figure 8. Blender truck models

effect inside the tunnel, followed closely by regular lamps. However, regular lights considerably reduce rendering time, making these lamps the best choice for the scene generator.

Finding the correct lighting setup was done by studying real videos taken inside a tunnel and analyzing intensity, color and positioning, and trying out those settings inside the virtual environment. Additionally, the lighting setup must be flexible in order to correspond with tunnel customization. In Blender this problem can be solved by parenting the lights to an identical Bezier curve that defines the tunnel's path. Next, the lights need to placed at a regular interval along the tunnel, creating a specific pattern. This process can be accelerated by using Blender's DupliFrame function.

DupliFrames stands for Duplication at Frames and is a modeling technique for objects which are repeated along a path, such as the wooden sleepers in a railroad, the boards in a fence or the links in a chain and thus also for creating the lighting setup along the tunnel's path. Figure 9 displays the results from the DupliFrame process. In this case, 22 lights are equally placed along the tunnel's path. This process is also used to position lamp placeholders along the ceiling and sides of the tunnel and to place markings on the road.

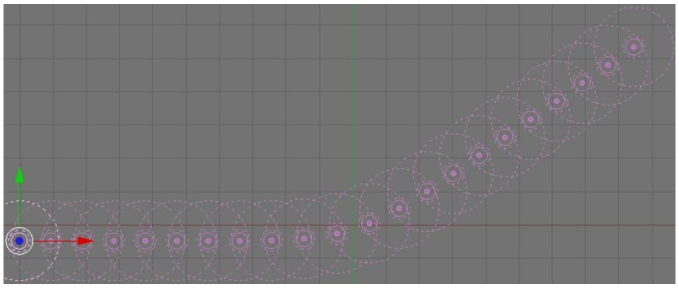


Figure 9. DupliFramed lights are equally placed along the tunnel's path

D. Adding a camera to the scene and generating traffic

The next step for the scene generator is defining a view by placing a camera into the scene. Rendered images are created from the camera's point of view, similar to shooting a real life video. At this point the entire 3D tunnel has been built, a lighting setup has been added, and the camera has been placed inside the tunnel.

The final step is generating the traffic, linking the traffic to the tunnel by copying the tunnel's path, and setting up individual animations for each vehicle.

The real time event character of game engines, where one can intervene and influence the scene in real time, is not possible using Blender to slow down or speed up a car. Neither is it possible to detect speed changes of cars in order to slow down or speed up other vehicles too. As Blender can only render predefined animations, each car needs to have a given speed description.

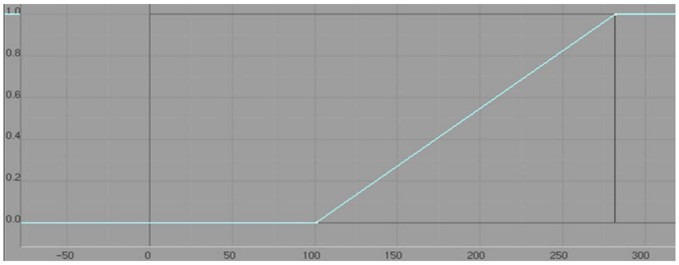


Figure 10. Exemplary speed description for a car

Once the animation is created and starts, nothing can be changed. Whenever the user changes some settings, the animation needs to be built again. Each animation consists of three parts: (i) the object that is animated, in this case the vehicle, (ii) the path to follow, i.e., an offset to the middle of the tunnel in order to choose a lane, and (iii) the speed description using interpolation curves. Note that a small, random deviation is added to the offset so that not every car drives exactly in the middle of the lane. Figure 10 presents an exemplary speed description for a car. The speed of a car can be set by tuning the frame rate. This frame rate is divided by the requested speed (in m/s), and multiplied with the total length of the visible route section (in m). This way, the number of frames for the entire route is retrieved. As presented in Figure 10, from the perspective of one specific camera, the car starts to drive at frame 100. About 180 frames later, the car arrives at the end of the visible route section for that camera. As the speed follows a straight line from 0 to 1, the position of the car will change linearly in time. As a result, the car will drive with constant speed from start to end. The speed description uses Bezier curves, composed out of Bezier triples (two handlers, one knot), linked by a smooth line.

Normal traffic occurs when no special events, such as traffic jams or collisions, occur during the traffic flow. In this case, the user can set four parameters: speed, flow rate (i.e., cars per minute), statistical distribution of vehicle classes, uni- or bidirectional traffic. These four parameters are used to simulate the desired traffic by repeating the animation for a random car and placing it in the scene. A Gaussian distribution is used to average the distance between the cars.

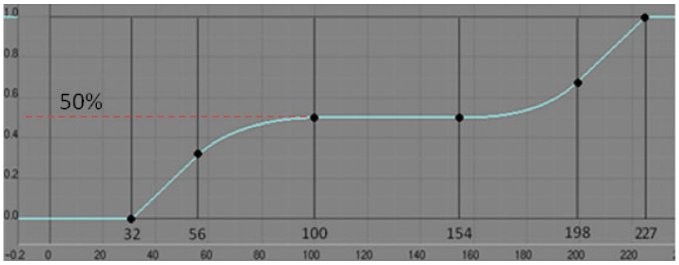
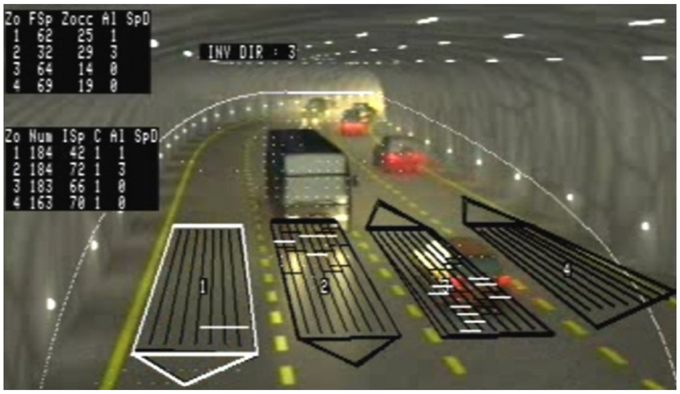


Figure 11. Speed description for a car that halfway stops and then speeds up again



```
Zo FSp Zocc Al SpD
 1  62   25  1
 2  32   29  3
 3  64   14  0
 4  69   19  0

Zo Num ISp C Al SpD
 1 184  42 1  1
 2 184  72 1  3
 3 183  66 1  0
 4 163  70 1  0
```

```
INV DIR : 3
```

Figure 12. Detection testing using video generated by Tunnel Simulator

Also special traffic events can be implemented. Figure 11 presents the speed description for a car that stops halfway.

The car after a stationary car should also stop to avoid a collision. Therefore, a custom speed description for these cars need to be provided as well by slightly shifting the new speed curve to the right and lowering this curve to make the car stop after this first car. A for loop is used to repeat this and let more than one car stop after the stationary car.

In order to make the stops more photorealistic, both back lights and flashes are added to the cars, lighting up when the car brakes, respectively warning when standing still. Both features are implemented using layer descriptions.

The same way, also traffic jams, falling objects, and ghost driver events are implemented in the Tunnel Simulator.

V. Evaluation

The Tunnel Simulator fully works according to the specifications. Users are able to create a small tunnel segment (a single driving lane) to a large (four driving lanes) tunnel segment. Evaluations at Traficon indicated that all scene generator properties are intuitive for most of the users. Users were able to create the desired tunnel with custom traffic events, which ultimately resulted in a movie and OVF file. Detection testing at Traficon (see Figure 12) indicated that the generated movies are well suited for their intended purpose, with high correlation between both OVF files. Video footage and a demonstration of the finished project can be found on [11].

VI. Conclusion

The developed Tunnel Simulator enables users to create a tunnel with custom traffic events by setting properties for the tunnel and each individual event. As a result, a scene is generated with the specified tunnel, traffic events and ground truth. The scene can then be rendered in order to use it for testing video dectection algorithms. This way, the quality and reliability of new video detection algorithms can be extensively tested without the need to create and record dangerous traffic situations to have video source material.

Future work includes the development of new scenes (e.g., intersections), vehicle tracking, the addition of fire and smoke events (e.g., burning vehicle), and pedestrian traffic.

Acknowledgment

The authors would like to thank Frédérique Vanneste, Karel Azou and Kristof Hennebel for their valuable work.

References

[1] M.P. Müller, Tunnel Safety: where are we now, Swiss Re, Risk Engineering Services, 2005.

[2] G. Wang, D. Xiao, and J. Gu, Review on vehicle detection based on video for traffic surveillance, Proc. of IEEE International Conference on Automation and Logistics, pp. 2961-2966, 2008.

[3] A. Bevilacqua and S. Vaccari, Real time detection of stopped vehicles in traffic scenes, Proc. of IEEE Conference on Advanced Video and Signal Based Surveillance, pp. 266-270, 2007.

[4] O. Akoz and M.E. Karsligil, Video-based traffic accident analysis at intersections using partial vehicle trajectories, Proc. of 17th IEEE International Conference on Image Processing, pp. 4693-4696, 2010.

[5] N. Farhi, M. Goursat, and J.P. Quadrat, The traffic phases of road networks, Transportation Research Part C, 19(1), pp. 85-102, 2011.

[6] J. Miller and E. Horowitz, FreeSim - a free real-time freeway traffic simulator, Proc. of IEEE Intelligent Transportation Systems Conference, pp. 18-23, 2007.

[7] T. Ishikawa, Development of a road traffic simulator, IEEE Vehicular Technology Society, 47(3), pp. 1066-1071, 1998.

[8] T. Winters, M. Johnson, and V. Paruchuri, LITS: Lightweight Intelligent Traffic Simulator, Proc. of International Conference on Network-Based Information Systems, pp. 386-390, 2009.

[9] L. Yong and C.Y. Li, A microscopic simulator for urban traffic systems, Proc. of 5th IEEE International Conference on Intelligent Transportation System, pp. 622- 626, 2002.

[10] P. Wessel and J. Rossberg, Increase traffic safety by increasing traffic management skills using advanced training simulator, Proc. of 4th International Conference on Traffic and Safety in Road Tunnels, pp. 1-10, 2007.

[11] Tunnel Simulator demonstration, http://elit.howest.be/demos.

SUMO – Simulation of Urban MObility

An Overview

Michael Behrisch, Laura Bieker, Jakob Erdmann, Daniel Krajzewicz

Institute of Transportation Systems
German Aerospace Center
Rutherfordstr. 2, 12489 Berlin, Germany
e-mail: Michael.Behrisch@dlr.de, Laura.Bieker@dlr.de, Jakob.Erdmann@dlr.de, Daniel.Krajzewicz@dlr.de

***Abstract*— SUMO is an open source traffic simulation package including net import and demand modeling components. We describe the current state of the package as well as future developments and extensions. SUMO helps to investigate several research topics e.g. route choice and traffic light algorithm or simulating vehicular communication. Therefore the framework is used in different projects to simulate automatic driving or traffic management strategies.**

Keywords- microscopic traffic simulation, software, open source

I. Introduction

The German Aerospace Center (DLR) started the development of the open source traffic simulation package SUMO back in 2001. Since then SUMO has evolved into a full featured suite of traffic modeling utilities including a road network capable to read different source formats, demand generation and routing utilities from various input sources (origin destination matrices, traffic counts, etc.), a high performance simulation usable for single junctions as well as whole cities including a “remote control” interface (TraCI) to adapt the simulation online.

In this paper, we will survey some of the recent developments and future prospects of SUMO including multimodal simulation and emission modeling. We start with an overview of recent projects, then give a more detailed description of the two topics mentioned before and finish with an outlook on the extensibility especially regarding simulation models and online interaction.

II. The SUMO Suite

SUMO was started to be implemented in 2001, with a first open source release in 2002. There are two reasons for making the work available as open source. The first is the wish to support the traffic simulation community with a free tool into which own algorithms can be implemented. While there are some open source traffic simulations available, most of them have been implemented within a student thesis and got unsupported afterwards. A major drawback – besides reinventing the wheel – is the almost non-existing comparability of the implemented models or algorithms. A common simulation platform should be of benefit here. The second reason for making the simulation open source was the wish to gain support from other institutions.

SUMO is not only a traffic simulation, but rather a suite of applications which help to prepare and to perform the simulation of traffic. As the traffic simulation “sumo” requires the representation of road networks and traffic demand to simulate in an own format, both have to be imported or generated using different sources.

SUMO road networks can be either generated using an application named “*netgen*” or generated by importing a digital road map. The road network importer “*netconvert*” allows to read networks from other traffic simulators as VISUM, Vissim, or MATsim. It also reads other common formats, as shapefiles or Open Street Map. Due to the lack of applications, the support for TIGER networks was dropped. But TIGER networks are also available as shapefiles and were included in the OSM data base. Besides these formats, “netconvert” is also capable to read less known formats, as RoboCup network format, or openDRIVE.

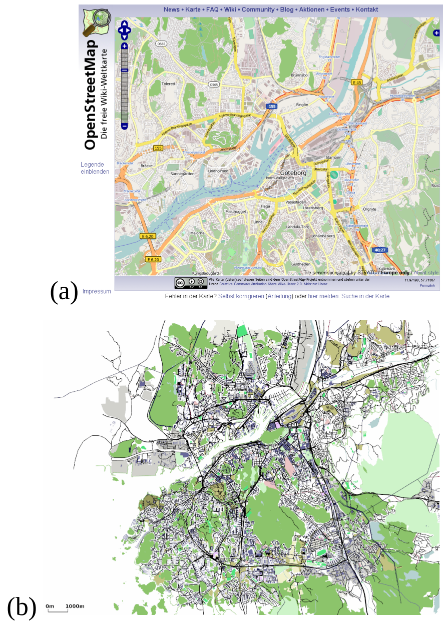


Figure 1. Example conversion of an OpenStreetMap; a) original OpenStreetMap view, b) network imported into SUMO.

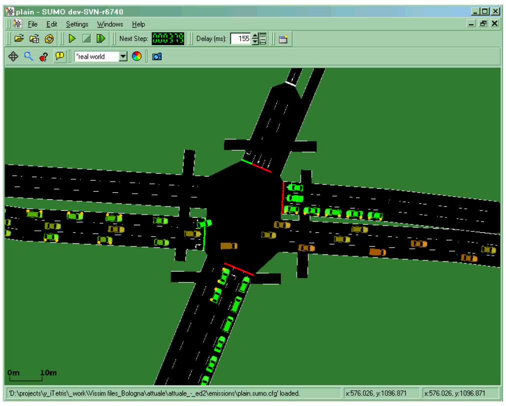
SUMO is a purely microscopic traffic simulation. Each vehicle is given explicitly, defined at least by an identifier (name), the departure time, and the vehicle's route through the network. If wanted, each vehicle can be described more detailed. The departure and arrival properties, such as the lane to use, the velocity, or the position can be defined. Each vehicle can get a type assigned which describes the vehicle's physical properties and the variables of the used movement model. Each vehicle can also be assigned to one of the available pollutant or noise emission classes. Additional variables allow the definition of the vehicle's appearance within the simulation's graphical user interface.

The definitions of vehicles can be generated using different sources. For large-scale scenarios usually so-called "origin/destination matrices" (O/D matrices) are used. They describe the movement between traffic assignment zones in vehicle number per time. Often, a single matrix is given for a single day what is insufficient for microscopic traffic simulations as the direction changes over time are not represented. Sometimes, matrices with a scale of 1h are available. For large-scale traffic simulations, they are the most appropriate source. The SUMO suite includes "*od2trips*", an application for converting O/D matrices to single vehicle trips. Besides disaggregating the matrix, the application also assigns an edge from the road network as depart/arrival position. The map from traffic assignment zones into edges is to be given to application as one input.

The resulting trips consist of a start and end road together with a departure time but no explicit route information. Routes are usually calculated by performing a traffic assignment employing a routing procedure such as shortest path calculations under different cost functions. Details on the models in SUMO can be found in section III.B.

A further route computation application, "*jtrrouter*", uses definitions of turn percentages at intersection for computing routes through the network. Such an approach can be used to set up the demand within a part of a city's road network consisting of up to ten nodes. A further application, "*dfrouter*", computes routes by using information from loop detectors. This approach is quite successful when applied to highway scenarios where the road network does not contain rings and the highway entries and exits are completely covered by detectors. It fails on inner-city networks with rings and if the coverage with induction loops is low.

The simulation is time-discrete with a default simulation step length of 1s. It is space-continuous and internally, each vehicle's position is described by the lane the vehicle is on and the distance from the beginning of this lane. When moving through the network, each vehicle's speed is computed using a so-called car-following model. SUMO uses an extension of the car-following model developed by Stefan Krauß [1]. Changing the lane is done using a model developed during the implementation of SUMO [12]. Two versions of the traffic simulation exist. The first is a pure command line application for efficient batch simulation. The second version is a graphical application which renders the performed simulation using openGL.



'D:\projects\y_iTetris_work\Vissim files_Bologna\attuale\attuale_-_ed2\emissions\plain.sumo.cfg' loaded.

Figure 2. Screenshot of the graphical user interface coloring vehicles by their CO_2 emission.

SUMO allows to generate various outputs for each simulation run. These range from simulated induction loops to single vehicle positions written in each time steps for all vehicles and up to complex values as information about each vehicle's trip or aggregated measures along a street or lane. Besides conventional traffic measures, SUMO was extended by a noise emission and a pollutant emission / fuel consumption model, see also section V.A.

In 2006 the simulation was extended by the possibility to interact with an external application via a socket connection. This API was implemented by Axel Wegener and his colleagues from the University of Lübeck, and was made available as a part of SUMO's official release. Within the iTETRIS project, see section IV.A, this API was reworked, integrating it closer to SUMO's architecture and specification.

TraCI is not the only contribution to SUMO from other parties. SUMO Traffic Modeler [13] allows to define a population for a given area and compute this population's mobility wishes which can be used as an input for the traffic simulation. The same is done by "*activitygen*" written by Piotr Woznica and Walter Bamberger from TU Munich. eWorld [14] allows to set up further environmental characteristics, such as weather condition and visualizes a running, connected simulation.

III. RESEARCH TOPICS

A. *Vehicular Communication*

The probably most popular application for the SUMO suite is modeling traffic within research on V2X – vehicle-to-vehicle and vehicle-to-infrastructure – communication. Here, usually SUMO is coupled to an external communication simulation, such as ns2 or ns3 [2] using TraCI. For obtaining a functioning environment for the simulation of vehicular communications, an applications instance which models the V2X application to simulate is needed. Additionally, a synchronization and message exchange mechanism has to be involved.

TraNS [15] was a very popular middleware for V2X simulation which realizes these needs. It was build upon

SUMO and ns2. Here, the TraNS extensions to ns2 were responsible for synchronizing the simulators and the application had also to be modeled within ns2. TraNS was the major reason for making TraCI open source. With the end of the projects the original TraNS authors were working on, TraNS itself got no longer maintained and works with a very old SUMO version only, as the TraCI API was changed.

A modern replacement for TraNS was implemented within the iTETRIS project [7]. The iTETRIS system couples SUMO and ns2's successor ns3. ns3 was chosen, as ns2 was found to be unstable when working with a large number of vehicles. Within the iTETRIS system, the "iTETRIS Control System", an application written in c++ is responsible for starting and synchronizing simulators. The V2X applications are modeled in own, language-agnostic programs. This clear distribution of responsibilities allows to implement own applications conveniently in ones favorite programming language.

A very flexible approach for coupling SUMO with other applications is the VSimRTI middleware developed by FhG Fokus [16]. Its HLA-based architecture does not only allow the interaction between SUMO and other communication simulators. It is also able to connect SUMO and Vissim, a commercial traffic simulation package. In [16], a system is described where SUMO was used to model large-scale areas coarsely, while Vissim was used for a fine-grained simulation of traffic intersections.

Besides the named possibilities to simulate vehicular applications, other implementations allow to use SUMO in combination with other communication simulators such as ns2 or ns3.

Many vehicular communication applications target at increasing safety. It should be stated, that up to now, microscopic traffic flow models are not capable to compute safety-related measures. SUMO's strength lies in simulation of V2X applications which aim at the improvement of traffic efficiency. Also, evaluating concepts for forwarding messages to their defined destination ("message routing") can be done using SUMO, see for example [17] or [18].

B. Route Choice and dynamic Navigation

The assignment of proper routes to a complete demand or a subset of vehicles is investigated both, on a theoretical base and as new applications. On the theoretical level, the interest lies in a proper modeling of how traffic participants choose a route – a path through the given road network – to their desired destination. As the duration to pass an edge of the road graph highly depends on the numbers of participants using this edge, the computation of routes through the network under load is a crucial step in preparing large-scale traffic simulations. Due to its fast execution speed, SUMO allows to investigate algorithms for this "user assignment" or "traffic assignment" on a microscopic base. Usually, such algorithms are investigated using macroscopic traffic flow models, or even using coarser road capacity models which do not resemble dissolving road congestions.

The SUMO suite supports such investigations using the "*duarouter*" application. By now, two algorithms for computing a user assignment are implemented, c-logit and Gawron's dynamic user assignment algorithm. Both are iterative and due to this time consuming. Possibilities to reduce the duration to compute an assignment were evaluated and are reported in [19]. A further possibility to reduce the computational effort is given in [20]. Here, vehicles are routed only once, directly by the simulation and the route choice is done based on a continuous adaptation of the edge weights during the simulation.

Practical applications for route choice mechanisms arise with the increasing intelligence of navigation systems. Navigation systems as Tom Tom's IQ routes ([21]) use on-line traffic information to support the user with a fastest route through the network regarding the current situation on the streets. A set of research is done on finding new ways of determining the state on the road network, where vehicular communication is one possibility. With the increased penetration rate of vehicles equipped with a navigation device, a further question arises: what happens if all vehicles get the same information? Will they all use the same route and generate new congestions? This question is not only relevant for drivers, but also for local authorities as navigation devices may invalidate concepts for keeping certain areas calm by routing vehicles through these areas. SUMO allows to address these topics, see, i.e., [9].

C. Traffic Light Algorithms

The evaluation of developed traffic light programs or algorithms for making traffic lights adaptable to the current traffic is one of the main applications for microscopic traffic flow simulations. As SUMO's network model is relatively coarse compared to commercial applications as Vissim, SUMO is usually not used by traffic engineers for evaluating real-life intersections. Still, SUMO's fast execution time and its open TraCI API for interaction with external applications make it a good candidate for evaluating new traffic control algorithms, both for controlling a single intersection ([22]) and for net-wide investigations. By distinguishing different vehicle types, SUMO also allows the simulation of public transport or emergency vehicle prioritization at intersections [5].

The first investigations were performed by implementing the traffic light algorithms to evaluate directly into the simulation's core. Over the years, this has showed to be hard to maintain. Using TraCI seems to be a more sustainable procedure currently.

D. Evaluation of Surveillance Systems

SUMO's capability to simulate large-scale scenarios allows the evaluation of new traffic surveillance systems. Within the VABENE project, a running traffic simulation was calibrated using conventional induction loop measures, and using vehicle densities and average velocities obtained from an airborne camera system which was mounted under a zeppelin. The taken pictures were processed on board of the zeppelin and the system sent the positions of vehicles to the ground center. Here, the number of vehicles running through the simulation was matched to the number of vehicles running on the observed street in reality.

Within the TrafficOnline project, a system for travel speed observation using GSM data was designed, implemented, and evaluated. SUMO's responsibility was to generate a virtual telephony behavior. For testing the travel speed recognition system, not only road traffic on both highways and within urban areas was modeled. Additionally, busses, light rail and fast rail were modeled to evaluate whether the system is able to detect the speeds on the roads even if additional moving participants exist.

IV. Recent Projects

SUMO was and is used and extended in several research projects. In the following, only some of the recent ones are named.

A. *iTetris*

The interest in V2X communication is increasing but it is expensive and may be even dangerous to implement such a system. For research studies which measure the benefits of a system before it is deployed into the real world, a simulation framework which simulates the interaction between vehicles and infrastructure of whole cities is needed. The aim of iTETRIS project was to develop such framework and to couple the communication simulator ns3 and SUMO using an open source system called "iCS" – iTETRIS Control System which was developed within the project. The iCS is responsible for starting the named simulators and additional programs which simulate the V2X applications. It is also responsible for synchronizing the participating simulators, and for the message exchange. Using this simulation framework it was possible to investigate the impacts on V2X communication strategies.

Within the project several traffic management strategies were simulated e.g. prioritization of emergency vehicles at controlled intersections [5] and rerouting vehicles over bus lanes using V2X communication [6].

B. *VABENE*

Traffic is more and more important for large cities. Big events or even catastrophes might cause traffic jams and problems to the transport systems and might even be dangerous for the people who live in the city. Public authorities are responsible to take the according action to prevent the worst case. The objective of VABENE is to implement a system which supports the public authority to decide which action should be taken.

The focus in this project lies on simulating the traffic of large cities. The system shows the current traffic state of the whole traffic network which helps the traffic manager to realize when a critical traffic state will be reached. To simulate the traffic of a large region like Munich and the area around Munich a mesoscopic traffic model was implemented into SUMO which is available for internal proposes only.

The simulation is restarted every 10 minutes, loads a previously saved state of the road network and computes the state for half an hour ahead. While running, the simulation state is calibrated using induction loop measures and measures collected from an airborne traffic surveillance system. Both, the current traffic state as well as the prediction of the future state is presented to the authorities. This system is the successor of demonstrators used during the pope's visit in Germany in 2005 and during the FIFA World Cup in 2006.

C. *CityMobil*

Microscopic traffic simulations also allow the evaluation of large scale effects of changes in vehicle or driver behavior such as the introduction of automated vehicles or electromobility. The former was examined with the help of SUMO in the EU project CityMobil where different scenarios of (partly) automated cars or personal rapid transit were set up on different scales, from a parking area up to whole cities.

On a small scale, the benefits of an autonomous bus system were evaluated. In this scenario, busses are informed about waiting passengers and adapt their routes to this demand. On a large scale, the influence of platooning vehicles was investigated, using the model of a middle-sized city of 100.000 inhabitants. Both simulations showed positive effects of the transport automation.

V. Recent Extensions

A. *Emission and Noise Modeling*

Within the iTETRIS project, SUMO was extended by a model for noise emission and a model for pollutant emission and fuel consumption. This was required within the project for evaluating the ecological influences of the developed V2X applications.

Both models are based on existing descriptions. 7 models for noise emission and 15 pollutant emission / fuel consumption models were evaluated, first. The parameter they need and their output were put against values available within the simulation and against the wanted output, respectively. Finally, HARMONOISE [23] was chosen as noise emission model. Pollutant emission and fuel consumption is implemented using a continuous model derived from values stored in the HBEFA database [24].

The pollutant emission model's implementation within SUMO allows to collect the emissions and fuel consumption of a vehicle over the vehicle's complete ride and to write this values into a file. It is also possible to write collected emissions for lanes or edges for defined, variable aggregation time intervals. The only available noise output collects the noise emitted on lanes or edges within pre-defined time intervals. A per-vehicle noise collecting output is not available. Additionally, it is possible to retrieve the noise, emitted pollutants, and fuel consumption of a vehicle in each time step via TraCI. Also, collected emissions, consumption, and noise level for a lane or a road can be retrieved.

Besides measuring the level of emissions or noise for certain scenarios, the emission computation was also used for investigating new concepts of vehicle routing and dependencies between the traffic light signal plans and emissions [25].

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-169-4

B. Person-based Intermodal Traffic Simulation

A rising relevance of intermodal traffic can be expected due to ongoing urbanization and increasing environmental concerns. To accommodate this trend SUMO was extended by capabilities for simulating intermodal traffic. We give a brief account of the newly added concepts and report on our experience with person-based intermodal simulation.

The conceptual center of intermodal-traffic is the individual person. This person needs to undertake a series of trips each of which may be taken with a different mode of transport such as personal car, public bus or walking. Trips may include traffic related delays, such as waiting in a jam, waiting for a bus or waiting to pick up an additional passenger. While all trips may be simulated independently it is important to note that earlier delays influence later trips of the person. The above concept is reflected in an extension of the SUMO route input. One can now specify a person as list of rides, stops and walks. A ride can stand for any vehicular transportation, both private and public. It is specified by giving a starting edge, ending edge and a specification of the allowed vehicles. Stops correspond to non-traffic related activities such as working or shopping. A walk models a trip taken by foot but it can also stand for other modes of transport which do not interfere with road traffic. Another extension concerns the vehicles. In addition to their route, a list of stops and a line attribute can be assigned. Each stop has a position, and a trigger which may be either a fixed time, a waiting time or the id of a person for which the vehicle must wait. The line attribute can be used to group multiple vehicles as a public transport route.

These few extensions are sufficient to express the above mentioned person trips. They are being used within the TAPAS [10][11] project to simulate intermodal traffic for the city of Berlin. Preliminary benchmarks have shown that the simulation performance is hardly affected by the overhead of managing persons.

In the future we would like to address the following issues:

- Online rerouting of persons. At the moment routing across trips must be undertaken before the start of the simulation. It is therefore not possible to compensate a missed bus by walking instead of waiting for the next bus.
- Visualization of persons.
- Smart integration of bicycles. Depending on road infrastructure bicycle traffic may or may not interact with road traffic.

Import modules for importing public time tables.

VI. Current Development

A. Car-Following and Lane-Change API

Within the iTETRIS project, first steps towards using other models than the used Krauß extension for computing the vehicles' longitudinal movement were taken. An API for implementing and embedding other car-following models was implemented. Some initial implementations of other models exist, though not all of them are able to deal correctly with multi-lane urban traffic. The work is assumed to continue, especially as the decision was taken to concentrate on extending the default model instead of sticking to a well-defined scientific model. What is already possible to do with car-following models is also meant to be implemented for lane-change models.

B. Model Improvements

One of the initial tasks SUMO was developed for was the comparison of traffic flow models, mainly microscopic car-following and lane-changing models. This wish requires a clean implementation of the models to evaluate. On the other hand, most models are concentrating to describe a certain behavior, e.g. spontaneous jams, making them inappropriate to be used within complex scenarios which contain a large variety of situations.

As a recent conclusion, next steps of SUMO development will go beyond established car-following models. Instead, an own model will be developed, aiming on its variability mainly. In a first step, the internal representation of road networks will be revalidated and cleaned. Then, the work will aim on coupling the car-following and the lane-changing models closer.

C. Interoperability

SUMO is not the only available open source traffic simulation platform. Some other simulations, such as MATsim [26], offer their own set of tools for demand generation, traffic assignment etc. It is planned to make these tools being usable in combination with SUMO by increasing the capabilities to exchange data. Besides connecting with other traffic simulation packages, SUMO is extended for being capable to interact with driving or world simulators. Within the DLR project "SimWorld Urban", SUMO is connected to the DLR driver simulator, allowing to perform simulator test drives through a full-sized and populated city area.

D. Network Editor

Since 2011, a graphical network editor is implemented. It allows to set up a complete road network for SUMO, including all needed information, such as correct lane number, speed limits, connections across intersections, and traffic lights. For now, this tool is not part of the open source package, but is held for internal purposes only.

VII. Summary

We have presented a coarse overview of the microscopic traffic simulation package SUMO, presenting the included application along with some common use cases, and the next steps within the development. We kindly invite the reader to participate in the development. Further information can be obtained via the project's web site [8].

REFERENCES

[1] S. Krauß. “Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics”. PhD thesis, 1998.

[2] ns3 Homepage [Online]. Available: http://www.nsnam.org/, accessed January 26, 2011.

[3] PTV Homepage. [Online] “Vissim”. Available: http://www.ptv.de/software/verkehrsplanung-verkehrstechnik/software-und-system-solutions/vissim/ accessed January 27, 2011.

[4] L. Bieker et. al. “Derivation of a fast, approximating 802.11p simulation model”. Intelligent Transport Systems Telecommunications (ITST2010), November 9-11, 2010, Kyoto, Japan.

[5] L. Bieker, “Emergency Vehicle prioritization using Vehicle-to-Infrastructure Communication”, Young Researchers Seminar 2011 (YRS2011), June 8-11, 2011, Copenhagen, Denmark.

[6] L. Bieker and D. Krajzewicz, “Evaluation of opening Bus Lanes for private Traffic triggered via V2X Communication”, (FISTS 2011), June 29- July 1, 2011, Vienna, Austria.

[7] iTETRIS Homepage [Online]. Available: http://www.ict-itetris.eu/10-10-10-community/ accessed January 26, 2011.

[8] D. Krajzewicz and M. Behrisch, L. Bieker, J. Erdmann, SUMO homepage. [Online]. Available: http://sumo.sourceforge.net/, accessed January 26, 2011.

[9] D. Krajzewicz, D. Teta Boyom, and P. Wagner, “Evaluation of the Performance of city-wide, autonomous Route Choice based on Vehicle-to-vehicle-Communictaion”. TRB 2008 (87. Annual Meeting), January 13-17, 2008, Washington DC, USA.

[10] R. Cyganski and A. Justen. “Maßnahmensensitive Nachfragemodellierung in mikroskopischen Personenverkehrsmodellen“. Deutsche Verkehrswissenschaftliche Gesellschaft, Schriftenreihe B, 2007.

[11] G. Hertkorn and P. Wagner. “Travel demand modelling based on time use data”. In: 10th International conference on Travel Behaviour Research, August 2004.

[12] D. Krajzewicz. “Traffic Simulation with SUMO - Simulation of Urban Mobility”. In: Fundamentals of Traffic Simulation International Series in Operations Research and Management Science. Springer. Seiten 269-294. ISBN 978-1-4419-6141-9. ISSN 0884-8289, 2010.

[13] L. G. Papaleondiou and M. D. Dikaiakos. “TrafficModeler: A Graphical Tool for Programming Microscopic Traffic Simulators through High-Level Abstractions”. In: Proceedings of the 69th IEEE Vehicular Technology Conference, VTC Spring 2009, 26-29 April 2009, Hilton Diagonal Mar, Barcelona, Spain 2009.

[14] eWorld homepage [Online]. Available: http://eworld.sourceforge.net/, accessed June 5, 2011.

[15] M. Piorkowski, M. Raya, A. Lugo, P. Papadimitratos, M. Grossglauser, and J.-P. Hubaux, “TraNS: Realistic Joint Traffic and Network Simulator for VANETs”, ACM SIGMOBILE Mobile Computing and Communications Review, pp. 31-33, 2008.

[16] D. Rieck, B. Schuenemann, I. Radusch, C. Meinel. „Efficient Traffic Simulator Coupling in a Distributed V2X Simulation Environment”. SIMUTools '10: Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, Torremolinos, Malaga, Spain, 2010. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, pp. 1-9, ISBN: 978-963-9799-87-5

[17] D. Borsetti, J. Gozalvez. “Infrastructure-Assisted Geo-Routing for Cooperative Vehicular Networks”. Proceedings of the 2nd IEEE (*) Vehicular Networking Conference (VNC 2010), 13-15 December 2010, New Jersey (USA).

[18] M. A. Leal, M. Röckl, B. Kloiber, F. de Ponte Müller and T. Strang. “Information-Centric Opportunistic Data Dissemination in Vehicular Ad Hoc Networks”. International IEEE Conference on Intelligent Transportation Systems (ITSC), 19-22 September 2010, Madeira Island (Portugal).

[19] M. Behrisch, D. Krajzewicz, and Y.-P. Wang. „Comparing performance and quality of traffic assignment techniques for microscopic road traffic simulations”. In: Proceedings of DTA2008. DTA2008 International Symposium on Dynamic Traffic Assignment, 2008-06-18 - 2008-06-20, Leuven (Belgien), 2008.

[20] M. Behrisch, D. Krajzewicz, P. Wagner, and Y.-P. Wang. “Comparison of Methods for Increasing the Performance of a DUA Computation”. In: Proceedings of DTA2008. DTA2008 International Symposium on Dynamic Traffic Assignment, 2008-06-18 - 2008-06-20, Leuven (Belgien), 2008.

[21] R.-P. Schäfer. “IQ routes and HD traffic: technology insights about tomtom's time-dynamic navigation concept”. In Proceedings of the the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering (ESEC/FSE '09). ACM, New York, NY, USA, 171-172. DOI=10.1145/1595696.1595698 http://doi.acm.org/10.1145/1595696.1595698 2009.

[22] D. Krajzewicz, E. Brockfeld, J. Mikat, J. Ringel, C. Rössel, W. Tuchscheerer, P. Wagner, and R. Woesler. “Simulation of modern Traffic Lights Control Systems using the open source Traffic Simulation SUMO”. In: Proceedings of the 3rd Industrial Simulation Conference 2005, Seiten 299-302. EUROSIS-ETI. 3rd Industrial Simulation Conference 2005, 2005-06-09 - 2005-06-11, Berlin (Germany). ISBN 90-77381-18-X.2005.

[23] R. Nota, R. Barelds, and D. van Maercke. “Harmonoise WP 3 Engineering method for road traffic and railway noise after validation and fine-tuning”. Technical Report Deliverable 18, HARMONOISE, 2005.

[24] INFRAS. HBEFA web site, 2011.

[25] D. Krajzewicz, L. Bieker, E. Brockfeld, R. Nippold, and J. Ringel. “Ökologische Einflüsse ausgewählter Verkehrsmanagementansätze“. Heureka '11, 13.-15. März 2011, Stuttgart, Germany.

[26] M. Balmer, M. Rieser, K. Meister, D. Charypar, N. Lefebvre, K. Nagel, K.W. Axhausen “MATSim-T: Architecture and Simulation Times”. In A. L. C. Bazzan and F. Klügl (eds.) Multi-Agent Systems for Traffic and Transportation Engineering, 57–78, Information Science Reference, Hershey, 2009.

Mechanisms Controlling the Sensitivity of Amperometric Biosensors in the Case of Substrate and Product Inhibition

Dainius Simelevicius and Romas Baronas
Faculty of Mathematics and Informatics
Vilnius University
Naugarduko 24, LT-03225 Vilnius, Lithuania
$\{$*dainius.simelevicius, romas.baronas*$\}$*@mif.vu.lt*

***Abstract*—Special case of amperometric biosensors is investigated in this paper. Processes of substrate as well as product inhibition take place during the operation of these biosensors. The operation of biosensors is modelled by employing non-stationary reaction-diffusion equations containing a non-linear term related to non-Michaelis-Menten kinetics. The equation system is solved numerically using finite difference technique. Apparent Michaelis constant is chosen as a good indicator of biosensor reliability. Its dependency on the substrate and product inhibition as well as the diffusion modulus and the Biot number was investigated.**

Keywords-modelling; simulation; apparent Michaelis constant; biosensor; inhibition.

I. Introduction

A biosensor is a device designed to measure concentration of some specific substance in a solution. Biosensors incorporate some biological material, usually an enzyme, thus its name. Enzymes are organic catalysts which catalyze very specific chemical reactions and do not infuence or participate in other reactions. This feature of enzymes is employed in biosensors for the recognition of particular chemicals in solutions [1]–[3]. Amperometric biosensors measure changes in the output current on the working electrode that occur due to the direct oxidation or reduction of products of the biochemical reaction. The output current is usually proportional to the concentration of an analyte (substrate) in a buffer solution. The concentration of an analyte is determined using the calibration curve prepared beforehand. Amperometric biosensors are known to be reliable, cheap and highly sensitive for environment monitoring, food analysis, clinical diagnostics, drug analysis and some other purposes [4]–[7].

Very frequently biosensors operate following the Michaelis-Menten kinetics scheme [2], [3],

$$\mathrm{E} + \mathrm{S} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} \mathrm{ES} \xrightarrow{k_2} \mathrm{E} + \mathrm{P}, \tag{1}$$

where E is the enzyme, S is the substrate, ES is the enzyme and substrate complex, and P is the reaction product, k_i is the reaction rate constant, $i = -1, 1, 2$. However, sometimes the kinetics of enzyme-catalysed reactions is much more complex. An inhibition, an activation, an allostery and other types of non-Michaelis-Menten kinetics are known for the diversity of enzymes [8]–[12].

This paper investigates the case when the standard scheme (1) is augmented with two more reactions

$$\mathrm{ES} + \mathrm{S} \underset{k_{-3}}{\overset{k_3}{\rightleftharpoons}} \mathrm{ESS}, \tag{2}$$

$$\mathrm{E} + \mathrm{P} \underset{k_{-4}}{\overset{k_4}{\rightleftharpoons}} \mathrm{EP}, \tag{3}$$

where ESS is a non-active complex with substrate molecule and EP is a non-active complex with product molecule, k_i is the reaction rate constant, $i = 3, -3, 4, -4$. The overall effect of reaction (2) is called substrate inhibition and the overall effect of reaction (3) is called product inhibition.

It is very important to investigate kinetic peculiarities of the biosensors [1]–[3]. In order to perform such investigation a model of a biosensor should be built [13], [14]. A thorough review on the modelling of the amperometric biosensors has been presented by Schulmeister [15] and more recently by Baronas et al. [16]. The same type of biosensors has been investigated in the paper by the same authors [17]. This paper enhances the results of investigation [17] and emphasizes on the sensitivity of biosensor at wide range of the inhibition constants, diffusion modulus and Biot values. Apparent Michaelis constant is used as a good indicator of biosensor sensitivity. A numerical simulation has been carried out using a finite difference technique [18], [19].

The rest of the paper is organised as follows: in Section II the mathematical model is described; Section III briefly describes the numerical model and the simulator; in Section IV we present results of numerical experiments; and, finally, the main conclusions are summarized in Section V.

II. Mathematical Model

Main parts of the amperometric biosensor are an electrode and a layer of enzyme applied on the electrode surface. The mathematical model consists of two layers: enzyme layer and diffusion layer. In enzyme layer enzymatic reaction and diffusion take place while in diffusion layer only diffusion

ISBN: 978-1-61208-169-4

takes place. Outside the diffusion layer is the part of solution where the concentration of the analyte is kept constant.

Consideration that electrode and enzyme layer are symmetrical as well as consideration of homogeneous distribution of the immobilized enzyme in the enzyme membrane allows definition of one-dimensional-in-space mathematical model of the biosensor [15].

A. Governing Equations

By omitting the details of the catalysis mechanism, reaction scheme (1)–(3) may be generalized to the following form:

$$\mathrm{S} \xrightarrow{\mathrm{E}} \mathrm{P}. \tag{4}$$

Applying the quasi-steady-state approximation the rate of the reaction (4) may be expressed as follows [17]:

$$v(s_e, p_e) = \frac{V_{max} s_e}{k_M \left(1 + p_e/k_p\right) + s_e \left(1 + s_e/k_s\right)}, \tag{5}$$

where $s_e(x,t)$ and $p_e(x,t)$ are the molar concentrations of the substrate S and the product P in the enzyme layer, x and t stand for space and time, respectively, V_{max} is the maximal enzymatic rate, k_M is the Michaelis-Menten constant, k_s is the substrate inhibition rate, and k_p is the product inhibition rate [1], [2], [20]. These latter parameters are expressed as follows:

$$V_{max} = k_2 e_0 = k_2(e_e + e_{es} + e_{ess} + e_{ep}), \tag{6a}$$

$$k_M = \frac{k_{-1} + k_2}{k_1}, \quad k_s = \frac{k_{-3}}{k_3}, \quad k_p = \frac{k_{-4}}{k_4}, \tag{6b}$$

where $e_e(x,t)$, $e_{es}(x,t)$, $e_{ess}(x,t)$ and $e_{ep}(x,t)$ are molar concentrations of the enzyme E, the ES complex, the ESS complex and the EP complex, respectively. e_0 is the total sum of the concentrations of all the enzyme forms, $e_0 = e_e + e_{es} + e_{ess} + e_{ep}$. e_0 is assumed to be constant in the entire enzyme layer.

Let $x = 0$ represents the electrode surface, $x = d_e$ is the boundary between the enzyme and the diffusion layers, and $x = d_e + d_d$ is the boundary between the diffusion layer and the bulk solution.

The governing equations for a chemical reaction network can be formulated by the law of mass action [1], [21]. Coupling of the enzyme-catalysed reaction in the enzyme layer with the one-dimensional-in-space diffusion, described by Fick‘s law, leads to the following equations of the reaction–diffusion type ($t > 0$):

$$\frac{\partial s_e}{\partial t} = D_{s_e} \frac{\partial^2 s_e}{\partial x^2} - v(s_e, p_e), \tag{7a}$$

$$\frac{\partial p_e}{\partial t} = D_{p_e} \frac{\partial^2 p_e}{\partial x^2} + v(s_e, p_e), \quad 0 < x < d_e, \tag{7b}$$

where d_e is the thickness of the enzyme layer, D_{s_e} and D_{p_e} are the diffusion coefficients of the substrate and the reaction product in the enzyme layer.

Outside the enzyme layer only the mass transport by diffusion of the substrate and the product takes place. We assume that the external mass transport obeys a finite diffusion regime,

$$\frac{\partial s_d}{\partial t} = D_{s_d} \frac{\partial^2 s_d}{\partial x^2}, \tag{8a}$$

$$\frac{\partial p_d}{\partial t} = D_{p_d} \frac{\partial^2 p_d}{\partial x^2}, \quad d_e < x < d_e + d_d, \quad t > 0, \tag{8b}$$

where $s_d(x,t)$ and $p_d(x,t)$ stand for concentrations of the substrate and the product in the diffusion layer, d_d is the thickness of the external diffusion layer, D_{s_d} and D_{p_d} are the diffusion coefficients in the diffusion layer.

B. Initial and Boundary Conditions

The biosensor operation starts when some substrate appears in the bulk solution ($t = 0$),

$$s_e(x,0) = 0, \quad p_e(x,0) = 0, \quad 0 \le x \le d_e, \tag{9a}$$

$$s_d(x,0) = 0, \quad p_d(x,0) = 0, \quad d_e \le x < d_e + d_d, \tag{9b}$$

$$s_d(d_e + d_d, 0) = s_0, \quad p_d(d_e + d_d, 0) = 0, \tag{9c}$$

where s_0 is the concentration of the analyte (substrate) in the bulk solution.

Due to the electrode polarization, concentration of the reaction product at the electrode surface ($x = 0$) is permanently reduced to zero [15],

$$p_e(0,t) = 0. \tag{10}$$

Since the substrate is not ionized, the substrate concentration flux on the electrode surface equals zero,

$$D_{s_e} \left. \frac{\partial s_e}{\partial x} \right|_{x=0} = 0. \tag{11}$$

The external diffusion layer ($d_e < x < d_e + d_d$) is treated as the Nernst diffusion layer [18]. According to the Nernst approach the layer of the thickness d_d remains unchanged with time. It is also assumed that away from it the solution is uniform in the concentration ($t > 0$),

$$s_d(d_e + d_d, t) = s_0, \tag{12a}$$

$$p_d(d_e + d_d, t) = 0. \tag{12b}$$

On the boundary between two regions having different diffusivities, the matching conditions have to be defined ($t > 0$),

$$D_{s_e} \left. \frac{\partial s_e}{\partial x} \right|_{x=d_e} = D_{s_d} \left. \frac{\partial s_d}{\partial x} \right|_{x=d_e}, \tag{13a}$$

$$s_e(d_e,t) = s_d(d_e,t), \tag{13b}$$

$$D_{p_e} \left. \frac{\partial p_e}{\partial x} \right|_{x=d_e} = D_{p_d} \left. \frac{\partial p_d}{\partial x} \right|_{x=d_e}, \tag{13c}$$

$$p_e(d_e,t) = p_d(d_e,t). \tag{13d}$$

According to these conditions, the substrate and the product concentration fluxes through the external diffusion layer are equal to the corresponding fluxes entering the surface of the enzyme layer. The concentrations of the substrate as well as the product from both layers are equal on the boundary between these layers.

C. Biosensor Response

The electric current is measured as a response of a biosensor in a physical experiment. The current depends on a flux of reaction product at an electrode surface. Thus the density i of the current at time t is proportional to the gradient of the product at the electrode surface, i.e., at the border $x = 0$, as described by Faraday's law,

$$i(t) = n_e F D_{p_e} \left. \frac{\partial p_e}{\partial x} \right|_{x=0}, \qquad (14)$$

where n_e is a number of electrons involved in the electrochemical reaction, and F is Faraday's constant ($F = 96486\,\mathrm{C/mol}$) [2], [15].

Usually the steady-state current is used as a response of amperometric biosensor. However usage of steady-state current is not convenient when biosensor exhibits substrate and product inhibition, because steady-state current is directly proportional to s_0 only in part of the calibration curve [17]. The maximal biosensor current does not have this drawback,

$$i_{max} = \max_{t>0} i(t), \qquad (15)$$

where i_{max} is the density of the maximal biosensor current.

D. Apparent Michaelis Constant

At the ideal conditions of the Michaelis-Menten model the rate of generalized reaction (4) is defined as follows:

$$v(s_0) = \frac{V_{max} s_0}{k_M + s_0}.$$

The maximal possible rate of generalized reaction (4) is equal

$$\lim_{s_0 \to \infty} v(s_0) = \lim_{s_0 \to \infty} \frac{V_{max} s_0}{k_M + s_0} = V_{max}$$

If the concentration s_0 is numerically equal to k_M then the rate of reaction (4) is equal to half the maximal possible reaction rate,

$$v(k_M) = \frac{V_{max} k_M}{k_M + k_M} = 0.5 V_{max}$$

If the biosensor would work at ideal Michaelis-Menten conditions, it would be possible to calculate k_M using the calibration curve $i_{max}(s_0)$, because $v(s_0)$ is proportional to $i_{max}(s_0)$,

$$V_{max} \sim \lim_{s_0 \to \infty} i_{max}(s_0), \quad 0.5 V_{max} \sim 0.5 \lim_{s_0 \to \infty} i_{max}(s_0).$$

When the Michaelis-Menten constant is calculated for the particular biosensor from the calibration curve it is called the apparent Michaelis constant k_{app},

$$k_{app} = \left\{ s_0^* : i_{max}(s_0^*) = 0.5 \lim_{s_0 \to \infty} i_{max}(s_0) \right\}. \qquad (16)$$

Greater k_{app} value means longer range of substrate concentrations in which the calibration curve resembles a linear function. Whereas other parts of the curve are largely not suitable for the biosensor operation. This is the reason why k_{app} is an attractive parameter that helps to measure the sensitivity of biosensor.

Usually, for real biosensors $k_{app} \neq k_M$ [3]. Theoretical modelling has shown that under certain conditions k_{app} depends on the biosensor geometry [22]. It has been shown that k_{app} can be increased by the restriction of the substrate diffusivity [23]. This result can be easily applied for the biosensor improvement by covering the enzyme layer of a biosensor with a permeable membrane [23].

E. Limitations of Mathematical Model

The presented mathematical model is a simplified view of processes taking place during physical biosensor operation. Some processes are not reflected in the mathematical model. Physical experiments must obey some constraints in order to minimize the influence of those neglected processes.

The quasi-steady-state approximation used in the mathematical model neglects the fact that concentrations of enzyme forms (e_e, e_{es}, e_{ess} and e_{ep}) change at the beginning of a physical experiment. If the equilibrium between enzyme forms is reached fast enough this approximation is quite accurate though [20].

The enzyme layer should be of uniform thickness and the enzyme should be homogeneously distributed throughout this layer. This is an assumption leading to the construction of one-dimensional mathematical model. This approach is widely used and reliable, even though these conditions are not always satisfied because the enzyme layer often has more complicated geometry [16].

III. Numerical Simulation

The non-linearity of the governing equations prevents us from solving the initial boundary value problem (7)–(13) analytically, hence the numerical model is constructed and solved using finite difference technique [15], [18], [24]. An implicit finite difference scheme was built on a uniform discrete grid with 200 points in space direction [11], [17], [25], [26]. The simulator has been programmed by the authors in C language [27].

In the numerical simulation, the biosensor response time was assumed as the time when the change of the biosensor current over time remains very small during a relatively long term or when the biosensor current reaches local maximum (which is the global function maximum too). A special dimensionless decay rate ε was used,

$$t_r = \min_{i(t)>0} \left\{ t : \frac{t}{i(t)} \frac{\mathrm{d}i(t)}{\mathrm{d}t} < \varepsilon \right\}, \quad i(t_r) \approx i_{max}, \tag{17}$$

where t_r is the biosensor response time. The decay rate value $\varepsilon = 10^{-3}$ was used in the calculations.

In all numerical experiments the following values were kept constant:

$$\begin{aligned} D_{s_e} &= D_{p_e} = 100\,\mu\mathrm{m}^2/\mathrm{s}, \\ D_{s_d} &= 2D_{s_e}, \quad D_{p_d} = 2D_{p_e}, \\ k_M &= 0.01\,\mathrm{M}, \quad d_e = 10\,\mu\mathrm{m}, \quad n_e = 1. \end{aligned} \tag{18}$$

IV. Results and Discussion

In order to conveniently analyse the simulation results, five dimensionless parameters were introduced:

$$\begin{aligned} K_{app} &= \frac{k_{app}}{k_M}, \quad K_s = \frac{k_s}{k_M}, \quad K_p = \frac{k_p}{k_M}, \\ \alpha^2 &= \frac{V_{max} d_e^2}{D_{S_e} k_M}, \quad Bi = \frac{d_e / D_{S_e}}{d_d / D_{S_d}} = \frac{D_{S_d} d_e}{D_{S_e} d_d}, \end{aligned} \tag{19}$$

where K_{app}, K_s and K_p are the dimensionless apparent Michaelis constant, dimensionless substrate inhibition constant and dimensionless product inhibition constant, respectively, α^2 is called the diffusion mudulus and Bi is the Biot number.

A. Apparent Michaelis Constant vs. Substrate and Product Inhibition

The dependence of the apparent Michaelis constant on the substrate and product inhibition rates were investigated in a wide range of inhibition constant values ($K_s, K_p \in [10^{-4}..10^{4}]$). The dependence on the substrate inhibition was investigated at three fixed rates of the product inhibition: no product inhibition ($K_p \to \infty$, curve 1), moderate product inhibition ($K_p = 1$, curve 2) and high product inhibition ($K_p = 0.01$, curve 3). In the case of no product inhibition, the reaction scheme (1)–(3) reduces to scheme (1), (2). The dependence on the product inhibition was investigated at three fixed rates of the substrate inhibition: no substrate inhibition ($K_s \to \infty$, curve 4), moderate substrate inhibition ($K_s = 1$, curve 5) and high substrate inhibition ($K_s = 0.01$, curve 6). In the case of no substrate inhibition, the reaction scheme (1)–(3) reduces to scheme (1), (3). Other parameters were kept as follows: $\alpha^2 = 0.01$, $Bi = 1/15 (d_d = 300\,\mu\mathrm{m})$. The results of the numerical simulation are depicted in Figure 1.

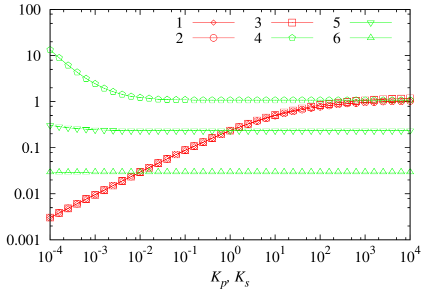


Figure 1. The dependence of the apparent Michaelis constant K_{app} on substrate (1, 2, 3) and product (4, 5, 6) inhibition constants K_s and K_p, respectively.

The apparent Michaelis constant does not dependent on the product inhibition rate at the very wide range of product inhibition constant values ($K_p \in [10^{-2}..10^{4}]$). However at extremely high rates of product inhibition ($K_p \in [10^{-4}..10^{-2}]$) and no substrate inhibition (curve 4), the apparent Michaelis constant is dependent on product inhibition rate change. K_{app} is inversely proportional to the K_p. However, the presence of substrate inhibition eliminates the effect. In the case of a moderate substrate inhibition ($K_s = 1$, curve 5) the infuence is barely observable and in the case of high substrate inhibition ($K_s = 0.01$, curve 6) the effect vanishes.

As one can see from the Figure 1 (curves 1, 2 and 3) the apparent Michaelis constant continuously and nonlinearly increases with an increase in K_s. At low rates of the substrate inhibition the slope of curves starts to decrease as the function approaches the maximal value of K_{app} at these particular biosensor parameters. As moderate and low product inhibition rates do not influence K_{app} value, all three curves depicting the dependence on K_s almost entirely coincide. However at low substrate inhibition values (high K_s values), the curve 3 representing high product inhibition ($K_p = 0.01$) slightly separates from curves 1, 2 representing no product inhibition and low product inhibition ($K_p = 1$). This is the same positive effect of the product inhibition that is clearly observed on curve 4.

After examination of Figure 1 we can conclude that the substrate and product inhibitions have opposite effects on the apparent Michaelis constant. However, the infuence of the substrate inhibition is evident at the very wide range of substrate inhibition values, while infuence of the product inhibition is evident only at the very high rates of product inhibition and when this effect is not masked by the opposite effect of substrate inhibition.

B. Apparent Michaelis Constant vs. Diffusion Modulus

To investigate the dependence of the apparent Michaelis constant on the diffusion modulus α^2, K_{app} was calculated simulating biosensor action at three values of the substrate inhibition: high substrate inhibition ($K_s = 0.01$), moderate substrate inhibition ($K_s = 0.1$) and low substrate inhibition ($K_s = 1$) as well as at three values of the Biot number: $Bi = 0.01$, $Bi = 1$ and $Bi = 100$. Calculation results are depicted in Figure 2.

As it is evident from Figure 2, the apparent Michaelis constant is directly proportional to the diffusion modulus

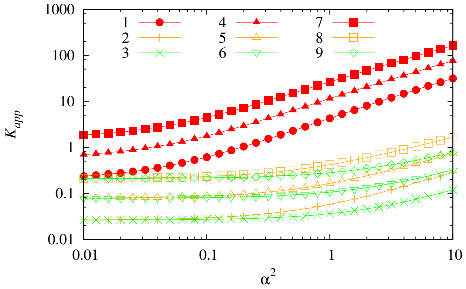


Figure 2. The dependence of the apparent Michaelis constant K_{app} on the diffusion modulus α^2 at three rates of the substrate inhibition (K_s): 0.01 (1, 2, 3), 0.1 (4, 5, 6), 1 (7, 8, 9) and at three rates of the Biot number Bi: 0.01 (1, 4, 7), 1 (2, 5, 8), 100 (3, 6, 9), $K_p = 0.1$.

α^2, not in a whole range of α^2 though. The value of Biot number determines the point at which the diffusion modulus starts to infuence the K_{app} value. As one can see from Figure 2, when the Biot number is low (curves 1, 4 and 7) the diffusion modulus starts infuencing the K_{app} at the values as low as $\alpha^2 = 0.01$, when the Biot number is moderate and high (curves 2, 5, 8 and 3, 6, 9, respectively) the diffusion modulus starts infuencing the K_{app} at the values of $\alpha^2 = 0.1$. By comparing the steepness of curves we can deduce that the Biot number also determines the sensitivity of the K_{app} to the α^2. The curves 1, 4 and 7 that represent a small value of the Biot number ($Bi = 0.01$) are steeper than curves 2, 5, 8 that represent a moderate Biot number value ($Bi = 1$) which are steeper than curves 3, 6, 9 that represent a high Biot number value ($Bi = 100$) accordingly. The more steep the curve is, the more sensitive is the apparent Michaelis constant to the diffusion modulus.

The substrate inhibition rate influences the apparent Michaelis constant in the whole investigated range of the diffusion modulus α^2 as well as at all investigated values of the Biot number. K_{app} is directly proportional to the substrate inhibition constant K_s.

The apparent Michaelis constant is inversely proportional to the Biot number. The values of K_{app} are higher at low Biot number values $Bi = 0.01$ (curves 1, 4, 7) than at high and moderate Biot number values $Bi = 100$ (curves 3, 6, 9) and $Bi = 1$ (curves 2, 5, 8), respectively. However when the Biot number is moderate and high and at lower values of diffusion modulus, the Biot number does not influence K_{app}.

C. Apparent Michaelis Constant vs. Biot Number

Figure 3 represents the effect of the Biot number on the apparent Michaelis constant. One can see in Figure 3, that K_{app} is a monotonous descreasing function of Bi. However, at higher values of Bi the function reaches the steady-state and K_{app} value sets in. The range of Bi where the function $K_{app}(Bi)$ is at steady-state depends on the diffusion modulus though. When the diffusion modulus is low $\alpha^2 = 0.1$ (curves 1, 4), the range of steady-state is wide ($Bi \in [1..100]$), when diffusion modulus is moderate and high $\alpha^2 = 1$ and $\alpha^2 = 10$, respectively, the range of steady-state is narrower.

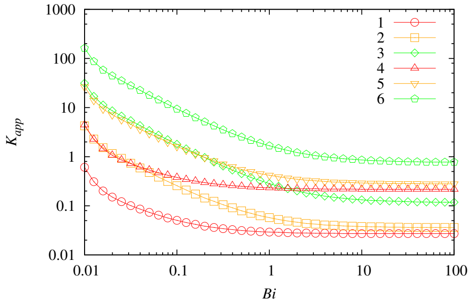


Figure 3. The dependence of the apparent Michaelis constant K_{app} on the Biot number Bi at two rates of the substrate inhibition (K_s): 0.01 (1, 2, 3), 1 (4, 5, 6) and at three rates of the diffusion modulus α^2: 0.1 (1, 4), 1 (2, 5), 10 (3, 6), $K_p = 0.1$.

V. Conclusion

The mathematical model (7)–(13) of the amperometric biosensor with the substrate and product inhibition can be successfully used to investigate the behaviour of the biosensor response at various sets of parameters. The model can be used as a tool to optimize the biosensor configuration prior to the experimental stage.

The substrate inhibition decreases the value of the apparent Michaelis constant, hence designers of biosensors should avoid the substrate inhibition if possible. Whereas the product inhibition may increase the value of the apparent Michaelis constant and make the biosensor more attractive (Figure 1).

If the substrate inhibition is unavoidable and the apparent Michaelis constant is low, the biosensor can be improved by increasing the diffusion modulus α^2 (Figure 2). Practically this can be achieved be increasing the enzyme layer thickness d_e or by increasing enzyme concentration e_0.

Another possibility to improve the biosensor is to increase the external diffusion layer thickness d_d or decrease the substrate diffusivity D_{S_d}. This can be achieved be decreasing the intensity of solution stirring or by covering the enzyme layer of a biosensor with a permeable membrane which would decrease the substrate diffusivity. In worst cases this method may at least move the value of apparent Michaelis constant close to the Michaelis-Menten constant. In the best cases, apparent Michaelis constant may overwhelm the Michaelis-Menten constant by a few orders of magnitude (Figure 3).

Acknowledgment

This research was funded by the European Social Fund under Measure VP1-3.1-ŠMM-07-K ”Support to Research of Scientists and Other Researchers (Global Grant)", Project "Developing computational techniques, algorithms and tools for efficient simulation and optimization of biosensors of complex geometry".

References

[1] H. Gutfreund, *Kinetics for the Life Sciences*. Cambridge: Cambridge University Press, 1995.

[2] F. W. Scheller and F. Schubert, *Biosensors*. Amsterdam: Elsevier Science, 1992.

[3] A. P. F. Turner, I. Karube, and G. S. Wilson, *Biosensors: Fundamentals and Applications*. Oxford: Oxford University Press, 1990.

[4] J. F. Liang, Y. T. Li, and V. C. Yang, "Biomedical application of immobilized enzymes," *Journal of Pharmaceutical Sciences*, vol. 89, no. 8, pp. 979–990, 2000.

[5] K. R. Rogers, "Biosensors for environmental applications," *Biosensors and Bioelectronics*, vol. 10, no. 6–7, pp. 533–541, 1995.

[6] F. W. Scheller, F. Schubert, and J. Fedrowitz, *Frontiers in Biosensorics II. Practical Applications*. Basel: Birkhäuser, 1997, vol. 2.

[7] D. Yu, B. Blankert, J. C. Vire, and J. M. Kauffmann, "Biosensors in drug discovery and drug analysis," *Analytical Letters*, vol. 38, no. 11, pp. 1687–1701, 2005.

[8] A. Chaubey and B. D. Malhotra, "Mediated biosensors," *Biosensors and Bioelectronics*, vol. 17, no. 6–7, pp. 441–456, 2002.

[9] A. Cornish-Bowden, *Fundamentals of Enzyme Kinetics*, 3rd ed. London: Portland Press, 2004.

[10] N. C. of the International Union of Biochemistry, "Symbolism and terminology in enzyme kinetics," *Biochemical Journal*, vol. 213, no. 3, pp. 561–571, 1983.

[11] R. Baronas, F. Ivanauskas, and J. Kulys, "The effect of diffusion limitations on the response of amperometric biosensors with substrate cyclic conversion," *Journal of Mathematical Chemistry*, vol. 35, no. 3, pp. 199–213, 2004.

[12] J. Kulys, "Biosensor response at mixed enzyme kinetics and external diffusion limitation in case of substrate inhibition," *Nonlinear Analysis: Modelling and Control*, vol. 11, no. 4, pp. 385–392, 2006.

[13] J. R. D. Corcuera, R. Cavalieri, J. Powers, and J. Tang, "Amperometric enzyme biosensor optimization using mathematical modeling," in *Proceedings of the 2004 ASAE / Csae Annual International Meeting*. Ottawa, Ontario: American Society of Agricultural Engineers, 2004, p. 47030.

[14] L. S. Ferreira, M. B. D. Souza, J. O. Trierweiler, O. Broxtermann, R. O. M. Folly, and B. Hitzmann, "Aspects concerning the use of biosensors for process control: experimental and simulation investigations," *Computers and Chemical Engineering*, vol. 27, no. 8, pp. 1165–1173, 2003.

[15] T. Schulmeister, "Mathematical modelling of the dynamic behaviour of amperometric enzyme electrodes," *Selective Electrode Reviews*, vol. 12, pp. 203–260, 1990.

[16] R. Baronas, F. Ivanauskas, and J. Kulys, *Mathematical Modeling of Biosensors*, ser. Springer Series on Chemical Sensors and Biosensors, G. Urban, Ed. Dordrecht: Springer, 2010, vol. 9.

[17] D. Šimelevičius and R. Baronas, "Computational modelling of amperometric biosensors in the case of substrate and product inhibition," *Journal of Mathematical Chemistry*, vol. 47, no. 1, pp. 430–445, 2010.

[18] D. Britz, *Digital Simulation in Electrochemistry*, 3rd ed., ser. Lecture Notes in Physics. Berlin: Springer, 2005, vol. 666.

[19] A. A. Samarskii, *The Theory of Difference Schemes*. New York: Marcel Dekker, 2001.

[20] B. Li, Y. Shen, and B. Li, "Quasi-steady state laws in enzyme kinetics," *The Journal of Physical Chemistry A*, vol. 112, no. 11, pp. 2311–2321, 2008.

[21] P. N. Bartlett and R. G. Whitaker, "Electrochemical imobilisation of enzymes: Part 1. theory," *Journal of Electroanalytical Chemistry*, vol. 224, no. 1–2, pp. 27–35, 1987.

[22] F. Ivanauskas, I. Kaunietis, V. Laurinavičius, J. Razumienė, and R. Šimkus, "Apparent Michaelis constant of the enzyme modified porous electrode," *Journal of Mathematical Chemistry*, vol. 43, no. 4, pp. 1516–1526, 2008.

[23] O. Štikonienė, F. Ivanauskas, and V. Laurinavičius, "The influence of external factors on the operational stability of the biosensor response," *Talanta*, vol. 81, no. 4-5, pp. 1245–1249, 2010.

[24] J. P. Kernevez, *Enzyme Mathematics. Studies in Mathematics and its Applications*. Amsterdam: Elsevier Science, 1980.

[25] R. Baronas, J. Kulys, and F. Ivanauskas, "Computational modeling of biosensors with perforated and selective membranes," *Journal of Mathematical Chemistry*, vol. 39, no. 2, pp. 345–362, 2006.

[26] R. Baronas, F. Ivanauskas, and J. Kulys, "Computational modeling of the behaviour of potentiometric membrane biosensors," *Journal of Mathematical Chemistry*, vol. 42, no. 3, pp. 321–336, 2007.

[27] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. Cambridge: Cambridge University Press, 1992.

A Generic Operational Simulation for Early Design Civil Unmanned Aerial Vehicles

Benjamin Schumann, Jim Scanlan
School of Engineering Sciences
University of Southampton
Southampton, UK
Benjamin.Schumann@soton.ac.uk
J.P.Scanlan@soton.ac.uk

Kenji Takeda
Microsoft Research
Cambridge, UK
kenjitak@microsoft.com

***Abstract*—Contemporary aerospace programmes often suffer from large cost overruns, delivery delays and inferior product quality. This is caused in part by poor predictive quality of the early design phase processes with regards to the operational environment of a product. This paper develops the idea of a generic operational simulation that can help designers to rigorously analyse and test their early product concepts. The simulation focusses on civil Unmanned Air Vehicle products and missions to keep the scope of work tractable. The research agenda is introduced along with ideas, initial results and future work. Designers specify details about their product, its environment and anticipated operational procedures. The simulation returns information that can help to estimate the value of the product using the value-driven design approach. Information will include recurring and non-recurring mission cost items. The research aim is to show that an operational simulation can improve early design concepts, thereby reducing delays and cost overruns. Moreover, a trade-off between mission fidelity and model generality is sought along with a generic ontology of civil Unmanned Air Vehicle missions and guidelines about capturing operational information.**

***Index Terms*—Aerospace testing; Remotely operated vehicles; Relational databases; Maintenance; Computational modeling; Object oriented modeling; Computer simulation; Discrete time systems; Unmanned aerial vehicles;**

I. Introduction

This paper introduces and explores the concept of an operational simulation for improving the early design process of aerospace products. The work aims to sketch out a road map to obtain a PhD. Section II establishes the current problems in early design and defines important concepts. Section III specifies the four main research questions to be answered by the project. Subsequently, initial results obtained so far are presented in section IV. Lastly, section V represents the main part of this paper, introducing the necessary steps towards answering the research questions.

II. Background

This section reviews the most important problems of the early design phase of aerospace products. Subsequently, it introduces the concept of Value-driven Design (VDD) and how it can help to alleviate these problems. Lastly, the concept of an operational simulation is introduced together with a more narrow definition applicable for this research.

A. Early design phase problems

Aerospace products are designed according to customer demand. Customers specify the required product and manufacturers focus their effort to meet expectations. During the early design phase, precise information about the final product and its environment is unknown [1]. Much work is based on expert opinion and subjective engineering judgement rather than rigorous, systematic and disciplined analysis [2]. However, wide-ranging decisions taking into account performance, environmental, legal, disposal and operational aspects must be agreed in a very short period of time. Early design phase decisions are generally acknowledged to determine the success of aerospace products to a great extend [3]. Therefore, improvements to the early design phase procedures can potentially improve aerospace products.

The current design process is guided by fixed, static specification documents that cannot account for the dynamic and complex nature of the product environment. This leads to unsatisfactory understanding of the product domain, which can be a primary cause for product failure [4]. The current processes do not account for the full life-cycle of a product, fail to evaluate operational costs properly and do not focus on the value a product generates [5]. Moreover, specification documents cannot capture all eventualities and details of the operational environment of the product. Partly due to these shortcomings, contemporary aerospace products regularly suffer from huge cost overruns, delivery delays and quality problems. It is becoming essential to found early design decisions upon tangible, tractable and rigorous decisions to create competitive products.

One important aspect currently skipped during early product design decision making is the operational environment the product will work in. As designers focus to meet customer expectations, they neglect to investigate how successful a product design will be in specific operational environments. Simulating the operational environment and the product interactions early on can help improve the current situation.

Another major problem occurring in early product design is the disregard for non-recurring costs in cost estimates [5]. However, non-recurring costs such as payload integration,

ISBN: 978-1-61208-169-4

transportation to the operational environment, support personnel costs and the influence of the design usually form a large part of total life-cycle costs. For example, about two thirds of the total cost of contemporary scientific Unmanned Air Vehicle (UAV) missions are non-recurring cost items [6]. An operational simulation can help to estimate these costs and improve the quality of cost estimates significantly.

B. Value-driven design

VDD emerged as a new approach to designing aerospace products that tries to reduce the problems of current design processes [7]. It aims to find the optimal design and not just any design that satisfies customer specifications. This is achieved by introducing a value function that incorporates life-cycle performance, product and operational information. The value function returns the product's value and allows intuitive comparison of different designs. The aim is to optimize a product for the value it delivers, possibly violating customer specifications for a better design and higher value [8]. The value function returns transparent and consistent scores to individual designs. Thereby, it replaces the traditional customer requirements like maximum weight or cost.

An optimized design can only be as good as the value function in use. The quality of the value function depends in part on the quality of the operational knowledge. In order to improve the operational knowledge during the early design process, an operational simulation can be helpful. By simulating operations, performance and environmental processes, novel insights can be gained into the product concept. Moreover, product performance can now be compared not only to different product designs but also to various operational scenarios.

C. Operational simulation

Operational simulations started to become widespread during the early 1990ies following growing animation capabilities and improved computing facilities. In the traditional sense, an operational simulation is used to support short-term planning and decisions within manufacturing scenarios. The models are highly detailed and realistic, feeding on real-time data to allow "live" decision making for operators [9]. The simulations can also be used to predict the near future to discern alternative decision scenarios. Today, operational simulations are predominantly used within transportation management and manufacturing: Railway timetabling is now conducted using operational simulations to verify scenarios and ensure operational stability [10]; operational simulations supported the development of hybrid cars by assessing the costs and benefits of batteries for various types of drivers [11].

In the context of this research, an operational simulation is defined as a non-analytical model recreating anticipated operations of a product during its service life. The model does not recreate existing missions. Instead, the aim is to embed the product into its future operational environment. Therefore, real-time data management and decision making are not aspects of the operational simulation presented here. This new approach to operational simulations is explained in more detail below.

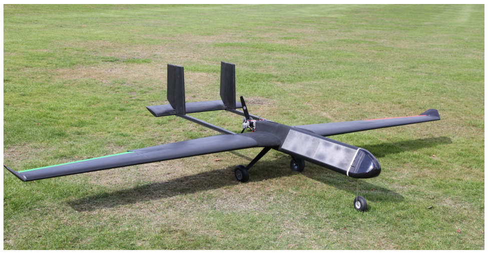


Fig. 1. The DECODE-UAV

III. Research questions

The aim of this research is to enable designers to create better products by improving the early design process. Products are tested within an operational simulation to observe how useful they are, how much value they produce and if important operational constraints exist. The research will investigate several key issues:

1) Can an operational simulation improve a product early on?
2) Is it possible to create a generic operational model to simulate different aerospace scenarios for various aerospace products? Where is the trade-off between adequate mission fidelity and sufficient model generality? What is adequate fidelity?
3) How can operational information be captured during the early design process?
4) What ontology can be used to unify various aerospace scenarios?

IV. Results so far

Some of these questions have been answered already while others remain open. This section presents the results obtained so far.

As a first step towards answering the research questions, a specific operational simulation has been created to support the design of a Search-and-Rescue (SAR) UAV for the DECODE-project (DEcision Environment for COmplex DEsigns) at the University of Southampton [12]. This UAV has been build based (among others) upon the results of the simulation and can be seen in Figure 1. The software of choice was AnyLogic [13], a Java-based simulation tool, which is able to combine Agent-based modelling with a discrete-event paradigm and visual algorithms.

The operational simulation is implemented into an iterative value-driven design work flow. Computer-aided Design (CAD), Computational Fluid Dynamics (CFD) and structural analysis tools supply product information such as weight, cruise speed and specific fuel consumption. Subsequently, 10 years of operations are simulated and operational parameters such as the total fuel used, the number of maintenance operations or the attrition of airframes are returned. This information

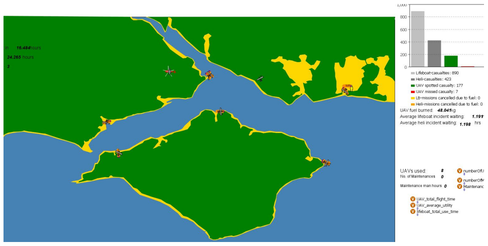


Fig. 2. The simulation animation

is used to calculate a value for the product design in order to optimize it.

The simulation helped to improve the product by supporting the early design phase in two ways: (i) It was used "actively" by designers as an optimization tool within the operational environment and (ii) it was used "reactively" to inform about extensive product attributes such as maximum permissible cost. It was shown that it is possible to capture, understand and simulate operational information during the early design phase (research question 1). This was done through interviews, data acquisition and careful model building (see Figure 2).

However, the simulation turned out to be highly specific, restricting its use for SAR operations within a pre-defined area only. Moreover, the simulation fidelity was very high and subsequent simplification did not alter characteristic results. The knowledge and experience gained will be used to create a flexible and generic operational simulation incorporating an optimum level of fidelity (research questions 3 and 4).

V. Future work

This section details the steps necessary to answer the research questions. First, the scope of the simulation is specified followed by details on how to unify various mission scenarios. Subsequently, the model building phase is described followed by how the results will be obtained and help to answer the research questions.

A. *Model scope*

A first step towards a generic operational simulation is to define the scope of the model. A focus on civil UAV operations is imposed in order to keep the task manageable. This choice is based on practical considerations: The civil UAV sector is starting to grow as it has the potential to support and replace many "dull, dirty and dangerous missions" [5]. It is easy to validate simulation results against reality because civil UAV operations are small in scale compared to commercial or military operations. However, it is desirable to keep the simulation open for use in other domains such as commercial airliners. Despite regulatory and liability issues waiting to be resolved, the market forecasts for civil UAVs promise rapid expansion [14]. NASA [5] has identified a number of key operational areas suitable for viable and cost-effective use of UAVs. Figure 3 presents a selection of missions planned to be included into this research. The portfolio covers the majority of possible civil mission applications.

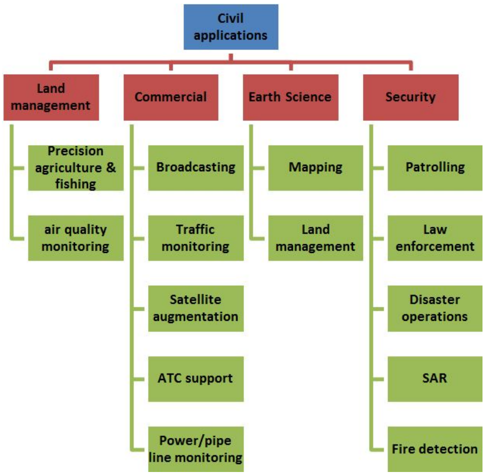


Fig. 3. Classification of civil UAV missions (adapted from [5])

B. *Unify mission specifications*

The next step will be to investigate mission characteristics and unify mission stages, requirements and definitions into a coherent ontology that can map any one mission into any other. This will enable designers to test-run products in pre-defined scenarios and also enable comparison of different missions. Thereby, a unique way to improve the early product design is found because one of the major barriers to successful market introduction is the capability to fly multiple types of missions [5].

One example of ontological unification are the various mission goals: Each one can be reduced into one of the following categories:

- to find something (casualties, animals, pollution)
- to cover a certain area (agriculture, patrolling)
- to stay at a fixed point (traffic, broadcasting)
- to follow a track (pipelines, borders)

Simplifications like this will be sought for other mission properties such as frequency, length, flight profiles (vertical and horizontal), typical weather, visibility and the number of UAVs involved.

The UAVs will be specified by parameters such as specific fuel consumption, range, endurance and fuel capacity. The on-board equipment characteristics will include the type of equipment (camera, sensor or radios), over-the-horizon communication requirements (what SatCom bandwidth, time of usage, amount of data) and on-board analysis equipment (image analysers, etc.), which reduce bandwidth. As with mission parameters, UAV parameters will be unified into one ontology.

All specifications will be accompanied by reliability estimates in order to increase trustworthiness into simulation results. Useful reliability figures will help users to estimate insurance costs as well. Moreover, all specifications can be entered using confidence intervals in order to reflect information uncertainty.

The operational simulation output will enable better cost estimation by including traceable and comprehensible operational information about UAV consumables, transit operations, maintenance, staff requirements, payload installation and SatCom requirements. However, some aspects will still require traditional cost estimates such as payload development, data analysis, documentation or mission planning.

C. Model building

Subsequently, the generic operational simulation will be constructed based on the developed ontology. Separating data and simulation logic is good modelling practice to support understanding of users and developers. As a first step, an external database will capture the ontology details. This ensures that more mission scenarios can be added to the simulation later on. User input will be possible through the external database, the ontology tool or directly in the runtime environment of the Java-application. Users can select a pre-defined typical mission as defined in the database. Alternatively, they vary specific mission characteristics to suit individual requirements. Subsequently, users enter UAV parameters and equipment details. Based on user requirements, the simulation will then run for one mission only, several missions, or simulate the whole product life-cycle. Output will be in simple text format for flexible data analysis.

Throughout model building, the mission scenarios will be validated and tested by real users such as the DECODE-team for SAR-missions.

D. Results

Once the model is built and validated, the research questions can be answered.

The work conducted so far already indicates that an operational simulation can improve a product early on in the design process [12]. This will be verified further using the advanced simulation model. A baseline case for an existing real UAV will be compared to a UAV optimized by the simulation for a range of mission scenarios.

The second research question will be answered while building and testing the model. It will be shown to what extend it is possible to create a generic operational model spanning a number of missions and a range of UAV-characteristics. The trade-off between model fidelity and generality will be discussed.

The extent of capturing operational information during the early design process will be defined by the input specifications required by the user. It will be discussed how much information is required by the user and how little information is sufficient to still produce useful results.

The ontology unifying various civil UAV missions will be explained.

VI. Conclusion

The lack of rigorous engineering analysis during the early design process leads to major problems with complex aerospace programmes. This research aims to improve the situation by introducing the concept of a generic operational simulation. This enables designers to test their product ideas and concepts in various operational scenarios in order to support cost and value analysis based on the environment the product will work in. Ongoing work has already shown the validity of this approach by optimizing the design of a civil Search-and-Rescue UAV developed at the University of Southampton. This work used a unique operational simulation specifically developed for Search-and-Rescue missions. The planned generic simulation will include the majority of possible civil UAV missions in order to give designers flexibility and the ability to compare designs and scenarios. It will be shown that an operational simulation can help finding the best initial concept based on rigorous analysis instead of engineering judgement and intuition. A trade-off between model fidelity and generality will be sought. A useful ontology for civil UAV missions will be developed along with best practices to capture operational information early in the design process.

Acknowledgment

This work is supported by the EPSRC Doctoral Training Centre grant no. EP/G03690X/1.

References

[1] M. Gries, “Methods for evaluating and covering the design space during early design development,” *Integration, the VLSI journal*, vol. 38, pp. 131–183, 2004.

[2] H. Raharjo, A. C. Brombacher, and M. Xie, “Dealing with subjectivity in early product design phase: a systematic approach to exploit quality function deployment potentials,” *Computer & Industrial Engineering*, vol. 55, pp. 253–278, 2008.

[3] M. Ivashkov, “Accel: a tool for supporting concept generation in the early design phase,” Ph.D. dissertation, Eindhoven University of Technology, 2004.

[4] E. S. K. Yu, “Towards modelling and reasoning support for early-phase requirements engineering,” *Requirements Engineering*, p. 226, 1997.

[5] T. H. Cox, C. J. Nagy, M. A. Skoog, I. A. Somers, and R. Warner, “Civil UAV Capability Assessment,” NASA, Tech. Rep., December 2004.

[6] B. Papadales, “Cost and business model analysis for civilian uav missions,” Moiré Incorporated, Tech. Rep., 2004.

[7] P. D. Collopy and P. Hollingsworth, “Value-driven design,” in *9th AIAA Aviation Technology, Integration and Operations Conference (ATIO)*, no. 2009-7099, AIAA. Hilton Head, South Carolina: AIAA, 21-23 September 2009.

[8] J. Cheung, J. Scanlan, J. Wong, J. Forrester, H. Eres, P. Collopy, P. Hollingsworth, S. Wiseall, and S. Briceno, “Application of value-driven design to commercial aero-engine systems,” in *10th AIAA Aviation Technology, Integration and Operations Conference*, no. AIAA 2010-9058. Fort Worth, Texas: AIAA, 13-15 September 2010.

[9] M. Andersson and G. Olsson, “A simulation based decision support approach for operational capacity planning in a customer order driven assembly line,” in *Proceedings of the 1998 Winter Simulation Conference*, D. J. Medeiros, E. F. Watson, J. S. Carson, and M. S. Manivannan, Eds., 1998, pp. 935–941.

[10] J. Demitz, C. Hübschen, and C. Albrecht, *Timetable planning and information quality*. WIT Press, 2010, vol. 1, ch. A, pp. 11–25.

[11] V. J. Winstead, “Method and apparatus for determining the operational energy cost of a hybrid vehicle,” US Patent 6 335 610, January 1, 2002.

[12] B. Schumann, J. Scanlan, and K. Takeda, “Evaluating design decisions in real-time using operations modelling,” in *Air Transport and Operations Symposium 2011 (ATOS)*, R. Curran and S. C. Santema, Eds. Delft University of Technology, March 2011.

[13] *AnyLogic*, XJ Technologies Software, 1992. [Online]. Available: www.xjtek.com

[14] K. Herrick, “Development of the unmanned aerial vehicle market: Forecasts and trends,” *Air & Space Europe*, vol. 2, no. 2, pp. 25–27, 2000.

Traceability Handling in Model-based Prediction of System Quality

Aida Omerovic*† and Ketil Stølen*†
*SINTEF ICT, Pb. 124, 0314 Oslo, Norway
†University of Oslo, Department of Informatics, Pb. 1080, 0316 Oslo, Norway
Email: {aida.omerovic,ketil.stolen}@sintef.no

***Abstract*—Our earlier research indicated the feasibility of the PREDIQT method for model-based prediction of impacts of architectural design changes, on the different quality characteristics of a system. The PREDIQT method develops and makes use of a multi-layer model structure, called prediction models. Usefulness of the prediction models requires a structured documentation of both the relations between the prediction models and the rationale and assumptions made during the model development. This structured documentation is what we refer to as trace-link information. In this paper, we propose a traceability scheme for PREDIQT, and an implementation of it in the form of a prototype tool which can be used to define, document, search for and represent the trace-links needed. The solution is applied on prediction models from an earlier PREDIQT-based analysis of a real-life system. Based on a set of success criteria, we argue that our traceability approach is useful and practically scalable in the PREDIQT context.**

***Keywords*-traceability; system quality prediction; modeling; architectural design; change impact analysis; simulation.**

I. Introduction

We have developed and tried out the PREDIQT method [1] [2] aimed for predicting impacts of architectural design changes on system quality characteristics and their trade-offs. Examples of quality characteristics include availability, scalability, security and reliability.

Important preconditions for model-based prediction are correctness and proper usage of the prediction models. The process of the PREDIQT method guides the development and use of the prediction models, but the correctness of the prediction models and the way they are applied are also highly dependent on the creative effort of the analyst and his/her helpers. In order to provide additional help and guidance to the analyst, we propose in this paper a traceability approach for documenting and retrieving the rationale and assumptions made during the model development, as well as the dependencies between the elements of the prediction models.

The approach is defined by a traceability scheme, which is basically a feature diagram specifying capabilities of the solution and a meta-model for the trace-link information. A prototype tool is implemented in the form of a relational database with user interfaces which can be employed to define, document, search for and represent the trace-links needed. The solution is illustrated on prediction models from an earlier PREDIQT-based analysis conducted on a real-life system [3].

The paper is organized as follows: Section II provides background on traceability. The challenge of traceability handling in the context of the PREDIQT method is characterized in Section III. Our traceability handling approach is presented in Section IV. Section V illustrates the approach on an example. Section VI argues for completeness and practicability of the approach, by evaluating it with respect to the success criteria. Section VII substantiates why our approach, given the success criteria outlined in Section III, is preferred among the alternative traceability approaches. The concluding remarks and future work are presented in Section VIII.

A full technical report [4] is available and includes: 1) an outline of the PREDIQT method, 2) guidelines for application of the prediction models which the success criteria for our traceability approach are deduced from, and 3) further details on traceability in PREDIQT.

II. Background on traceability

IEEE [5] provides two definitions of traceability:

1) Traceability is the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another; for example, the degree to which the requirements and design of a given software component match.
2) Traceability is the degree to which each element in a software development product establishes its reason for existing.

Traceability research and practice are most established in fields such as requirements engineering and model-driven engineering (MDE). Knethen and Paech [6] argue: “Dependency analysis approaches provide a fine-grained impact analysis but can not be applied to determine the impact of a required change on the overall software system. An imprecise impact analysis results in an imprecise estimate of costs and increases the effort that is necessary to implement a required change because precise relationships have to be identified during changing. This is cost intensive and error prone because analyzing the software documents requires detailed understanding of the software documents and the relationships between them." Aizenbud-Reshef et al. [7] furthermore state: "The extent of traceability practice is viewed as a measure of system quality and process maturity and is mandated by many standards" and "With complete traceability, more accurate costs and schedules of changes can be determined, rather than depending on the programmer to know all the areas that will be affected by these changes".

IEEE [5] defines a trace as "A relationship between two or more products of the development process." According to the OED [8], however, a trace is defined more generally as a "(possibly) non-material indication or evidence showing what has existed or happened". As argued by [9]: "If a developer works on an artifact, he leaves traces. The software configuration management system records who has worked on the artifact, when that person has worked on it, and some systems also record which parts of the artifacts have been changed. But beyond this basic information, the changes themselves also reflect the developer's thoughts and ideas, the thoughts and ideas of other stakeholders he may have talked to, information contained in other artifacts, and the transformation process that produced the artifact out of these inputs. These influences can also be considered as traces, even though they are usually not recorded by software configuration management systems."

A traceability link is a relation that is used to interrelate artifacts (e.g., by causality, content, etc.) [9]. In the context of requirements traceability, [9] argues that "a trace can in part be documented as a set of meta-data of an artifact (such as creation and modification dates, creator, modifier, and version history), and in part as relationships documenting the influence of a set of stakeholders and artifacts on an artifact. Particularly those relationships are a vital concept of traceability, and they are often referred to as traceability links. Traceability links document the various dependencies, influences, causalities, etc. that exist between the artifacts. A traceability link can be unidirectional (such as depends-on) or bidirectional (such as alternative-for). The direction of a link, however, only serves as an indication of order in time or causality. It does not constrain its (technical) navigability, so traceability links can always be followed in both directions".

In addition to the different definitions, there is no commonly agreed basic classification [9]. A taxonomy of the main concepts within traceability is suggested by [6].

An overview of the current state of traceability research and practice in requirements engineering and model-driven development is provided by [9], based on an extensive literature survey. Another survey [10] discusses the state-of-the-art in traceability approaches in MDE and assesses them with respect to five evaluation criteria: representation, mapping, scalability, change impact analysis and tool support. Moreover, Spanoudakis and Zisman [11] present a roadmap of research and practices related to software traceability.

Traces can exist between both model- and non-model artifacts. The means and measures applied for obtaining traceability are defined by so-called traceability schemes. A traceability scheme is driven by the planned use of the traces. The traceability scheme determines for which artifacts and up to which level of detail traces can be recorded [9]. A traceability scheme thus defines the constraints needed to guide the recording of traces, and answers the core questions: what, who, where, how, when and why. Additionally, there is tacit knowledge (such as why), which is difficult to capture and to document. A traceability scheme helps in this process of recording traces and making them persistent.

According to Wieringa [12], representations and visualizations of traces can be categorized into matrices, cross-references, and graph-based representations. As elaborated by Wieringa, the links, the content of the one artifact, and other information associated with a cross reference, is usually displayed at the same time. This is however not the case with traceability matrices. So, compared to traceability matrices, the user is (in the case of cross-references) shown more local information at the cost of being shown fewer (global) links. As models are the central element in MDE, graph-based representations are the norm. A graph can be transformed to a cross-reference. Regarding the notation, there is, however, no common agreement or standard, mostly because the variety and informality of different artifacts is not suitable for a simple, yet precise notation.

Traceability activities are generally not dependent on any particular software process model. Knethen and Paech [6] argue that the existing traceability approaches do not give much process support. They specify four steps of traceability process: 1) define entities and relationships, 2) capture traces, 3) extract and represent traces, and 4) maintain traces. Similarly, Winkler and Pilgrim [9] state that traceability and its supporting activities are currently not standardized. They classify the activities when working with traces into: 1) planning for traceability, 2) recording traces, 3) using traces, and 4) maintaining traces.

Trace models are usually stored as separate models, and links to the elements are (technically) unidirectional in order to keep the connected models or artifacts independent. Alternatively, models can contain the trace-links themselves and links can be defined as bidirectional. While embedded trace-links pollute the models, navigation is much easier [9]. Thus, we distinguish between external and internal storage, respectively. Anquetil at al. [13] argue: "Keeping link information separated from the artifacts is clearly better; however it needs to identify uniquely each artifact, even fined-grained artifacts. Much of the recent research has focused on finding means to automate the creation and maintenance of trace information. Text mining, information retrieval and analysis of trace links techniques have been successfully applied. An important challenge is to maintain links consistency while artifacts are evolving. In this case, the main difficulty comes from the manually created links, but scalability of automatic solution is also an issue."

Various tools are used to set and maintain traces. Surveys of the tools available are provided by [6], [9], [11] and [7]. Bohner and Arnold [14] found that the granularity of documentation entities managed by current traceability tools is typically somewhat coarse for an accurate impact analysis.

III. The challenge

Three interrelated sets of models are developed during the process of the PREDIQT method: Design Model which specifies system architecture, Quality Model which specifies the system quality notions, and Dependency Views (DVs) which represent the interrelationship between the system quality and the architectural design. The PREDIQT process consists of three overall phases: *Target modeling*, *Verification of prediction models*, and *Application of prediction models*.

Figure 1 provides an overview of the elements of the prediction models, expressed as a UML [15] class diagram. A Quality Model is a set of tree-like structures which clearly specify the system-relevant quality notions, by defining and decomposing the meaning of the system-relevant quality terminology. Each tree is dedicated to a target system-relevant quality characteristic. Each quality characteristic may be decomposed into quality sub-characteristics, which in turn may be decomposed into a set of quality indicators. As indicated by the relationship of type aggregation, specific sub-characteristics and indicators can appear in several Quality Model trees dedicated to the different quality characteristics. Each element of a Quality Model is assigned a quantitative normalized metric and an interpretation (qualitative meaning of the element), both specific for the target system. A Design Model represents the relevant aspects of the system architecture, such as for example process, dataflow, structure and rules.

A DV is a weighted dependency tree dedicated to a specific quality characteristic defined through the Quality Model. As indicated by the attributes of the Class *Node*, the nodes of a DV are assigned a name and a QCF (*Quality Characteristic Fulfillment*). A QCF is value of the degree of fulfillment of the quality characteristic, with respect to what is represented by the node. The degree of fulfillment is defined by the metric (of the quality characteristic) provided in the Quality Model. Thus, a complete prediction model has as many DVs as the quality characteristics defined in the Quality Model. Additionally, as indicated by the *Semantic* dependency relationship, semantics of both the structure and the weights of a DV are given by the definitions of the quality characteristics, as specified in the Quality Model. A DV node may be based on a Design Model element, as indicated by the *Based on* dependency relationship. As indicated by the self-reference on the *Node* class, one node may be decomposed into children nodes. Directed arcs express dependency with respect to quality characteristic by relating each parent node to its immediate children nodes, thus forming a tree structure. Each arc in a DV is assigned an EI (*Estimated Impact*), which is a normalized value of degree of dependence of a parent node, on the immediate child node. The values on the nodes and the arcs are referred to as parameter estimates. We distinguish between prior and inferred parameter estimates. The former ones are, in the form of empirical input, provided on leaf nodes and all arcs, while the latter ones are deduced using the DV propagation model for PREDIQT [3].

The intended application of the prediction models does not assume implementation of change on the target system, but only simulation of effects of the independent architectural design changes quality of the system (in its currently modelled state). Since the simulation is only performed on the target system in its current state and the changes are simulated independently (rather than incrementally), versioning of the prediction models in not necessary. Hence, maintenance of both prediction models and trace information is beyond the scope of PREDIQT.

Trace-link information can be overly detailed and extensive while the solution needed in a PREDIQT context has to be applicable in a practical real-life setting within the limited resources allocated for a PREDIQT-based analysis. Therefore, the traceability approach should provide sufficient breadth and accuracy for documenting, retrieving and representing of the trace-links, while at the same time being practically applicable in terms of comprehensibility and scalability. The right balance between the completeness and accuracy of the trace information on the one side, and practical usability of the approach on the other side, is what characterizes the main challenge in proposing the appropriate solution for traceability handling in PREDIQT. Therefore, the trace-link creation efforts have to be concentrated on the traces necessary during the application of the prediction models.

It is, as argued by [9], an open issue to match trace usage and traceability schemes, and to provide guidance to limit and fit traceability schemes in a such way that they match a projects required usage scenarios for traces. One of the most urgent questions is which requirements a single scenario imposes on the other activities (in particular planning and recording) in the traceability process.

Moreover, it is argued by Aizenbud-Reshef et al. [7] that the lack of guidance as to what link information should be produced and the fact that those who use traceability are commonly not those producing it, also diminishes the motivation of those who create and maintain traceability information. In order to avoid this trap, we used the PREDIQT guidelines [4] for the analyst as a starting point, for deriving the specific needs for traceability support. The guidelines are based on the authors' experiences from industrial trials of PREDIQT [3] [2]. As such, the guidelines are not exhaustive but serve as an aid towards a more structured process of applying the prediction models and accommodating the trace

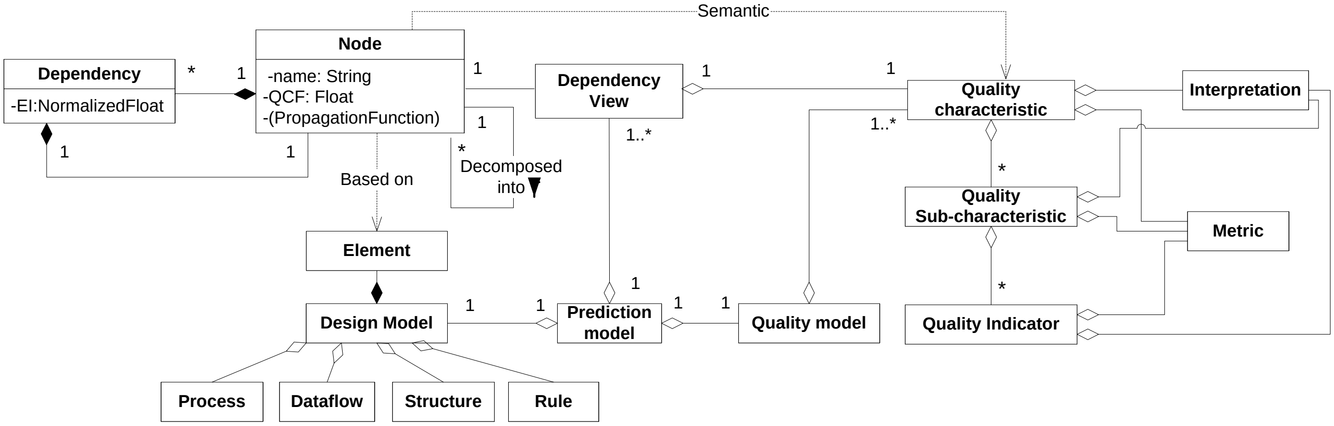


Figure 1. An overview of the elements of the prediction models, expressed as a UML class diagram

information during the model development, based on the needs of the "Application of prediction models"-phase.

The specific needs for traceability support in PREDIQT are summarized below:

1) There is need for the following kinds of trace-links:
   - links between the Design Model elements
   - links from the Design Model elements to DV elements
   - links from DV elements to Quality Model elements (i.e. traces to the relevant quality indicators and rationale for the prior estimates)
   - links to external information sources (documents, measurement, domain experts) used during the development of DV structure and estimation of the parameters
   - links to rationale and assumptions for: Design Model elements, the semantics of the DV elements, as well as structure and prior parameter estimates of the DVs
2) The traceability approach should have facilities for both searching with model types and model elements as input parameters, as well as for reporting linked elements and the link properties
3) The traceability approach should be flexible with respect to granularity of trace information
4) The traceability approach should be practically applicable on real-life applications of PREDIQT

These needs are in the sequel referred to as the **success criteria** for the traceability approach in PREDIQT.

IV. Our solution

This section starts by presenting our traceability scheme for PREDIQT. Then, a prototype tool for trace-link management, implementing the needs specified through the traceability scheme, is presented.

A. Traceability scheme

We propose a traceability scheme in the form of a meta-model for trace-link information and a feature diagram for capabilities of the solution. The types of the trace-links and the types of the traceable elements are directly extracted from Success Criterion 1 and represented through a meta-model shown by Figure 2. The *Element* abstract class represents a generalization of a traceable element. The *Element* abstract class is specialized into the five kinds of traceable elements: *Design Model Element*, *DV Element*, *Quality Model Element*, *External Information Source*, and *Rationale and Assumptions*. Similarly, the *Trace Link* abstract class represents a generalization of a trace-link and may be assigned a rationale for the trace-link. The *Trace Link* abstract class is specialized into the six kinds of trace-links.

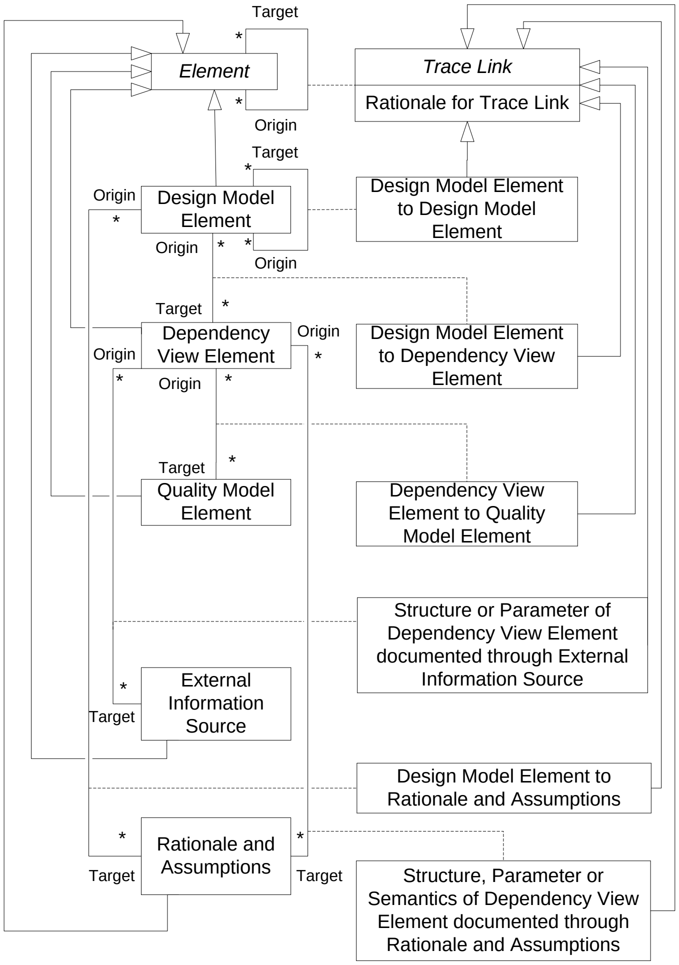

Target
Origin
Element
Trace Link
Rationale for Trace Link
Design Model Element
Design Model Element to Design Model Element
Dependency View Element
Design Model Element to Dependency View Element
Quality Model Element
Dependency View Element to Quality Model Element
Structure or Parameter of Dependency View Element documented through External Information Source
External Information Source
Design Model Element to Rationale and Assumptions
Rationale and Assumptions
Structure, Parameter or Semantics of Dependency View Element documented through Rationale and Assumptions
*

Figure 2. A meta model for trace-link information, expressed as a UML class diagram

Pairs of certain kinds of traceable elements form binary relations in the form of unidirectional trace-links. Such relations are represented by the UML-specific notations called association classes (a class connected by a dotted line to a link which connects two classes). For example, trace-links of type *Design Model Element to Design Model Element* may be formed from a *Design Model Element* to a *Dependency View Element*. The direction of the link is annotated by the origin (the traceable element that the trace-link goes from) and the target (the traceable element that the trace-link goes to). Since only distinct pairs (single instances) of the traceable elements (of the kinds involved in the respective trace-links defined in Figure 2) can be involved in the associated specific kinds of trace-links, uniqueness (property of UML association classes) is present in the defined trace-links. Due to the binary relations (arity of value 2) in the defined trace-links between the traceable elements, only two elements can be involved in any trace-link. Furthermore, multiplicity of all the traceable elements involved in the trace-links defined is of type “many”, since an element can participate in multiple associations (given they are defined by the meta-model and unique).

The main capabilities needed are represented through a feature diagram [9] shown by Figure 3. Storage of trace-links may be internal or external, relative to the prediction models. A traceable element may be of type prediction model element (see Figure 1) or non-model element. Reporting and searching functionality has to be supported. Trace-link info has to include link direction, link meta-data (e.g. date, creator, strength) and cardinality (note that all links are binary, but a single element can be origin or target for more than one trace-link). Typing at the origin and the target ends of a trace-link as well as documenting rationale for trace-link, are optional.

B. Prototype traceability tool

We have developed a prototype tool in the form of a database application with user interfaces, on the top of MS Access [16]. The prototype tool includes a structure of tables for organizing the trace information, queries for retrieval of the trace info, a menu for managing work flow, forms for populating trace-link information, and facilities for reporting trace-links. A screen shot of the entity-relationship (ER) diagram of the trace-link database is shown by Figure 4. The ER diagram is normalized, which means that the data are organized with minimal needs for repeating the entries in the tables. Consistency checks are performed on the referenced fields. The data structure itself (represented by the ER diagram) does not cover all the constraints imposed by the meta-model (shown by Figure 2). However, constraints on queries and forms as well as macros can be added in order to fully implement the logic, such as for example which element types can be related to which trace-link types.

The five traceable element types defined by Figure 2 and their properties (name of creator, date, assumption and comment), are listed in Table *TraceableElementType*. Similarly, the six trace-link types defined by Figure 2 and their properties (scope, date, creator and comment), are listed in Table *TraceLinkType*. Table *TraceableElement* specifies the concrete instances of the traceable elements, and assigns properties (such as the pre-defined element type, hyperlink, creator, date, etc.) to each one of them. Since primary key attribute in Table *TraceableElementType* is foreign key in Table *TraceableElement*, multiplicity between the two respective tables is one-to-many.

Most of the properties are optional, and deduced based on: 1) the core questions to be answered by traceability scheme [9] and 2) the traceability needs for using guidelines for application of prediction models (specified in [4]). The three Tables *TargetElements*, *OriginElements* and *TraceLink* together specify the concrete instances of trace-links. Each link is binary, and directed from a concrete pre-defined traceable element – the origin element specified in Table *OriginElements*, to a concrete pre-defined traceable element – the target element specified in Table *TargetElements*. The trace-link itself (between the origin and the target element) and its properties (such as pre-defined trace-link type) are specified in Table *TraceLink*. Attribute *TraceLinkName* (associated with a unique *TraceLinkId* value) connects the three tables *TraceLink*, *OriginElements* and *TargetElements* when representing a single trace-link instance, thus forming a cross-product when relating the three tables. The MS Access environment performs reference checks on the cross products, as well as on the values of the foreign key attributes. Target elements and origin elements participating in a trace-link, are instances of traceable elements defined in Table *TraceableElement*. They are connected through the Attribute *ElementId* (displayed as *ElementName* in the tables where it has the role of foreign key). Thus, multiplicity between Table *TraceableElement* and Table *TargetElements*, as well as between Table *TraceableElement* and Table *OriginElements*, is one-to-many. Similarly, since primary key attribute in Table *TraceLinkType* is foreign key in Table *TraceLink*, multiplicity between the two respective tables is one-to-many.

A screen shot of the start menu is shown by Figure 5. The sequence of the buttons represents a typical sequence of actions of an end-user (the analyst), in the context of defining, documenting and using the trace-links. The basic definition of the types of the traceable elements and the trace-links are provided first. Then, concrete traceable elements are documented, before defining specific instances of the trace-links and their associated specific origin and target elements, involved in the binary trace-link relations. Finally, reports can be obtained, based on search parameters such as for example model types, model elements, or trace-link types.

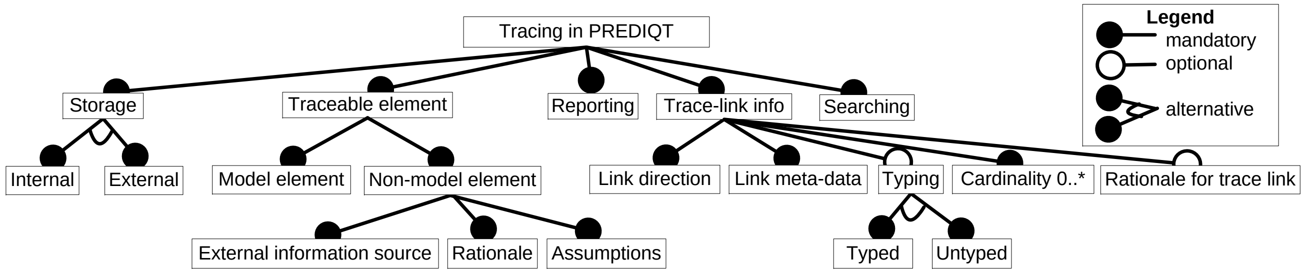


Figure 3. Main capabilities of the traceability approach

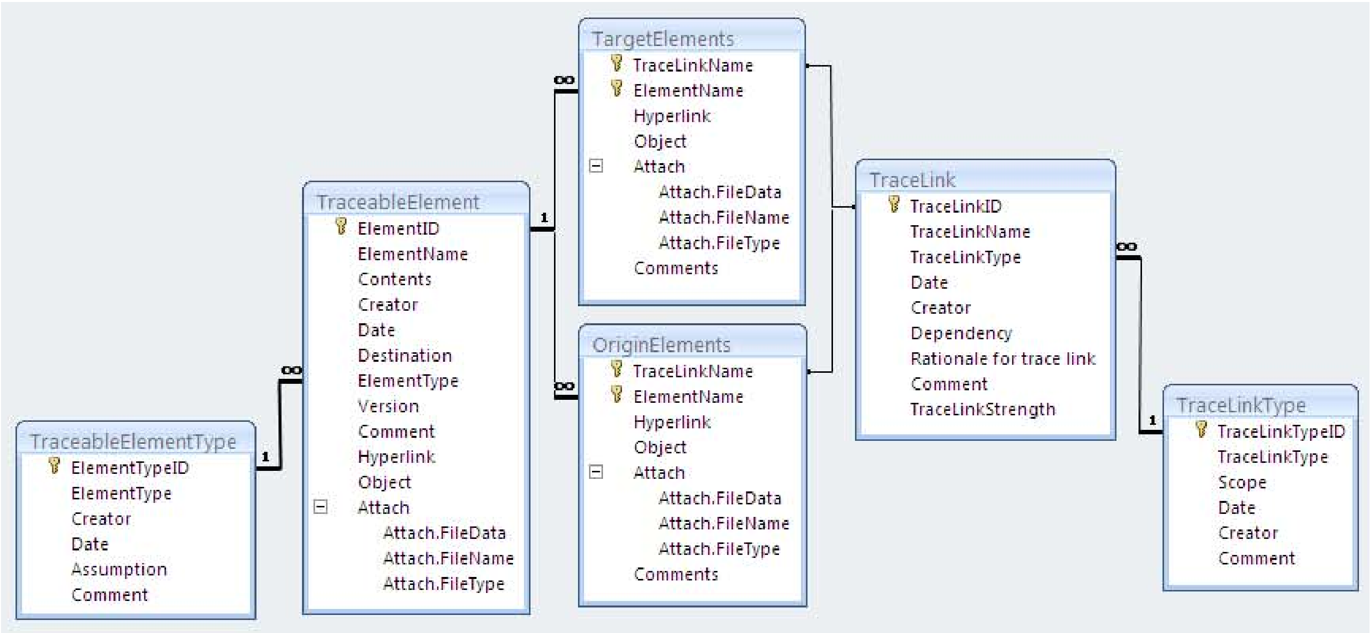


Figure 4. Entity-relationship diagram of the trace-link database of the prototype traceability tool

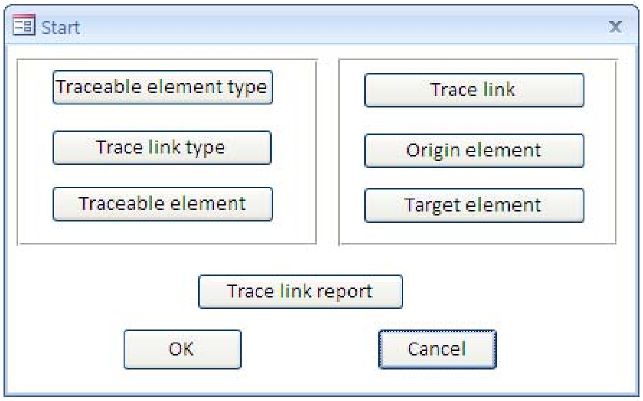


Figure 5. A screen shot of the start menu of the prototype traceability tool

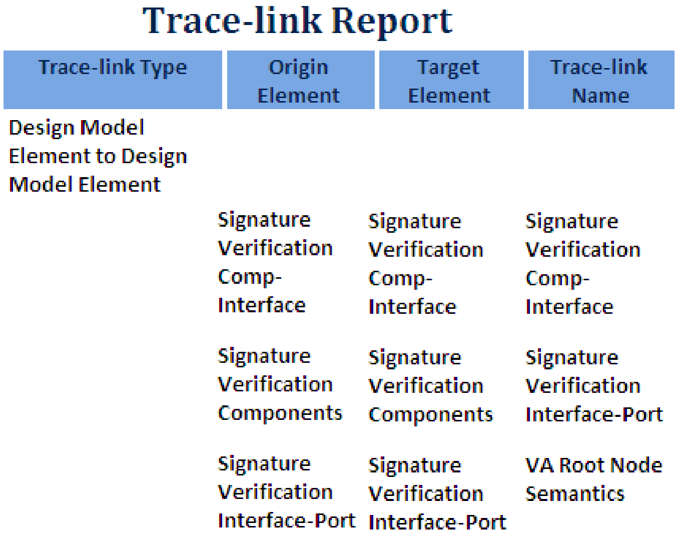
V. Applying the solution on an example

This section exemplifies the application of our solution for managing traces in the context of prediction models earlier developed and applied during a PREDIQT-based analysis [3] conducted on a real-life system.

The trace-link information was documented in the prototype tool, in relation to the model development. The trace-links were applied during change application, according to the guidelines for application of prediction models (specified in [4]). We present the experiences obtained, while the process of documentation of the trace-links is beyond the scope of this paper.

The prediction models involved are the ones related to "Split signature verification component into two redundant components, with load balancing", corresponding to Change 1 in [3]. Three Design Model diagrams were affected, and one, two and one model element on each, respectively. We have tried out the prototype traceability tool on the Design Model diagrams involved, as well as Availability (which was one of the three quality characteristics analyzed) related Quality Model diagrams and DV. Documentation of the trace-links involved within the Availability quality characteristic (as defined by the Quality Model) scope, took approximately three hours. Most of the time was spent on actually typing the names of the traceable elements and the trace-links.

18 instances of traceable elements were registered in the database during the trial: seven Quality Model elements, four DV elements, four Design Model elements and three elements of type "Rationale and Assumptions". 12 trace-links were recorded: three trace-links of type "Design Model Element to Design Model Element", three trace-links of type "Design Model Element to DV Element", one trace-link of type "Design Model Element to Rationale and Assumptions", three trace-links of type "DV Element to Quality Model Element", and two trace-links of type "Structure, Parameter or Semantics of DV Element Documented through Rationale and Assumptions", were documented.



Trace-link Report

Trace-link Type	Origin Element	Target Element	Trace-link Name
Design Model Element to Design Model Element			
	Signature Verification Comp-Interface	Signature Verification Comp-Interface	Signature Verification Comp-Interface
	Signature Verification Components	Signature Verification Components	Signature Verification Interface-Port
	Signature Verification Interface-Port	Signature Verification Interface-Port	VA Root Node Semantics

Figure 6. A screen shot of an extract of a trace-link report from the prototype traceability tool

An extract of a screen shot of a trace-link report (obtained from the prototype tool) is shown by Figure 6. The report included: three out of three needed (i.e., actually existing, regardless if they are recorded in the trace-link database) “Design Model Element to Design Model Element” links, three out of four needed “Design Model Element to DV Element” links, one out of one needed “Design Model Element to Rationale and Assumptions” link, three out of six needed “DV Element to Quality Model Element” links and one out of one needed “Structure, Parameter or Semantics of DV Element Documented through Rationale and Assumptions” link.

Best effort was made to document the appropriate trace-links without taking into consideration any knowledge of exactly which of them would be used when applying the change. The use of the trace-links along with the application of change on the prediction models took totally 20 minutes and resulted in the same predictions (change propagation paths and values of QCF estimates on the Availability DV), as in the original case study [3]. Without the guidelines and the trace-link report, the change application would have taken approximately double time for the same user.

All documented trace-links were relevant and used during the application of the change, and about 73% of the relevant trace-links could be retrieved from the prototype tool. Considering however the importance and the role of the retrievable trace-links, the percentage should increase considerably.

Although hyperlinks are included as meta-data in the user interface for element registration, an improved solution should include interfaces for automatic import of the element names from the prediction models, as well as user interfaces for easy (graphical) trace-link generations between the existing elements. This would also aid verification of the element names.

VI. Why our solution is a good one

This section argues that the approach presented above fulfills the success criteria specified in Section III.

A. *Success Criterion 1*

The traceability scheme and the prototype tool capture the kinds of trace-links and traceable elements, specified in the Success Criterion 1. The types of trace-links and traceable elements as well as their properties, are specified in dedicated tables in the database of the prototype tool. This allows constraining the types of the trace-links and the types of the traceable elements to only the ones defined, or extending their number or definitions, if needed. The trace-links in the prototype tool are binary and unidirectional, as required by the traceability scheme. Macros and constraints can be added in the tool, to implement any additional logic regarding trace-links, traceable elements, or their respective type definitions and relations. The data properties (e.g. date, hyperlink or creator) required by the user interface, allow full traceability of the data registered in the database of the prototype tool.

B. *Success Criterion 2*

Searching based on user input, selectable values from a list of pre-defined parameters, or comparison of one or more database fields, are relatively simple and fully supported based on queries in MS Access. Customized reports can be produced with results of any query and show any information registered in the database. The report, an extract of which is presented in Section V, is based on a query of all documented trace-links and the related elements.

C. *Success Criterion 3*

The text-based fields for documenting the concrete instances of the traceable elements and the trace-links, allow level of detail selectable by the user. Only a subset of fields is mandatory for providing the necessary trace-link data. The optional fields in the tables can be used for providing additional information such as for example rationale, comments, links to external information sources, attachments, strength or dependency. There are no restrictions as to what can be considered as a traceable element, as long at it belongs to one of the element types defined by Figure 2. Similarly, there are no restrictions as to what can be considered as a trace-link, as long at it belongs to one of the trace-link types defined by Figure 2. The amount of information provided regarding the naming and the meta-data, are selectable by the user.

D. *Success Criterion 4*

Given the realism of the prediction models involved in the example, the size and complexity of the target system they address, the representativeness of the change applied on them, the simplicity of the prototype tool with respect to both the user interfaces and the notions involved, as

ISBN: 978-1-61208-169-4

well as the time spent on documenting the trace-links and using them, the application of the approach presented in Section V indicates the applicability of our solution on real-life applications of PREDIQT, with limited resources and by an average user (in the role of the analyst).

The predictions (change propagation paths and values of QCF estimates) we obtained during the application of our solution on the example were same as the ones from the original case study [3] (performed in year 2008) which the models stem from. Although the same analyst has been involved in both, the results suggest that other users should, by following PREDIQT guidelines and applying the prototype traceability tool, obtain similar results.

The time spent is to some degree individual and depends on the understanding of the target system, the models and the PREDIQT method. It is unknown if the predictions would have been the same (as in the original case study) for another user. We do however consider the models and the change applied during the application of the solution, to be representative due to their origins from a major real-life system. Still, practical applicability of our solution will be subject to future empirical evaluations.

VII. Why other approaches are not better in this context

This section evaluates the feasibility of other traceability approaches in the PREDIQT context. Based on our literature review and the results of the evaluation by Galvao and Goknil [10], we argue why the alternative traceability approaches do not perform sufficiently on one or more of the success criteria specified in Section III.

Almeida et al. [17] propose an approach aimed at simplifying the management of relationships between requirements and various design artifacts. A framework which serves as a basis for tracing requirements, assessing the quality of model transformation specifications, meta-models, models and realizations, is proposed. They use traceability cross-tables for representing relationships between application requirements and models. Cross-tables are also applied for considering different model granularities and identification of conforming transformation specifications. The approach does not provide sufficient support for intra-model mapping, thus failing on our Success Criterion 1. Moreover, possibility of representing the various types of trace-links and traceable elements is unclear, although different visualizations on a cross-table are suggested. Tool support is not available, which limits applicability of the approach in a practical setting. Searching and reporting facilities are not available. Thus, it fails on our Success Criteria 1, 2 and 4.

Event-based Traceability (EBT) is another requirements-driven traceability approach aimed at automating trace-link generation and maintenance. Cleland-Huang, Chang and Christensen [18] present a study which uses EBT for managing evolutionary change. They link requirements and other traceable elements, such as design models, through publish-subscribe relationships. As outlined by [10], "Instead of establishing direct and tight coupled links between requirements and dependent entities, links are established through an event service. First, all artefacts are registered to the event server by their subscriber manager. The requirements manager uses its event recognition algorithm to handle the updates in the requirements document and to publish these changes as event to the event server. The event server manages some links between the requirement and its dependent artefacts by using some information retrieval algorithms." The notification of events carries structural and semantic information concerning a change context. Scalability in a practical setting is the main issue, due to performance limitation of the EBT server [10]. Moreover, the approach does not provide sufficient support for intra-model mapping. Thus, it fails on our Success Criteria 1 and 4.

Cleland-Huang et al. [19] propose Goal Centric Traceability (GCT) approach for managing the impact of change upon the non-functional requirements of a software system. Softgoal Interdependency Graph (SIG) is used to model non-functional requirements and their dependencies. Additionally, a traceability matrix is constructed to relate SIG elements to classes. The main weakness of the approach is the limited tool support, which requires manual work. This limits both scalability in a practical setting and searching support (thus failing on our Success Criteria 4 and 2, respectively). It is unclear to what degree granularity of the approach would suffice the needs of PREDIQT.

Cleland-Huang and Schmelzer [20] propose another requirements-driven traceability approach that builds on EBT. The approach involves a different process for dynamically tracing non-functional requirements to design patterns. Although more fine grained than EBT, there is no evidence that the method can be applied with success in a practical real-life setting (required through our Success Criterion 4). Searching and reporting facilities (as required through our Success Criterion 2) are not provided.

Many traceability approaches address trace maintenance. Cleland-Huang, Chang and Ge [21] identify the various change events that occur during requirements evolution and describe an algorithm to support their automated recognition through the monitoring of more primitive actions made by a user upon a requirements set. Mäder and Gotel [22] propose an approach to recognize changes to structural UML models that impact existing traceability relations and, based on that knowledge, provide a mix of automated and semi-automated strategies to update the relations. Both approaches focus on trace maintenance, which is as argued in Section III, not among the traceability needs in PREDIQT.

Ramesh and Jarke [23] propose another requirements-driven traceability approach where reference models are used to represent different levels of traceability information and links. The granularity of the representation of traces depends on the expectations of the stakeholders [10]. The reference models can be implemented in distinct ways when managing the traceability information. As reported by [10], "The reference models may be scalable due to their possible use for traceability activities in different complexity levels. Therefore, it is unclear whether this approach lacks scalability with respect to tool support for large-scale projects or not." In PREDIQT context, the reference models are too broad, their focus is on requirements traceability, and tool support is not sufficient with respect to searching and reporting (our Success Criterion 2).

We could however have tried to use parts of the reference models by Ramesh and Jarke [23] and provide tool support based on them. This is done by [24] in the context of product and service families. The authors discuss a knowledge management system, which is based on the traceability framework by Ramesh and Jarke [23]. The system captures the various design decisions associated with service family development. The system also traces commonality and variability in customer requirements to their corresponding design artifacts. The tool support has graphical interfaces for documenting decisions. The trace and design decision capture is illustrated using sample scenarios from a case study. We have however not been able to obtain the tool, in order to try it out in our context.

A modeling approach by Egyed [25] represents traceability information in a graph structure called a footprint graph. Generated traces can relate model elements with other models, test scenarios or classes [10]. Galvao and Goknil [10] report on promising scalability of the approach. It is however unclear to what degree the tool support fulfills our success criterion regarding searching and reporting, since semantic information on trace-links and traceable elements is limited.

Aizenbud-Reshef et al. [26] outline an operational semantics of traceability relationships that capture and represent traceability information by using a set of semantic properties, composed of events, conditions and actions [10]. Galvao and Goknil [10] state: the approach does not provide sufficient support for intra-model mapping; a practical application of the approach is not presented; tool support is not provided; however, it may be scalable since it is associated with the UML. Hence, it fails on our Success Criteria 1 and 2.

Some approaches [27] [28] [29] that use model transformations can be considered as a mechanism to generate trace-links. Tool support with transformation functionalities is in focus, while empirical evidence of comprehensibility of the approaches in a practical setting, is missing. The publications we have retrieved do not report sufficiently on whether these approaches would offer the searching facilities, the granularity of trace information, and practical applicability needed for use in PREDIQT context (that is, by an analyst who is not an expert in the tools provided).

VIII. Conclusion and future work

Our earlier research indicates the feasibility of the PREDIQT method for model-based prediction of impacts of architectural design changes on system quality. The PREDIQT method produces and applies a multi-layer model structure, called prediction models, which represent system design, system quality and the interrelationship between the two.

Based on the success criteria for a traceability approach in the PREDIQT context, we put forward a traceability scheme. Based on this, a prototype tool which can be used to define, document, search for and represent the trace-links needed, is developed. We have argued that our solution offers a useful and practically applicable support for traceability in the PREDIQT context.

Performing an analysis of factors such as cost, risk, and benefit and following the paradigm of value-based software-engineering, would be relevant in order to stress the effort on the important trace-links. As argued by [9], if the value-based paradigm is applied to traceability, cost, benefit, and risk will have to be determined separately for each trace according to if, when, and to what level of detail it will be needed later. This leads to more important artifacts having higher-quality traceability. There is a trade-off between the semantically accurate techniques on the one hand and cost-efficient but less detailed approaches on the other hand. Finding an optimal compromise is still a research challenge. Our solution proposes a feasible approach, while finding the optimal one is subject to further research.

Further empirical evaluation of our solution is also necessary to test its feasibility on different analysts as well as its practical applicability in the various domains which PREDIQT is applied on. Future work should also include standard interfaces and procedures for updating the traceable elements from the prediction models into our prototype traceability tool. As model application phase of PREDIQT dictates which trace-link information is needed and how it should be used, the current PREDIQT guidelines focus on the application of the prediction models. However, since the group of recorders and the group of users of traces may be distinct, structured guidelines for recording the traces during the model development should also be developed as a part of the future work.

Acknowledgment

This work has been conducted as a part of the DIGIT (180052/S10) project funded by the Research Council of Norway, as well as a part of the NESSoS network of excellence funded by the European Commission within the 7th Framework Programme.

References

[1] A. Omerovic, A. Andresen, H. Grindheim, P. Myrseth, A. Refsdal, K. Stølen, and J. Ølnes, "A Feasibility Study

in Model Based Prediction of Impact of Changes on System Quality," in *International Symposium on Engineering Secure Software and Systems*, vol. LNCS 5965. Springer, 2010, pp. 231–240.

[2] A. Omerovic, B. Solhaug, and K. Stølen, "Evaluation of Experiences from Applying the PREDIQT Method in an Industrial Case Study," in *Fifth IEEE International Conference on Secure Software Integration and Reliability Improvement*. IEEE, 2011.

[3] A. Omerovic, A. Andresen, H. Grindheim, P. Myrseth, A. Refsdal, K. Stølen, and J. Ølnes, "A Feasibility Study in Model Based Prediction of Impact of Changes on System Quality," SINTEF, Tech. Rep. A13339, 2010.

[4] A. Omerovic and K. Stølen, "Traceability Handling in Model-based Prediction of System Quality," SINTEF, Tech. Rep. A19348, 2011.

[5] "Standard Glossary of Software Engineering Terminology: IEEE Std.610. 12-1990," 1990.

[6] A. Knethen and B. Paech, "A Survey on Tracing Approaches in Practice and Research," Frauenhofer IESE, Tech. Rep. 095.01/E, 2002.

[7] N. Aizenbud-Reshef, B. T. Nolan, J. Rubin, and Y. Shaham-Gafni, "Model Traceability," *IBM Syst. J.*, vol. 45, no. 3, pp. 515–526, 2006.

[8] J. Simpson and E. Weiner, *Oxford English Dictionary*. Clarendon Press, 1989, vol. 18, 2nd edn.

[9] S. Winkler and J. von Pilgrim, "A survey of Traceability in Requirements Engineering and Model-driven Development," *Software and Systems Modeling*, vol. 9, no. 4, pp. 529–565, 2010.

[10] I. Galvao and A. Goknil, "Survey of Traceability Approaches in Model-Driven Engineering," in *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference*, 2007.

[11] G. Spanoudakis and A. Zisman, "Software Traceability: A Roadmap," in *Handbook of Software Engineering and Knowledge Engineering*. World Scientific Publishing, 2004, pp. 395–428.

[12] R. J. Wieringa, "An Introduction to Requirements Traceability," Faculty of Mathematics and Computer Science, Vrije Universiteit, Tech. Rep. IR-389, 1995.

[13] N. Anquetil, U. Kulesza, R. Mitschke, A. Moreira, J.-C. Royer, A. Rummler, and A. Sousa, "A Model-driven Traceability Framework for Software Product Lines," *Software and Systems Modeling*, 2009.

[14] S. Bohner and R. Arnold, *Software Change Impact Analysis*. IEEE Computer Society Press, 1996.

[15] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified Modeling Language Reference Manual*. Pearson Higher Education, 2004.

[16] "Access Help and How-to," accessed: May 19, 2011. [Online]. Available: http://office.microsoft.com/en-us/access-help/

[17] J. P. Almeida, P. v. Eck, and M.-E. Iacob, "Requirements Traceability and Transformation Conformance in Model-Driven Development," in *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference*, 2006, pp. 355–366.

[18] J. Cleland-Huang, C. K. Chang, and M. Christensen, "Event-Based Traceability for Managing Evolutionary Change," *IEEE Trans. Softw. Eng.*, vol. 29, pp. 796–810, 2003.

[19] J. Cleland-Huang, R. Settimi, O. BenKhadra, E. Berezhanskaya, and S. Christina, "Goal-centric Traceability for Managing Non-functional Requirements," in *Proceedings of the 27th international conference on Software engineering*. ACM, 2005, pp. 362–371.

[20] J. Cleland-Huang and D. Schmelzer, "Dynamically Tracing Non-Functional Requirements through Design Pattern Invariants," in *Proceedings of the 2nd International Workshop on Traceability in Emerging Forms of Software Engineering*. ACM, 2003.

[21] J. Cleland-Huang, C. K. Chang, and Y. Ge, "Supporting Event Based Traceability through High-Level Recognition of Change Events," *Computer Software and Applications Conference, Annual International*, vol. 0, p. 595, 2002.

[22] P. Mäder, O. Gotel, and I. Philippow, "Enabling Automated Traceability Maintenance through the Upkeep of Traceability Relations," in *Proceedings of the 5th European Conference on Model Driven Architecture - Foundations and Applications*. Springer-Verlag, 2009, pp. 174–189.

[23] B. Ramesh and M. Jarke, "Toward Reference Models for Requirements Traceability," *IEEE Transactions on Software Engineering*, vol. 27, no. 1, pp. 58–93, 2001.

[24] K. Mohan and B. Ramesh, "Managing Variability with Traceability in Product and Service Families," *Hawaii International Conference on System Sciences*, vol. 3, 2002.

[25] A. Egyed, "A Scenario-Driven Approach to Trace Dependency Analysis," *IEEE Transactions on Software Engineering*, vol. 29, no. 2, pp. 116–132, 2003.

[26] N. Aizenbud-Reshef, R. F. Paige, J. Rubin, Y. Shaham-Gafni, and D. S. Kolovos, "Operational Semantics for Traceability," in *Proceedings of the ECMDA Traceability Workshop, at European Conference on Model Driven Architecture*, 2005.

[27] F. Jouault, "Loosely Coupled Traceability for ATL," in *In Proceedings of the European Conference on Model Driven Architecture (ECMDA) workshop on traceability*, 2005, pp. 29–37.

[28] D. S. Kolovos, R. F. Paige, and F. Polack, "Merging Models with the Epsilon Merging Language (EML)," in *MoDELS'06*, 2006, pp. 215–229.

[29] J. Falleri, M. Huchard, and C. Nebut, "Towards a Traceability Framework for Model Transformations in Kermeta," in *Proceedings of the ECMDA Traceability Workshop, at European Conference on Model Driven Architecture*, 2006, pp. 31–40.

A Simulation-Based Innovation Forecasting Approach Combining the Bass Diffusion Model, the Discrete Choice Model and System Dynamics

An Application in the German Market for Electric Cars

Luis Antonio de Santa-Eulalia
Travail, Économie et Gestion
Téluq – Université du Québec à Montréal
Québec City, Canada
leulalia@teluq.uqam.ca

Donald Neumann
Graduate School for Advanced Manufacturing Engineering
Universität Stuttgart
Stuttgart, Germany
donald.neumann@gsame.uni-stuttgart.de

Jörg Klasen
EnBW - Energie Baden-Württemberg AG
Karlsruhe, Germany
j.klasen@enbw.com

Abstract—**This work presents a novel simulation-based forecasting approach combining concepts from the Bass Diffusion Model and the Discrete Choice Model from a System Dynamics perspective. The proposed approach allows for the forecasting of the adoption rate and its timing, by understanding the underlying preferences of individual customers and social forces influencing it. A real-scale preliminary application in the German market for electric cars, parameterized through a Conjoint Analysis, is provided. Simulation results indicate that battery charging technology and infrastructures are crucial for the success of electric cars in Germany.**

Keywords—Forecasting Innovation; System Dynamics; Bass Diffusion Model; Discrete Choice Model; Conjoint Analysis; Electric Vehicles (EV); German Electric Car Market.

I. INTRODUCTION

Understanding the adoption process of new products is crucial for most businesses. It is also important for governments when creating policies to regulate the market or to define the necessary infrastructure to support new technologies being introduced, such as medical equipment or electric vehicles.

Although largely investigated since the last century, diffusion processes still remain complex phenomena. Various methodologies, approaches and computer models have been developed to investigate the market diffusion of new products.

In order to contribute to the scientific advancement in this area, this paper proposes a novel simulation-based approach for evaluating how consumers' preferences and social forces influence the introduction of new products. The proposed approach merges concepts from the traditional Bass Diffusion Model with the Discrete Choice Model from a System Dynamics perspective. Compared to other approaches, our model offers the following advantages: a. both timing and market-share can be jointly estimated; b. the model is fully flexible with respect to the number of product attributes, and; c. the model is easily parameterized through Conjoint Analysis without the need of market data. This is illustrated by the real-scale application to the German market for electric cars. The results demonstrate the potential of the proposed approach, to support the understanding of the main drivers for product adoption.

This paper is organized as follows: section II presents a literature review and highlights the research gap; section III overviews the theoretical background employed in the proposed model; section IV introduces the proposed approach; section V presents the preliminary application in the German market for electric cars; section VI proposes future research; and finally, section VII outlines final remarks and conclusions.

II. RELATED WORKS

A myriad of innovation forecasting studies is provided in the literature. The present work concerns Diffusion models, Discrete Choice Models and System Dynamics approaches, as well as the ones applied in the electric car market.

The Bass Diffusion Model [1] is probably the most widely used approach in management science [2]. In its algebraic form, the Bass Model is somewhat restricted to a small set of parameters and strong underlying assumptions. Some works partially relaxed some of these assumptions (e.g. Dodson and Muller [3]) and others extended the model (e.g. Kalish [4], Chatterjee and Eliashberg [5], and Horsky [6]). Interestingly, Bass [2] himself commented on some possible extensions for his seminal work. Two relatively recent state-of-the-art reviews are provided in Frenzel and Grupp [7] and in Meade and Islam [8].

While the Bass Diffusion Model captures innovation timing, the Discrete Choice Model, another popular approach, captures consumers' appraisal of the product's utility [9]. Many interesting works exist in the literature, including Anas [10], that relates information theory with Discrete Choice Models; Drakopoulos [11] discusses the psychological aspects underlying the theory of rational consumers; Kim et al. [9] propose an adjusted Discrete Choice Model that incorporates the choice behavior of the consumer into the dynamics of product diffusion; Lee et al. [12] put forward a methodological framework derived from a static utility function based on the Discrete Choice Model and the Bass Diffusion Model.

ISBN: 978-1-61208-169-4

System Dynamics is also employed in this area. Milling [13] provides an example of the innovation diffusion process from a System Dynamics perspective. The basic structure of his model is identical to the mixed-influence of the Bass model [14], thus the characteristics of the product are not considered explicitly. Mooy [15] used a System Dynamics model with the sociological theory of Memetics, and more recently Park et al. [16] developed a marketing penetration forecasting model for hydrogen vehicles, also using a generalized Bass model in a System Dynamics framework.

Maier [14] explains that variables such as pricing, quality, technical capabilities, etc. can impact on the probability of a purchase, but in his case this probability serves as a multiplier that affects the coefficient of innovation and imitation, or that can delay or speed up the demand. The total product utility is not considered explicitly through a Discrete Choice perspective.

More specifically in the electric car market, many works propose forecasting approaches in the literature, including Discrete Choice (e.g., Beggs [17]), conjoint experiments (e.g., Segal [18], Ewing and Sarigolli [19]), and equation-based models (Urban et al. [20]). An approach quite related to the present work is Klasen and Neumann [21], which combines the Bass Diffusion Theory with the Discrete Choice Model in an agent-based framework to investigate the feasibility of the German's goal for the electric car's adoption rate in next decade. Another contribution from the literature, which is close to the present work, is Meyer and Winebrake [22], but it is dedicated to hydrogen vehicles and the refueling infrastructure. Similarly to the present work, their System Dynamics model encapsulates concepts from the Diffusion Theory and the Discrete Choice Model, but consumers' preference utilities are limited to fuel cost, vehicle price and station density. Moreover, the proposed model does not directly incorporate social forces in a utility model.

Despite their contribution to the concerned literature, and to the best of the authors' knowledge, no work exists which deals directly with consumer preferences and diffusion processes within a System Dynamics perspective for the electric car market. Thus, the model and application domain proposed herein are original.

III. Theorethical BackGround

This section introduces the main concepts employed in the proposed model.

A. Bass Diffusion Model

Traditionally, economic models of innovations' diffusion are founded on biological and sociological research [23]. Perhaps the most well known work in the area is the Bass Diffusion Model [1], which distinguishes between two types of customers: innovators and imitators. This model is described as a set of differential equations employing a small number of parameters. Basically, Bass defined the rate of adoption $S(t)$ as a function of the potential market share $T(t)$, the actual number of adopters $A(t)$, an innovation coefficient p and an imitation coefficient q. Bass formulated it is as following:

$$S(t) = qT(t) + (p-q)A(t) - p[A(t)]^2/T(t). \qquad (1)$$

The Bass model assumes that everything in a diffusion process (e.g., customers' individual characteristics, availability of information about a product, positive and negative personal recommendations, etc.) can be modeled through the parameters q and p. Despite the fact that the Bass model is largely used, its inherent assumptions have been criticized in the literature [8]. Additionally, the Bass Model is not easily parameterized when no market data is available. Thus, radically new products, which imply changes in consumers' behavior, as the electric car, does restrict the use of the Bass Diffusion Model.

Diverse approaches have emerged to improve or extend the Bass model, including the Discrete Choice Model and System Dynamics [21].

B. Discrete Choice Model

The Discrete Choice Model allows for the determination of the relative purchase probability based on products' utilities [24], describing products as a finite set of perfectly substitutable attributes. In short, the probability P_i^k that an individual i will choose a product k from a set of alternatives A_i is given by:

$$P_i^k = 1/\left(1 + \sum_{l\in A_i, l\neq k} e^{(V_i^k - V_i^l)}\right), \qquad (2)$$

where V_i^k is the deterministic component of the utility, described through expressed attitudes toward that alternative. This utility is assumed to be a linear additive function of the product attribute score, such as:

$$V_i^k = \sum_{j\in S^k} a_j^k x_{ij}^k + \sum_{j\in S} b_j x_{ij}^k, \qquad (3)$$

Where x_{ij}^k is the score given by individual i to the k^{th} product alternative of the j^{th} attribute; a_j^k is the utility weight reflecting the importance of the j^{th} attribute defined uniquely for the k^{th} alternative; b_j is the utility weight reflecting the importance of the j^{th} generic attribute defined consistently for all alternatives; S^k is the set of attributes relevant to alternative k only, which is not common to all other alternatives and; S is the set of attributes common to the description of all available alternatives.

It is important to note that both (2) and (3) assume that the individual preferences structure is fixed and depends only on the product attributes, which contradicts one fundamental notion of the Bass diffusion Model, i.e., that preference is also influenced by social forces (e.g. interaction between adopters and non-adopters) through time [21]. Thus, innovation timing cannot be forecasted directly through the use of Discrete Choice Models. This opens interesting opportunities by combining both diffusion and the Discrete Choice Model to incorporate social aspects and consumer preferences. Moreover, the linear structure of equation (3) enables the identification of its coefficients through a least square analysis, using a Conjoint Experiment, even in the case where products are fictitious. Thus, the combination of the Bass with the Discrete Choice Model allows one to forecast not only purchase probability based on product attributes, but also diffusion timing, with a relatively simple form of parameterization, namely, conjoint experiment. System Dynamics provides an interesting framework for doing so.

C. *System Dynamics Applied to Innovation Diffusion*

System Dynamics is an approach for modeling and understanding the behavior of complex systems over time through the study of the system's information-feedback structure. Thereby, interactions among the system structure, amplification in policies and time delays in decision and actions can be analyzed [25]. Basically, the mathematical description of a system dynamic model is realized with the help of differential equations. These equations simulate the resulting behavior of the system over time. The basic elements of the system dynamics model are feedbacks, flows, accumulation of flows (i.e. stocks) and time delays.

The coarse structure of the Bass model is roughly schematized as a System Dynamics model in Fig. 1 (for a detailed explanation of System Dynamics and the Bass model, please refer to Sterman [26]).

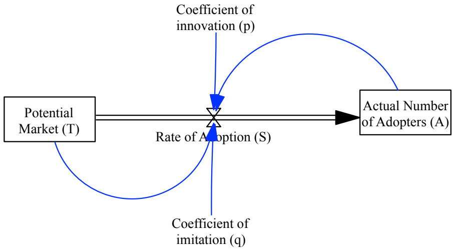

Coefficient of innovation (p)
Potential Market (T)
Rate of Adoption (S)
Actual Number of Adopters (A)
Coefficient of imitation (q)

Figure 1: Bass model from a system dynamics' perspective (inspired by [14]).

In this case, the rate $S(t)$ consumes the stock $T(t)$ and feeds stock $A(t)$, regulated by parameters p and q. In contrast with Bass' original algebraic formulation, the System Dynamics model easily allows diverse policy studies, such as a change in parameters p and q, or even structural changes, such as adding other feedback loops, for example. Consequently, System Dynamics provides an interesting framework to combine the fundamental structure of the Bass Model (to take into consideration innovation timing and social aspects of the diffusion process) with the basic ideas of Discrete Choice Models (incorporating customers' preferences explicitly in accordance with several products' attributes). In the next section, a model describing this possibility is discussed.

IV. PROPOSED METHODOLOGY AND SIMULATION MODEL

The general methodology employed in this work is summarized in Fig. 2 and explained afterwards.

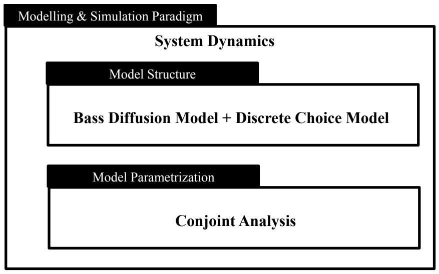

Modelling & Simulation Paradigm
System Dynamics
Model Structure
Bass Diffusion Model + Discrete Choice Model
Model Parametrization
Conjoint Analysis

Figure 2: Proposed methodology.

A. *Modelling and Simulation Paradigm*

The proposed System Dynamic model is depicted in Fig. 3.

This figure shows that the basic structure of the Bass Model (see Section IIIA) is employed, including the typical $A(t)$, $S(t)$ and $T(t)$. In addition, the traditional Bass model is extended in many ways. First, based on Sterman [26], the model captures the replacements/substitutions purchases by the variable discarding rate DR^k of the product alternative k. This is necessary because for the electric car (and many other durables), the adoption timing is slow and can easily overcome the product's life cycle. In this case, based on the car's lifecycle lc, obsolete products have to be replaced, moving consumers back to the potential market when the product is discarded. The rate at which consumers move back was modeled approximately as the adoption rate $S(t)$, delayed by the average lifecycle lc of the product. As the average lifecycle is relatively long for many durables (like cars), the repeated purchase decisions are reasonably similar to the initial purchase decisions; thus after discarding consumers reenter the potential customers' pool [26].

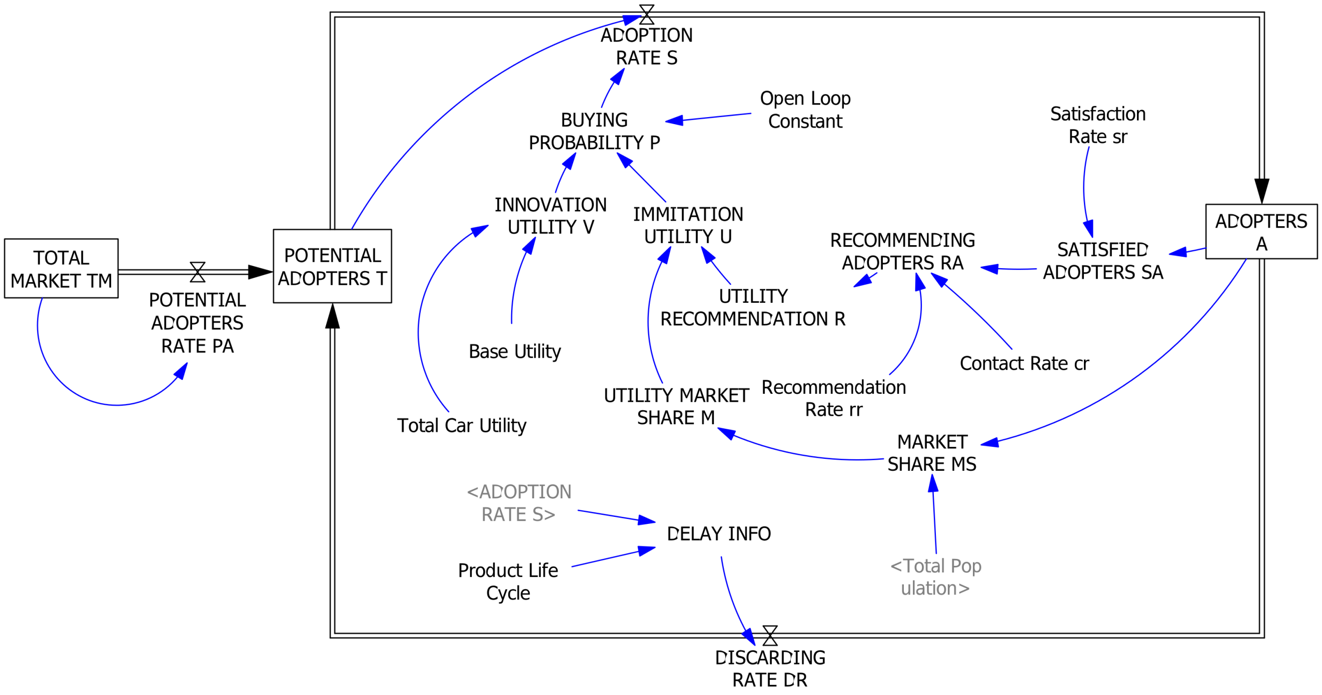


Figure 3: Proposed model.

Another improvement to the traditional model is the inclusion of the total market *TM*(*t*), which represents the untapped market, as suggested by Maier [14]. The stock of potential adopters is increased by *PA*(*t*) rate, i.e. the flow coming from the untapped market, which represents actual consumers of other products that may become new customers at a rate that also depends on the average product lifecycle *lc*. This corrects the traditional diffusion model for the substitution of durables, because not all consumers are immediately available as potential adopters, but only those that need to replace the product after it reaches the end of its lifecycle. Based on this, it is possible to define:

$$A(t) = \int_{t_0}^{t} (S(t) - DR(t))dt, \tag{4}$$

$$S(t) = P \times T(t), \tag{5}$$

where *P* is explained in the next subsection and,

$$DR(t) = S(t - lc), \tag{6}$$

i.e., the discarding rate is delayed by the lifecycle lc in respect to *t*, and:

$$T(t) = \int_{t_0}^{t} (PA(t) + DR(t) - S(t))dt, \tag{7}$$

$$PA(t) = (Total\ Population)/lc, \tag{8}$$

$$TM(t) = Total\ Population - \int_{t_0}^{t} PA(t)dt. \tag{9}$$

The most important contribution of the proposed model is indicated at the center of Fig. 3. Replacing the traditional coefficients *p* and *q*, the buying probability is determined through a model inspired by the Discrete Choice approach combined with the Diffusion Theory, as explained in the next subsection. In this way, the ideas underlying the Bass Model are maintained with the advantage of easy parameterization through Conjoint Analysis, even in the case of radical innovations when no market data is available.

B. *Model Structure for the Buying Process*

The fundamental structural contribution of the proposed model lies on the substitution of buying probabilities by the innovation and imitation coefficients. It was assumed that both innovative and imitative behaviors originate though utility assessment, as proposed by Klasen and Neumann [21]. A similar approach was also recently employed by Goldenberg et al. [27]. In this case, P_i^k is not calculated through (2) as the traditional Discrete Choice Model, since the utility assessment of products V_i^k is replaced by VS_i^k:

$$VS_i^k(t) = V_i^k(t) + U_i^k(t), \tag{10}$$

where $V_i^k(t)$ is defined in (3) and represents the innovation utility, similarly to the innovation coefficient *p* of the Bass model; $U_i^k(t)$ is the imitation utility of a product alternative *k* for individual *i*, representing the coefficient *q* of Bass. By doing so, besides incorporating consumer preferences as preconized by the Discrete Choice Model, equation (10) combines characteristics of the classical diffusion model, including social components. These social components are derived from the perception of clients of the market share and positive recommendations from their entourage, as following:

$$U_i^k(t) = R_i^k(t) + M_i^k(t), \qquad (11)$$

where $R_i^k(t)$ represents the utility of positive recommendations and $M_i^k(t)$ the utility of the market share:

$$R_i^k(t) = f(RA_i^k(t)), \qquad (12)$$

$$M_i^k(t) = g(MS^k(t)), \qquad (13)$$

where $RA_i^k(t)$ represents the quantity of recommending adopters for the k^{th} alternative obtained by individual i; and $MS^k(t)$ is the market share (percentage) of the k^{th} alternative. Both functions *f* and *g* are parameterized with the help of a conjoint experiment, explained in the next subsection. $RA_i^k(t)$ and $MS^k(t)$ are calculated as follows:

$$RA_i^k(t) = rr_i^k \times cr_i^k \times SA^k(t), \qquad (14)$$

$$MS^k(t) = A(t) \times sr^k, \qquad (15)$$

where rr_i^k is the recommendation rate for the k^{th} alternative received by individual i; cr_i^k is the contact rate of individual *i* with people who adopted the k^{th} alternative; $SA^k(t)$ is the quantity of satisfied adopters choosing the k^{th} alternative; sr_i^k is the satisfaction rate of those adopting the k^{th} alternative; and the already defined *A*(*t*) is the total quantity of adopters.

C. Model Parametrization

In order to parameterize the simulation model, the proposed methodology employs a conjoint experiment. The Conjoint Analysis is probably the marketers' favorite methodology for determining how consumers decide among competing products, according to Green et al. [28]. Basically, it measures trade-offs of survey responses concerning preferences and intentions to buy. A conjoint experiment is performed through a field research employing interviews and semi-structured questionnaires with potential consumers. The results are the consumers' individual utility functions for each product attribute (in equation 3).

In the present work, these utility functions constitute the necessary parameters for the utility loop in Fug. 3. As mentioned before, both functions *f* and *g* are produced based on a Conjoint Analysis. For *f*, based on quantities of recommending adopters *A*(*t*), it was possible to define the values of the corresponding utility function of positive recommendations $R_i^k(t)$. Similarly, based on the possible market share $MS^k(t)$, it was possible to determine the corresponding utility function of the market share $M_i^k(t)$. Finally, the innovation utility V_i^k was determined from the sum of the *total car utility* and the *base utility*, both resulting from the Conjoint Analysis. The *total car utility* corresponds to the consumer's average utility of an assumed technology. The *base utility* can be interpreted as a utility deficit of some product alternatives in relation to others. This deficit can be explained by different reasons, e.g. product or technology related uncertainty, lack of information and residual preferences not measured by other products' attributes. For further detail on Conjoint Analysis, the reader is referred to Ewing and Sarigollii [19], Klasen and Neumann [21] and Lee et al. [12].

V. Preliminary Application

The proposed simulation model was applied in a preliminary industrial-scale case in the German market for electric cars.

A. Simulation Problem

As a promising technology to reduce greenhouse gas emissions, the electric car appears, together with other complimentary technologies such as hybrid cars, to be an interesting alternative for consumers. Believing that it is a good alternative, the German government has established an official market goal of 1.000.000 cars sold by 2020, a market share of approximately 2,32%. Recent governmental reports suggest, though, that without government intervention, only 450.000 cars will be sold [9]. Great uncertainty is related to this market, since consumers' reaction to technological limitations, loading infrastructure and green energy generation are still not well understood.

Consequently, understanding the market potential and consumers' preferences is crucial not only for validating the market goal but also for deriving public policies to support new technological and infrastructural developments. As explained in section II, many works propose approaches to forecast the electric car market in the literature, but to the best of the authors' knowledge, no work puts forward an approach dealing directly with consumer preferences and diffusion processes within a system dynamics framework, such as the one proposed herein.

This approach provides an interesting simulation tool to forecast the market share when consumers' preferences favor technical attributes, such as in the electric car market. In addition, it allows for evaluating how social aspects and the product's utility influence the adoption process.

Thus, in this context, the preliminary simulation experiment aims at "*understanding how the main driver of infrastructure, the battery loading time, influences the diffusion of electric cars in Germany*". This simulation objective highlights that the main goal of the simulation study lies in the comparative analysis of different charging technologies. Consequently, absolute forecasting accuracy was not primarily pursued. Also, the scope of the study was limited to the comparison between present internal combustion cars and electric vehicles.

Based on the literature and on the interaction with some automakers, a list of 18 attributes that differentiate an electric car from a conventional one was produced and verified through interviews with two experts, one from the automobile industry and another from a consulting firm. Among the attributes, but not limited to them, price, battery range, variable cost per km, battery charging time, battery durability, CO_2 emissions, maximal velocity, acceleration, loading space, noise level and model exclusivity were included. The simulated technology was based on the newly announced Renault ZE technology, which offers a total range of 180 km and three battery loading possibilities: a normal 7 hours charging at home and/or at the working place, a 30 minutes fast charging and a five minutes battery exchange. As fast charging and battery exchange infrastructures require expensive investments, this simulation enables one to understand how the market can evolve in the case these infrastructures are indeed installed.

The field research consisted of 291 interviews with subjects, conducted between September and October 2009. From these, only those with relevant driving behavior, i.e. mainly city and short distance travellers, were identified as potential consumers. This filtering criterion yielded 183 potential consumers that effectively took part in the conjoint experiment. Thus, based on this proportion, the potential market for electric cars accounts for 63% of the total German car market. With the results of the conjoint experiment, the model was parameterized and simulations were performed, together with some sensitivity analysis.

B. *Simulation and Results*

The proposed model was implemented through Vensim® PLE 5.10d and then configured using information from the Conjoint Analysis, described previously.

Fig. 4 shows the simulation results of the potential market share (in percentage) for the three technologies: a normal 7 hours charging, a fast 30 minutes charging and a five minutes battery replacement.

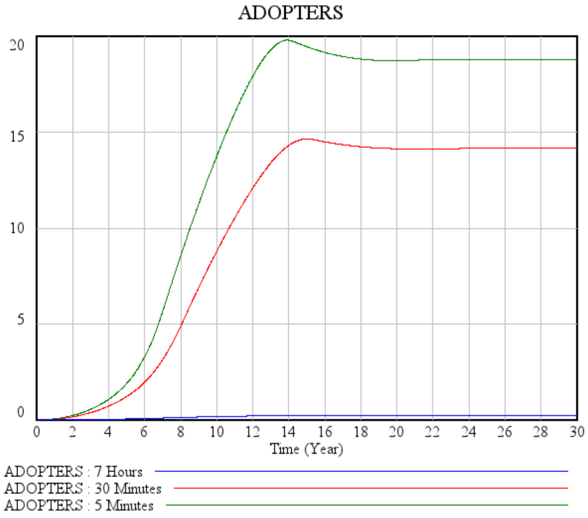


Figure 4: Simulation results for the potential market share.

A number of conclusions can be drawn from Fig. 4. First, if no fast charging infrastructure is available, consumers are not willing to purchase the electric car. The restrictions imposed by limited charging possibilities (only at work, at special public parking lots and at home), together with the long loading time, require a drastic change of consumers' behavior, leading to a less than 0,1% market share. Second, fast high-power charging infrastructure that allows the batteries to be charged in 30 minutes accounts for a market-share of 14% in 15 years. According to this scenario, the German government goal of 1.000.000 cars (3,6% of the market) could be reached in 2020. The impediment lies on the infrastructure, which is not available yet. The latter requires not only high investment, but also time to be built, suggesting that the decision of the first cities where this infrastructure will be built is of a great strategic importance. Third, if battery exchange stations are available, the market-share might increase further, yielding 18% of the potential market (11% of the total market). Although this infrastructure requires a high investment in a stock of available batteries, it could account for a second step in the development of the market.

Even though our simulated experiment offers an idea of the potential market-share, results must be interpreted with care. Modeling assumptions and consumers' high uncertainty about the technology are possible sources of error. Despite the fact that absolute values should be read with caution, comparative analyses, as the one presented here, are valid. Thus, our analysis shows that investment in the right infrastructure can determine the success or failure of electric cars. In our analysis, we restricted ourselves to two main limitations of the electric car, i.e. the battery loading time and necessary infrastructure. Nevertheless, many other scenarios could be created and additional analysis could be done in future works, as discussed in the next section.

VI. Further Research

The preliminary application illustrates the utility of the proposed model. Several additional studies can be performed using the proposed approach and dataset, including some supplementary validation.

An initial structural validation (which is the validity of the set of relations used in the model, as compared with the real processes) was performed based on the literature. This structural validation increases the level of certainty to acceptable levels. Some additional research efforts will be done in the future to deeply verify some assumptions strongly influencing the forecasting behaviour and accuracy. This additional validation effort will also be done through additional literature review in specific areas related to these assumptions.

Another important future work in the electric car market refers to the study of other drivers of the adoption process, such as car price when compared to conventional vehicles, battery durability, maximum speed, and so forth.

Other market sectors can be also investigated in the future, consequently more statistical certainty will be gained as other important application domains will be tested in the future.

Finally, additional works could be performed to develop simpler parameterization approaches in order to increase the model's usability in practice.

VII. Conclusions

This paper proposes a novel simulation-based approach for investigating the innovation adoption process. By combining the Bass Diffusion Theory with the Discrete Choice Model and Conjoint Analysis, from a System Dynamics perspective, it is possible to evaluate how diffusion timing, social aspects and consumer preferences in terms of the products' characteristics influence the introduction of new products in the market.

An illustration of the proposed simulation model for the electric car market in Germany was provided. As a preliminary real-scale application, it was possible to demonstrate how battery charging technology and infrastructure drive customer adoption. This simulation experiment shows the potential of the proposed approach, supporting the understanding of the main drivers of product adoption in strategic planning through an intuitive method.

Several future research works are under way, including additional structural and behavioral validation efforts, as well as assumptions and parameterization investigations.

REFERENCES

[1] F.M. Bass, "A New Product Growth Model for Consumer Durables", Management Science, Vol. 15, No. 5, pp. 215–227, 1969.

[2] F. Bass, "A New Product Growth for Model Consumer Durables: The Bass Model", Management Science, Vol. 50, No. 12, pp. 1833-1840, 2004.

[3] J. A. Dodson and E. Muller, "Models of new product diffusion through advertising and word-of-mouth", Management Science, Vol. 24, No. 15, pp. 1568-1578, 1978.

[4] S. Kalish, "A new product adoption model with price, advertising and uncertainty", Management Science, Vol. 31, No. 12, pp. 1569-1585, 1985.

[5] R. Chatterjee and J. Eliashberg,"The innovation diffusion process in a heterogeneouspopulation: a micromodeling approach", Management Science, Vol. 36, No. 9, pp. 1057-1079, 1990.

[6] D. Horsky, "A diffusion model incorporationg product benefits, price, income and information, Marketing Science, Vol. 9, No. 4, pp. 342-365, 1990.

[7] A. Frenzel and H. Grupp, "Using models of innovation diffusion to forecast market success: a practitioners' guide", Research Evaluation, Vol. 18, No. 1, pp. 39–50, 2009.

[8] N. Meade and T. Islam, "Modelling and forecasting the difusion of innovation: a 25-year review", International Journal of Forecasting, Vol. 22, pp. 519-545, 2006.

[9] W.-J. Kim, J.-D. Lee and T.-Y. Kim, "Demand forecasting for multigenerational products combining discrete choice and dynamics ofdiffusion under technological trajectories", Technological Forecasting & Social Change, Vol. 72, pp. 825–849, 2005.

[10] A. Anas, "Discrete choice theory, information theory and the multinomial logit and gravity models", Transportation Research Part B: Methodological, Vol. 17, No. 1, pp. 13-23, 1983.

[11] S. A. Drakopoulos,"The implicit psychology of the theory of the rational consumer: an interpretation", Australian Economic Papers, Vol. 29, No. 55, pp. 182-198, 1990.

[12] J. Lee, Y. Cho, J.-D. Lee and C.-Y. Lee, "Forecasting future demand for large-screen television sets using conjoint analysis with diffusion model", Technological Forecasting & Social Change, Vol. 73, pp. 362–376, 2006.

[13] P. Milling, "Decision support for marketing new products", In System Dynamics: On the Move, J. Aracil, J. A. D. Machuca and M. Karsky, Eds., Seville, Spain: The System Dynamics Society, pp. 787-793, 1986.

[14] F. H. Maier, "New product diffusion models in innovation management: a system dynamics perspective", System Dynamics Review, Vol. 14, No. 4, pp. 285-308, 1998.

[15] Mooy, R.M., Langley, D.J. and Klok, J., "The ACMI adoption model: predicting the diffusion of innovation", Proc. of the 2004 System Dynamics Conference, July 25 – 29, 2004, Oxford, England.

[16] S. Y. Park, J. W. Kim and D. H. Lee, "Development of a market penetration forecasting model for Hydrogen FuelCell Vehicles considering infrastructure and cost reduction effects", Energy Policy, 2011.

[17] S. Beggs, S. Cardell, and J. Hausman, "Assessing the potential demand for electric cars", Journal ofEconometrics, Vol. 17, No. 1, pp. 1-19, 1981.

[18] R. Segal, "Forecasting the market for electric vehicles in california using conjoint analysis", Energy Journal, Vol. 16, No. 3, pp 89-112, 1995.

[19] G. Ewing and E, Sarigollii,"Assessing consumer preferences for clean-fuel vehicles: A discrete choice experiment", Journal of Public Policy & Marketing, Vol. 19, No. 1, pp. 106-118, 2000.

[20] G. L. Urban and J. R. Hauser,"Design and marketing of new products', Prentice Hall, 1980.

[21] J. Klasen and D. Neumann, "An agent-based method for planning innovations", International Journal of Innovation and Sustainable Development. In Press.

[22] P. E. Meyer and J. Winebrake, "Modeling technology diffusion of complementary goods: The case of hydrogen vehicles and refueling infrastructure", Technovation, Vol. 29, pp. 77–91, 2009.

[23] J.-H. Thun, A. Größler and P. M. Milling, "The Diffusion of Goods Considering Network Externalities: A System Dynamics-Based Approach", Proc. of The 18th International Conference of The System Dynamics Society Sustainability in the Third Millennium, August 6 - 10, 2000, Bergen, Norway.

[24] D. H. Gensch and W. W. Recker, "The multinomial, multiattribute logit choice model", Journal of Marketing Research, Vol. 16, No. 1, pp. 124-132, 1979.

[25] J. W. Forrester, "Industrial Dynamics", Cambridge: MIT Press, 1961.

[26] J. D. Sterman, "Business Dynamics: systems thinking and modeling for a complex world", Boston: McGraw-Hill Higher Education, 2000.

[27] J. Goldenberg, B. Libai and E. Muller, "The chilling effects of network externalities", International Journal of Research in Marketing, Vol. 27, No. 1, pp. 4-15, 2010.

[28] P. E. Green, A. M. Krieger, Y. Wind, "Thirty Years of Conjoint Analysis: Reflections and Prospects", Interfaces, Vol. 31, No. 3, pp. S56-S73, 2001.

[29] Die Zeit. Merkel erhält Elektroauto-Bericht. On-line Newspaper. Available at http://www.zeit.de/news-052011/16/iptc-hfk-20110516-3-30385754xml. Last accessed on May, 20th. 2011.

Plaque Lesion Classification Fuzzy Model Based on Various Color Models

Yuslinda Wati Mohamad Yusof, Hadzli Hashim, Khairul Anam Sulaiman,
Suhaila Subahir, Noor Ezan Abdullah and Fairul Nazmie Osman
Faculty of Electrical Engineering
Universiti Teknologi Mara (UiTM)
Shah Alam, Malaysia
yuslinda@salam.uitm.edu.my, hadzli66@gmail.com, s.khairul@ymail.com, suhailas@salam.uitm.edu.my,
ezan_nea7@yahoo.com , fnazmie81@yahoo.com

Abstract **– This paper investigates discrimination of plaque lesion from other types of psoriasis lesions using fuzzy logic technology. The proposed intelligent model can aid dermatologist in doing pre-diagnosis of psoriasis lesion particularly in hospitals that are scarce of expert persons. Skin lesions can be represented in terms of enhanced image pixel indices from various color models such as *RGB*, *HSV* and *YCbCr*. These indices are used as inputs in designing an intelligent model based on fuzzy algorithm. However prior to that, numerical analysis is done statistically in order to select only significant color components that would infer plaque discrimination from the non-plaque group samples. The outcome of the designed fuzzy model has produced sensitivity and specificity of 72.72% and 90.09% respectively. Eventually, the overall accuracy of the fuzzy model is 81.82%, and is about 7% higher when compared to the optimized ANN model developed earlier from previous work.**

Keyword *–* ***RGB; HSV; YCbCr; Fuzzy logic; MATLAB; SPSS.***

I. INTRODUCTION

Psoriasis comes from the Greek word *psora*, meaning the state of being affected with itch. It is an immune mediated, genetic disease manifesting in the skin and/or the joints. This chronic scaling disease belongs to the papulosquamos diseases group of skin disorders [1]. In psoriasis, the epidermis layer beneath the outer skin surface thickens because of an abnormality growth of melanocyte cells causing dilation of blood vessels for nourishment. The immune system then sends faulty signals that speed up the movement rate of these cells to the surface within days instead of weeks where they will pile up with the dead cells, sometimes creating white, flaky layer over a patch of an inflamed skin. Psoriasis is prevalence worldwide effecting 1% to 2% of the population [2]. In Malaysia, the top five skin disorders seen by dermatologists each year are psoriasis, chronic eczema, allergies, occupational dermatitis and acne [3]. About 3% of the citizen population was reported to suffer from psoriasis alone, compared to only 0.756% that was having malignant skin problems [4]. Psoriasis has variety of clinical presentations. The major ones are the scaly plaques, noduled guttate, reddish erythroderma, and sometimes creating pus in pustular. It may range from just a few spots anywhere on the body to large areas of involvement. Nevertheless, it is not contagious or spreadable from one part of the body to another or from one person to another. However, individuals with severe psoriasis have a profound emotional and social as well as physical impact on their quality of life [5]. The exact cause of psoriasis is unknown, but it is common that psoriasis is heritable [6].

Dermatology is about medical study on skin diseases or lesions [2]. The fundamental concept of learning practiced by dermatologist is by looking at the skin lesion, applying morphological learning method and then follow-up by differential diagnosis steps in order to identify the types of diseases. These methods would include matching the skin lesion appearance to the closest appearance photo from a library text as guidance for the diagnosis. Basically, dermatologist needs to know the preliminary spatial and spectrum information when diagnosing a skin lesion. Since the human eye has limitation in performing such difficult tasks, dermatologists sometimes would rely on true color digital imaging to assist them in their work. Nowadays, with the rapid advances in computer and video technology, producing such a low-cost biomedical imaging based on analysis of color and texture of skin lesions has become very feasible. These support systems not only enhance the dermatologist ability to communicate with patients and colleagues but the quantified data images can be used more efficiently and perhaps diagnose the lesion with better accuracy and efficiency [7].

Many research works have applied available computer vision technology and other sophisticated image processing techniques on capturing and improving images of skin lesions. Characteristics such as **A**symmetry in the shape of lesion, irregular lesion **B**order, variegated **C**oloring in the lesion and **D**iameter of the lesion are useful and important features for dermatologist whenever skin diagnosis is concerned [1]. All of the lesion features based on the **ABCD** rule can be proposed as the front-end inputs in designing an artificial intelligence for automated dermatological diagnoses of a specific lesion. In fact, several work concerning realization of this idea have already been described and implemented where artificial neural network (ANN) were used as the lesion classifier [8-10]. These work focused on discriminating malignant melanoma from other types of skin tumor based on color features where several color models have been experimented to describe the colors characteristics of melanoma in terms of means and variances. Such models are the primary *RGB*, spherical color coordinates (*CIE-*$L\alpha\beta$), relative chromaticity (*rgb*), *CIE-LUV*, and the non-linear transformation model of *YCbCr* and *HSV*. Other artificial intelligence related work that used fuzzy algorithm for skin color image segmentation of lesions was described in these literatures [12-14]. However, these works mainly focused only on enhancement of selected lesion segmentation that utilized *RGB* and *HSV* color models. No discrimination between lesions was reported.

Application of neural network in automated diagnosis of psoriasis skin lesion has been initiated by Hashim *et. al.* His investigation was on discriminating plaque lesion from other psoriasis using ANN algorithm where several color models were used and reported in these literatures [14-16]. Most of the trained optimized ANN models were trained using Levenberg-Marquardt algorithm and later, validated based on the receiver operating characteristic (ROC) curve that conveniently displays diagnostic accuracy expressed in terms of *sensitivity* and *specificity* with respect to selected threshold values. When comparing between color models, the investigations have shown that *HSV* is the best color space for discriminating plaque with accuracy of 75.21% [15]. This result will be used for comparison later on with the investigation outcome of this paper.

As an additional study of the previous work, investigation on using fuzzy algorithm is proposed as the pattern classifier based on various color models. Therefore, this paper described the scope of work in Section I, followed by methodology in Section II. Section III is for the results and discussion. Finally, Section IV is the conclusion of the findings.

II. Scope of work

Scope of this study is on designing an intelligent fuzzy model that can discriminate plaque from non-plaque group of lesions based on three color models *i.e. RGB*, *YCbCr* and *HSV*. Digital image processing technique is applied to produce quantitative color measurements. Each lesion sample from the respective group is represented by its differential mean pixel index which will be elaborated in the following section. The group indices are tested statistically using Statistical Package for Social Sciences (SPSS) tool in order to observe any discrimination through independent *t*-test and eventually, computation of measurement range with respect to the group's mean, upper and lower confidence limit (*UCL* and *LCL*) [17]. The information is then used as inputs to design intelligent model using a fuzzy algorithm.

III. Methodology

A. Data Collection

Three sets of digital images of psoriasis lesions representing plaque, guttate, and erythroderma were captured from psoriasis patients under controlled environment at the Hospital Universiti Kebangsaan Malaysia (HUKM) and Hospital Melaka. These images have 600 samples representing the 3 lesions (plaque:guttate:erythroderma), cropped and divided into training set of 468 samples with a proportion of 234:117:117 representing each lesion respectively. The testing set has 132 samples with 66:33:33 proportions. Just notice that all these sets are evenly distributed between plaque and non-plaque samples. These samples were then separated by three categorized color models which are the *RGB*, *YCbCr* and *HSV*. The samples were later clustered into two groups: plague and non-plague (comprised of guttate and erythroderma) and acts as inputs for fuzzy logic system. Initially, the *RGB* component color images were acquired using FinePix 6900 Zoom (FujiFilm) digital camera, with pixel resolution of 786x512. This size is sufficient for analysis, as all relevant details of the lesions are shown [18]. During the photo session, the camera was placed at a distance of one foot directly above the patient's skin.

B. Pre-processing

Each image from the stored database was being filtered using median filter technique in order to remove any artifacts such as small white ellipse lines or dots. These artifacts can be considered as impulsive noise and may thus be reduced using a median filter given by [19]:

$$P_{med}\left(m,,n\right) = median\Big\{P\left(m-k, n-1\right) \Big| \ -\frac{N_{med}-1}{2} \le k, \quad l \le \frac{N_{med}-1}{2} \wedge 1 \le m-k \le m \wedge 1 \le n-1 \le N \Big\} \qquad (1)$$

where N_{med} is an odd number that indicates the selected size of a two dimensional median filter. *P* represents all the color components. Note that only square median filter kernel was considered. After the filtering process, region of interest (ROI) which includes a sample of normal skin and three samples of lesion area were selected. Each image was carefully studied and observed by the dermatologists involved in this research. Once the regions have been identified by them, the images were cropped out sequentially with the normal skin first and followed by the other lesion sample. All samples were then been resized to a dimension of 256 by 256 pixel area for consistency in this research [14-16]. The differential method of gathering lesion sample color indices is defined as:

$$P_{hsv}(i,j) = [P_{hsv}\ lesion\ (i,j) - X_{hsv}\ skin] \qquad (2)$$

where X_{hsv} are the computed mean index for each color component of the selected ROI normal skin sample.

HSV parameters were derived non-linearly from *RGB* through the following mathematical conversion [15];

$$H = \begin{cases} \text{undefined} & \text{if } Max = Min \\ 60\,x\,\dfrac{G-B}{Max-Min}+0 & \text{if } Max = R \\ & \text{and } G \ge B \\ 60\,x\,\dfrac{G-B}{Max-Min}+360 & \text{if } Max = R \\ & \text{and } G < B \\ 60\,x\,\dfrac{B-R}{Max-Min}+120 & \text{if } Max = G \\ 60\,x\,\dfrac{R-G}{Max-Min}+240 & \text{if } Max = B \end{cases} \qquad (3)$$

$$S = \begin{cases} 0 & \text{if } Max = 0 \\ 1 - \dfrac{Min}{Max}, & otherwise \end{cases} \tag{4}$$

$$V = Max[R, G, B] \tag{5}$$

YCbCr parameters were then created from the corresponding gamma adjusted RGB (red, green, blue) source using two defined Kb and Kr and the transformation are as follows:

$$Y \;\; = 0.299{*}R' + 0.587{*}G' + 0.114{*}B' \tag{6}$$

$$Cb = -0.168736{*}R' - 0.331264{*}G' + 0.5{*}B' \tag{7}$$

$$Cr = 0.5{*}R' - 0.418688{*}G' - 0.081312{*}B' \tag{8}$$

Here, *R'*, *G'* and *B'* are assumed to be nonlinear (gamma-adjusted) and has a nominal range from 0 to 1, with 0 representing the minimum intensity and 1 the maximum. The prime symbols denote the use of gamma adjustment. The resulting luma (*Y*) value will then have a nominal range from 0 to 1, and the chroma color difference(*Cb* and *Cr*) values will have a nominal range from -0.5 to +0.5 [16].

C. *Inference Test*

Independent *t*-test was applied to obtain *p*-values that provide valuable information because it measures the amount of statistical evidence that supports the alternative hypothesis in this work as shown below.

Null hypothesis (no discrimination), H_0 :

$$\mu_y - \mu_z = 0 \tag{9}$$

Alternative hypothesis (has discrimination), H_1 :

$$\mu_y - \mu_z \neq 0 \tag{10}$$

Where μ_y ,μ_z is the population mean.

Statisticians translated *p*-value into 4 different descriptive terms, which are [17]:

- *p*-value <0.01:
  There is overwhelming evidence to infer that H_1 is true
- 0.01<*p*-value < 0.05:
  There is strong evidence to infer that H_1 is true
- 0.05<*p*-value < 0.1:
  There is weak evidence to infer that H_1 is true
- 0.1<*p*-value < 1.0:
  There is no evidence to infer that H_1 is true

D. *Fuzzy Inference Systems*

Basically, fuzzy logic based in fuzzy inference systems (FIS) or Fuzzy-rule based systems as depicted in Figure 1 [20]. In this system a fuzzy fication interface transforms the input in degrees of match with linguistic values. Then a decision-making unit performs the inference operations on the rule base. Finally, a defuzzification which is a process of combining applicable fuzzy rules in order to assign a value to a given output interface transforms the fuzzy result of the interference process in a crisp output. Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. The mapping then provides a basis from which decisions can be made, or patterns discerned. The process of fuzzy inference involves all of the pieces that are described in the previous sections: membership functions, fuzzy logic operators, and *if then* rules. There are two types of fuzzy inference systems that can be implemented in the Fuzzy Logic Toolbox: Mamdani type and Sugeno-type. These two types of inference systems vary somewhat in the way outputs are determined. But, in this study, Mamdani type is selected due it more suitable for human input [20].

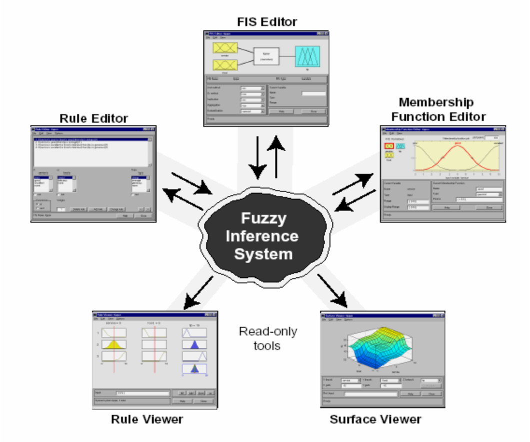
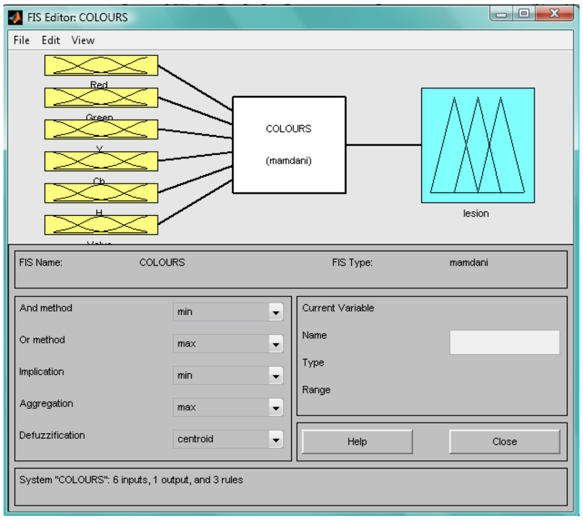


Figure1. Fuzzy inference systems (FIS)

E. *The FIS editor*

The fuzzy inference system (FIS) Editor displays general information about a fuzzy inference system as shown in Figure 2. There is a simple diagram at the top that shows the names of each input variable on the left, and those of each output variable on the right. The sample membership functions shown in the boxes are just icons and do not depict the actual shapes of the membership functions [20].



FIS Name: COLOURS

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

Name

Type

Range

System "COLOURS": 6 inputs, 1 output, and 3 rules

Figure 2. FIS Editor for colors

F. *Input and Output Membership Function*

The fuzzy sets and the membership functions are defined in the following manner: if *X* is a collection of objects, then a fuzzy sets *A* in *X* is defined as a set of order pairs [20], as in

$$A = \{x, \mu_A(x) \mid x \in X\} \quad (9)$$

From (7), *A* represent a fuzzy set and $\mu_A(x)$ is a memberships function of x in *A*.

G. *Fuzzy Rule Base*

Fuzzy rule are a collection of linguistic statements that describe how the FIS should make a decision regarding classifying an input or controlling an output. This fuzzy logic was used production rules that consist of s precondition (IF-part) and a consequence (THEN-part) to represent the relation among the linguistic variables and to derive actions from inputs. There are six rules were defined for fuzzy logic system.

H. *Defuzzification*

The choice of defuzzification method depends on the precision of the result [20]. There are five types of defuzzification methods where for this study, the Min of maximum (MoM) technique was chosen. This technique takes the output distribution and finds its mean of maxima to come up with one crisp number.

I. *Performance Indicators*

Optimizations of the designed models for best learning coefficients were based on performance indicators such as sensitivity, specificity, diagnostic accuracy and receiver operating characteristic curve. Sensitivity and specificity are commonly used terms that generally describe the accuracy of a test. Sensitivity is a measure of the ratio or percentage of ‘true’ lesions (*TP*) and a positive diagnostic test result $(D+)$. It represents the actual percentage of a ‘true’ lesion disease realized by a positive test result and is also known as true positive rate (*TPR*), defined as [14-16]:

$$Sensitivity: TPR = \frac{TP}{D+} \quad (13)$$

Specificity measures the ratio or percentage of ‘false’ lesions (*TN*) and with a negative diagnostic test result $(D-)$. It is actually represents the actual percentage of a ‘false’ lesion condition realized by a negative diagnostic test. Specificity is also termed as true negative rate (*TNR*) and is given as:

$$Specificity: TNR = \frac{TN}{D-} \quad (14)$$

The percentage for diagnostic accuracy (*DA*) refers to the percentage of samples that have been correctly classified or diagnosed, and have output values within the predefined threshold range for the respective output level. It can be derived as:

$$Accuracy: DA = \frac{TPR + TNR}{N} x100\% \quad (15)$$

IV. Result and Discussion

Table I shows multiple comparison independent *t*-tests result to obtain *p*-values for quantitative measurement. It is observed that three color component; *B*, *S* and *Cb* has *p*-value >0.05 which implies that plaque cannot be discriminated from either both or one of the non-plaque lesions. Thus only six color components were considered for constructing the fuzzy model.

TABLE I PLAQUE INDEPENDENT *t*-TEST MULTIPLE COMPARISON

Component	Non-plaque	Significant *p*-value
R	guttate	0.00
	erythroderma	0.00
G	guttate	0.00
	erythroderma	0.00
B	guttate	0.00
	erythroderma	0.06
Y	guttate	0.00
	erythroderma	0.00
Cb	guttate	0.00
	erythroderma	0.00
Cr	guttate	0.501
	erythroderma	0.00
H	guttate	0.00
	erythroderma	0.00
S	guttate	0.22
	erythroderma	0.90
V	guttate	0.00
	erythroderma	0.00

Three inputs from each of these components for the model would then represent the *Low*, *Medium* and *High*

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-169-4

level while the output has two levels for plaque and non-plaque. Ranges for each input membership function are taken from the outcome measurements of error plot obtained by the descriptive explorer SPSS with respect to *UCL* and *LCL*. The ranges for these inputs are shown in Table II. The number of training set samples is 468 and each sample's values were set as plaque boundary in membership function editor for FIS. Eventually, the triangular (trimf) has been used because it is the simplest way to describe the range of output. However, for input membership function, gaussian (gaussmf) has been used where it has more suitable curves that have advantage of being smooth and nonzero at all points.

TABLE II INPUT MEMBERSHIP FUNCTION FOR LOW, MEDIUM AND HIGH LEVEL

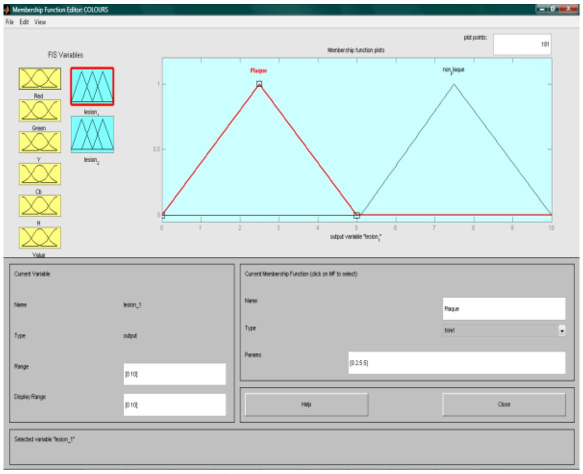
Component	Low Level		Medium level		High level	
	Range		Range		Range	
	LCL	*UCL*	*LCL*	*UCL*	*LCL*	*UCL*
R	-36.931	-29.171	-25.806	-16.355	14.220	32.6249
G	-53.276	-46.785	-27.776	-20.371	5.942	18.173
Y	-40.102	-34.548	-21.47	-14.927	7.657	19.546
Cb	2.8027	4.1084	6.6155	8.222	-0.8584	0.8737
H	0.8033	0.8611	0.6819	0.7384	0.5436	0.6799
V	-31.731	-24.955	-8.7594	-0.8617	20.0866	36.7455

Table III below shows the output ranges where it is divided into two categories; plaque and non plaque. Range of 0-5 represents plaque while range of 5.1-10, is for non-plaque. For example, if a new lesion sample is being applied to this designed fuzzy model, and has output range between 0-5, therefore that this sample is classified as plaque.

TABLE III OUTPUT RANGE OF MEMBERSHIP FUNCTION

Output	Range
Plaque	0.00 – 5.00
Non-plaque	5.10 – 10.00

Figure 3 describes the membership function for this investigation. The six yellow boxes shown on the left-hand side of the figure are the inputs membership function that have being applied to build the fuzzy system while the blue color box presents both targeted outputs. The membership function's value was obtained from previous input tables. The collection of linguistic statements was combined and a decision was made regarding classifying an input or controlling an output. In this system, fuzzyfication has transformed each input in terms of degree of matching with the respective linguistic values and later, an inference operation was performed by a decision making unit.



plot points: 181

Current Variable
Name lesion_1
Type output
Range [0 10]
Display Range [0 10]

Current Membership Function (click on MF to select)
Name Plaque
Type trimf
Params [0 2.5 5]

Selected variable "lesion_1"

Figure3. Membership function creates based on range obtained

Table IV shows the performance indicators output produced by the fuzzy model. The testing set comprised of 132 samples of lesion tested in the fuzzy rule viewer. The percentage accuracy for plaque or TPR (sensitivity) is 72.72%, implying the model can recognize 48 samples out of 100 positively diagnosed samples. Only 18 samples would be falsely indentified as non-plaque. Alternatively when the model was tested with 100 negatively diagnosed as non-plaque, the percentage accuracy TNR (specificity) is 90.09%. This implies that the model has better capability to recognize non-plaque samples. Only 6 samples were falsely identified as positive. The overall diagnostic accuracy (DA) is calculated using equation (15) and for this fuzzy model, it is 81.82%.

TABLE **IV** OUTPUT AFTER TESTING USING FUZZY LOGIC

Group	Tested Samples	Plaque	Non-plaque
Plaque	66	TP=48 (TPR=72.72%)	18
Non-plaque	66	6	TN=60 (TNR=90.09%)

Finally, the performance of the designed fuzzy model is compared to ANN best model performance as described in Section I. Table V shows the comparison in terms of DA% and as a note, the number of training and testing samples used for designing and validating both models are the same. From the table, it is noted that the fuzzy model has higher percentage accuracy and outperformed ANN model by atleast 7%. This simulation results indicates that the fuzzy model is better in discriminating plaque lesion or otherwise non-plaque lesion.

TABLE V
PERFORMANCE COMPARISON BETWEEN FUZZY MODEL AND ANN BEST MODEL FOR INTELLIGENT PLAQUE LESION CLASSIFICATION

	Fuzzy Model	ANN Model
Diagnostic Accuracy (%)	81.82	75.00

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-169-4

V. CONCLUSION

The aim of this investigation is to design an intelligent fuzzy model that can discriminate plaque skin lesions from non-plaque or otherwise. This dermatological pre-diagnostic procedure utilized on lesion sample's color image processing and fuzzy logic algorithm. Combination of various color models, *i.e. RGB*, *YCbCr* and *HSV* were chosen because of widely being used for skin image study and also to be consistent with the previous work that used ANN as the pattern classifier. The front-end phase involved enhanced samples image processing for feature extraction in terms of differential mean pixel indices. These indices were statistically using independent *t*-test in order to identify the valid color components that have strong evidence for discriminating plaque from non-plaque group. The identified color components' parameters in terms of ranges of statistical measurements with respect to *UCL* and *LCL* were later used as the input membership function when designing the fuzzy model. While training the model with 432 samples, fuzzy fication has transformed each input in terms of degree of matching with the respective linguistic values and later, an inference operation was performed by a decision making unit where the targeted model output is to decide between plaque and non-plaque. Performance validation was conducted later on the fuzzy model with 132 samples of lesion and the result showed that it produces an overall accuracy of 81.82%. This proposed fuzzy model has also outperformed the best designed ANN model from previous work by at least 7% in terms of overall accuracy. The outcome of this experiment has concluded that fuzzy algorithm has better accuracy and can be recommended to be utilized in developing intelligent plaque psoriasis classification model.

ACKNOWLEDGMENT

The authors would like to acknowledge Dr. Saadiah Sulaiman from Hospital Universiti Kebangsaan Malaysia (HUKM) and Dr. Roshidah Baba from Hospital Melaka (HM) for providing the psoriasis raw data images prior to the preparation of this project.

REFERENCES

[1] K. Wilheim, P. Elsner, E. Berardesca, and H. Maibach, "Atopic Dermatitis and Other Skin Diseases," in *Bioengineering of the Skin*: *Skin Imaging & Analysis, 2nd Edition*, CRC Press, 2006, pp. 289-296.

[2] H. M. Sander, L. F. Norris, P. E. Phillips, and A. Menter, "The annual cost of psoriasis," *Journal American Academy Dermatology*, vol. 28, pp. 422-425, 1993.

[3] S. C. Lim, R. Baba, and H. Hashim, "Malaysian psoriasis registry - a start," Cyberjaya, Malaysia: Paper presented at the 29th Annual Congress of Dermatology Society of Malaysia, 2004.

[4] G. C. C. Lim and Y. Halimah (Eds), "Second report of the national cancer registry, cancer incidence in Malaysia 2003," National Cancer Registry, Ministry of Malaysia, Kuala Lumpur, ISSN pp. 1675-8870, 2004.

[5] S. W. Weiss, A. B. Kiimball, D. J. Liewehr, A. Blauvelt, M. L. Turner, and E. J. EmanueL, "Quantifying the harmful effect of psoriasis on healh-related quality of life," *Journal American Academy Dermatology*, vol. 47, no. 4, pp. 512-518, 2005.

[6] J. T. Elder, A. T. Bruce, J. E. Gudjonsson. *et. al*, "Molecular Dissection of Psoriasis: Integrating Genetics and Biology , " *Journal of Investigative Dermatology*, vol. 130, pp. 1213-1226, 2009.

[7] N. E. Abdullah, H. Hashim, A. S. Kusim and E. A. Akmar, "Diagnostic Model of Guttate Lesion Utilizing Gaussian RGB Indices Through ANN," in Proc. of the 5th Asia Modelling Symposium 2011 (AMS 2011), Kuala Lumpur, Malaysia, May 23 2011.

[8] R. Dua, D. G. Beetner, W. V. Stoecker, and D. C. Wunsch, "Detection of basal cell carcinoma using electrical impedance and neural networks," *IEEE Trans. on Biomedical Engineering*, vol. 51, no. 1, pp. 66-70, 2004.

[9] E. Zagrouba and W. Barhoumi, "A preliminary approach for the automated recognition of malignant melanoma," *Image Anal. Stereol.*, vol. 23, pp. 121-135, 2004.

[10] A. Sboner, P. Bauer, G. Zumiani, C. Eccher, E. Blanzieri, S. Forti, and M. Cristofolin, "Clinical validation of an automated system for supporting early diagnosis of melanoma," *Skin Research and Technology*, vol. 10, pp. 184-192, 2004.

[11] T. Y. Ng, T. L. Benny, and Y. M. Fung, "Determining the Asymmetries of Skin Lesions with Fuzzy Borders," in Proc. of the 3rd IEEE Symposium on BioInformatics and BioEngineering (BIBE'03), ISBN 0-7695-1907-5, 2003.

[12] S. Keke, Z. Peng, and L. Guohui, "Study on Skin Color Image Segmentation Used by Fuzzy-c-means arithmetic, " in Proc. of the 7th IEEE Int. Conf. on Fuzzy Systems and Knowledge Discovery (FSKD 2010), ISBN 978-1-4244-5934-6, 2010.

[13] K. Shang, L. Ying, N. Hai-jing, and L. Yu-fu, "Method of reducing dimensions of segmentation feature parameter applied to skin erythema image segmentation," in Proc. of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, China, September 1-4, 2005.

[14] N. E. Abdullah, H. Hashim, F. N. Osman, "Comparison between Various Supervised ANN Algorithm uisng RGB Indices for Plague Classification," in Proc. of the Int. Conf. on Electronic Devices, System & Applications (ICDESA 2010), Kuala Lumpur, Malaysia, April 12-12 2010.

[15] M. M. Kamal, H. Hashim, F. N. Osman and R. H. A. Rashid, "Intelligent Classification of Plaque Lesion with Emulation of Human Vision Perception," in Proc. of the 8th WSEAS Int. Conf. on Circuits, Systems, Electronics, Control & Signal Processing (CSECS'09), Tenerife, Canary Islands, Spain, December 14-16, 2009.

[16] H. Hashim, F. N. Osman, and N. Khairudin, "Automated Plaque Diagnosis Utilizing Levenberg Marquardt & Radial Basis Function With Supervised Training Of Chromatic Colors," in Proc. of the 10th WSEAS International Conference on Neural Networks (NN'09), Prague, Czech Republic, 2009.

[17] G. Keller and B. Warrack, "Inference About the Comparisons of Two Populations," in *Statistics for Management and Economics*, 5th. ed California: Duxbury Thompson Learning, 2000, pp. 394-470.

[18] A. Bittorf, M. Fartasch, G. Schuler, and T. L. Diepgen, "Resolution requirements for digital images in dermatology," *Journal American Academy Dermatology*, vol. 37, pp. 195-198, 1997.

[19] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision 3rd Edition*. London: Chapman Hall, 2010.

[20] S. J.Chapman, *MATLAB Programming for Engineers* 3rd Edition, Thomson, 2004.

A Multi-Layer Constraint and Decision Support System for Construction Operation

A simulation framework for the exterior construction works in high rise building

Amir Elmahdi, Hong-Ha Le, Hans-Joachim Bargstädt
Institute for Construction Engineering and Management
Bauhaus Universität Weimar, Germany
emails: amir.elmahdi@uni-weimar.de, hong-ha.le@uni-weimar.de, hans-joachim.bargstaedt@uni-weimar.de

***Abstract*—Up till now, traditional methods of planning construction projects such as 2D technologies including bar charts, network diagrams and other scheduling tools, which are typical methods for modeling the progress of a project, have been the only methods considered. These planning tools are incapable of acknowledging different aspects of construction projects, that it to say, they ignore the space requirements, in which activities are executed and the impact of weather on construction processes. This leads to drawbacks during the execution of these tasks such as work area overlapping, reduction in productivity, safety hazards, long hauling paths and poor quality of work. Therefore, the objective of this paper is to acknowledge the requirements of spatial aspects and the impact of weather, and to integrate these requirements into one simulation framework model. We propose a constraints-based simulation framework. That is to say, the simulation framework executes only tasks, which satisfy the requirement of the space availability and sensitivity towards weather conditions. The implementation of the multi-layer decision support system for construction operations was carried out in Plant Simulation environment and the Simulation Toolkit Shipbuilding (STS) Tool Box Component. This concept will assist site managers in planning and updating the near-term activities for the exterior construction work of high rise buildings. This is on-going research; therefore, other aspects such as execution strategies for different tasks will be developed in a future approach. This research contributes a new approach to integrating different construction problems in one simulation model, which have not been much considered in previous work.**

Keywords-Weather impact; spatial constraints; construction operation; plant simulation.

I. INTRODUCTION

Construction projects are subject to quite a number of influencing factors and are characterized by many unavoidable disturbances. Typical influencing factors in the construction processes are workspace overlapping, missing material and poor quality of work or delays caused by workspace congestion and/or bad weather conditions. They are difficult to predict beforehand and their impact is difficult to evaluate by using the current traditional planning tools due to their complexity [1]. Usually, project managers deal with these disturbances and uncertainties mainly as they happen, using their experiences, historical data and “gut engineering feeling”. However, the stochastic nature of construction processes, the principles superintended in the intervention of each resource as well as the respective interaction between the resources and the associated workspace requirements for the different trades cannot be managed efficiently using traditional planning tools, such as 2D technologies including bar charts, network diagrams and other scheduling tools [2]. These methods are not capable of considering the different influencing parameters of construction operations such as the attributes of component installation and their spatial information as well as the impact of weather [3]. This leads to drawbacks during the execution of these tasks such as overlapping of work areas, reduction in productivity, safety hazards, and poor quality of work. Traditional measures such as working on weekends or increasing the number of workers have been applied to overcome these disturbances. However, they are often used without detailed analysis and so they do not give the desired effect or are oversized. Therefore, project managers are in need of new and more innovative *decision support tools* that extend the use of traditional planning tools from the planning phase to assist them at the construction phase in order to deliver the end product of the project on time and on budget.

Simulation models have been used to examine the impact of possible disturbances of construction processes and to compare all possible strategies and methodologies for task execution so that project managers identify the actual encountered problems and analyze their impact [4]. Thus they can undertake appropriate measures and identify the most appropriate and cost effective solutions. However, simulation models lack the development of an integrated system, which incorporates different construction aspects such as technology, space, logistics, or weather conditions, etc. in one simulation model.

The paper is structured as follows: first, we summarize previous work on workspace modeling and the impact of weather on construction activities. Then, we introduce the research work at the Institute for Construction Engineering and Management at Bauhaus University. Next, we describe our methodology to address the proposed objective of this paper. Lastly, we summarize our conclusion and future work.

II. MOTIVATION

The exterior construction work phase is shaped by the involvement of many individual trades, or rather companies. That is to say, there are many different activity fields. Thus, there are a variety of construction processes, material properties and storage methods, equipment, and different workspace needs. These trades are restricted to perform their job in limited workspaces. Moreover, the type of work focuses mainly on manual assembly, adjustment, and some awkward positioning in operations such as on scaffolding, ceiling and wall surfaces [5]. Furthermore, there is a high degree of interdependencies and technological dependencies of the individual trades.

In another scenario, weather conditions are also a factor, which affect the exterior construction work phase. Benjamin and Greenwald suggest that 50% of construction activities are sensitive to weather conditions [6]. The final product of construction operations is a collection of different interactions between materials, multiple pieces of equipment and crew members, which are completely dependent on the weather status. Therefore the impact of weather is an important factor, which should be seriously considered in estimating construction time. Project planners normally prepare for additional time in the construction schedule to consider delays due to bad weather conditions. However, the buffer time is used without exact analysis and is based on the experience of project managers.

This paper proposes a multi-layer decision support system. We have developed our system in a constrained-based simulation framework environment, which includes an integration of technological dependencies, the dynamic nature of workspace management and the impact of weather on construction operations at the project level. Then, the fulfillment of the constraints at the construction site can be checked and verified. The developed system has been implemented for trades in the exterior construction work of high rise buildings.

III. Literature Review

In the following subsections, we highlight the previous work related to the research areas of workspace problems, weather impact and some main researches at Bauhaus Universität Weimar.

A. Related work on workspace modeling

Workspace is conventionally addressed solely by different researchers. For instance, Riley and Sanvido develop a methodology to constrict workspace by the actual work in place and the amount of space available [7][8]. They extended their previous proposal by defining various space patterns for different trades based on selected methods of working. Akinci et al. developed a methodology to model construction activities in 4D CAD models, by formalizing the general description of space requirements through a computer system. By using 4D CAD a user can automatically generate the project-specific workspaces according to schedule information and represent the workspaces in four dimensions with their relationships to construction methods [9]. Akbas proposes a geometry-based process model (GPM) that uses geometric models to create and simulate workflows and work locations. This method provides spatial insight into the planning of workspaces and space buffers for repetitive crew activities [10].

B. Related work on weather impact

Similarly, the impact of weather on construction activities has also been set to different studies to determine their severity on construction operations. Some studies estimate the relationship between weather parameters and productivity or task duration using regression analysis or neural networks [11][12]. Related to the impact of bad weather conditions on productivity and the duration of construction activities, some other researchers have pointed out how these impacts affect baseline schedules and have also analyzed weather-related construction claims [13][14][15]. These researches provide a decision support framework to analyze the impact of weather on the whole schedule, where the input data is construction processes, weather historical data, and the impacts of weather, whereas the output is the final schedule with a weather-related delay duration. Some construction types have been researched concerning how weather impacts different construction types such as masonry, highway construction, general construction, transportation construction [16], wind turbine construction [17], and earthwork [18].

C. Research at the Bauhaus Universität Weimar

At Bauhaus-Universität Weimar, the Institute for Construction Engineering and Management researches on different aspects of modeling construction and manufacturing processes. For instance, Beißert et al. propose a Constraint-based Simulation for modeling construction processes [19][20]. Thereby, the construction tasks and their constraints for production such as technological dependencies, availability and capacity can be specified, and valid execution schedules can be generated. Further work developed by Voigtmann et al. concern construction site logistics between construction sites and work locations [21][22].

Le and Bargstädt have developed a simulation model to acknowledge the impact of weather on construction processes [23][24]. They developed a network component "WEATHER" within the software Plant Simulation. This network generates weather data and makes decisions on how weather may impact construction processes. The impact of weather has been divided into 3 cases: (1) temporarily prevents workers from working, (2) affects the delivery of material by preventing cranes from operating, (3) reduces labour productivity causing the extension of activities' construction duration [23]. Thereby weather thresholds such as wind velocity, temperature, humidity and precipitation for the first and second cases need to be decided. Besides, relationships between productivity and weather parameters need to be estimated for the third case. The weather impact decisions made by the "WEATHER" component are finally integrated into the construction processes as weather constraints. Thus, the weather-related schedule is provided with the estimated delays.

Similarly, Elmahdi and Bargstädt acknowledge the workspace requirements within the schedule plan for good workmanship. Based on literature and site observations, they classified the different required areas in large scale building projects [1]. With this they propose a semi-automatic methodology to generate the required areas for the scheduled project activities [3]. The acknowledgement of workspace requirements are developed in a new network component “SpatialNetwork”. Furthermore, the “SpatialNetwork” is embedded as an additional constraint component within the software Plant Simulation. Thus the fulfillment of spatial constraints at the construction site can be checked and verified.

However, all these advanced works have been considered in separate models. That is to say, there is no interaction between the “WEATHER”, “SpatialNetwork” and the site logistic components. Therefore, the objective of this paper is to combine three aspects within one simulation framework. To achieve this goal we have to acknowledge technological dependencies with individual trades, workspace requirements and the impact of weather. Furthermore, we integrate these requirements as additional constraints. Then, the fulfillment of these constraints at the construction site can be checked in a hierarchical structure and verified.

IV. Multi-layer constraint and decision Simulation framework

We propose a constraint-based simulation to achieve the objective of this paper. Therefore, in the following subsections, the fundamental idea of the constraint-based simulation concept is introduced and the framework of the multilayer simulation framework is described.

A. Constraint-based simulation

The proposed multi-layer simulation framework to acknowledge the technological and spatial constraints as well as the impact of weather on construction processes is a constraint based simulation environment. Spriprasert and Dawood define constraint in the context of construction as “one that restricts, limits or regulates commencement or progress of work-face operations to achieve construction products within agreed time, cost and quality [25]”. They classify the different types of constraints into three major groups: physical, contract and enabler constraints [25]. Koenig et al formalize two characteristics for the constraints: hard constraints and soft constraints [26]. Hard constraints define conditions that are embedded with work steps, which must be fulfilled before work steps can be started. Soft constraints describe also conditions that are embedded with work steps. However, these conditions are not necessary to be completely fulfilled. Our framework acknowledges this approach for spatial aspects and the impact of weather.

B. Description of multi-layer constraint and decision simulation framework

Fig. 1 illustrates the concept of the multi-layer simulation framework to investigate the impact of weather conditions and spatial conflicts on construction performance. The framework consists of five layers which represent different types of constraints and decisions.

In this framework, the weather and spatial impacts are represented as hard or soft constraints of the construction processes. For example, safety issues for crane operation such as wind conditions or the required workspaces such as material or equipment are describe as hard constraints. On the other hand, the size of workspace and the labor productivity are described as soft constraints. The three cases of weather impact mentioned in the previous section are considered in this framework. Besides these, the required spaces and space conflicts are also examined.

The first layer describes the technological constraints. The technological constraints include global and local constraints. While global constraints describe the priority of the scheduled works between different trades such as the installation of windows before facade and/or between different objects for one trade (such as window-East before window-West), local constraints describe the sequence of their defined multi work steps for one element within one object or the sequence of work steps between two different elements within one object. The second layer controls the fulfillment of weather impact criteria such as wind velocity for crane operation. Furthermore, it identifies weather-sensitive construction activities. The identification of required space types (material, equipment and laborers) and the availability of the required workspaces are described in the third layer. Finally, decisions concerning weather impact and the spatial constraints are proposed in the last two layers.

This multi-layer framework describes two types of interaction: *horizontal interaction* and *vertical interaction.* Vertical interaction describes the sequences between two processes in two different layers. For example, the crane operation is first checked in the weather layer. This step is achieved by comparing the current wind velocity with wind thresholds [23]. If the crane can be operated then we identify in the spatial layer the required type of defined work steps such as material and equipment spaces. If this constraint is fulfilled then material can be transported by crane to the execution positions, which is performed in the weather layer. Horizontal interaction describes sequences between two processes within one layer. For example, the processes in the last layer interact with each other horizontally. The aim of presenting the layer concept is to show how the sequences of processes within one layer interact with each other. In the same way we can show how these processes interact with different processes in different layers. Thus, different constraints of construction processes are presented as flexible, logical and transparent. The input data of this framework consists of the construction process data, weather data and spatial data. The construction process data include: the hierarchical description of project activities that are required to fulfill the final product, the assembly strategy for the different elements, their technological dependencies (local and global constraints) and date constraint, which allows the specification of the start dates of individual tasks, the required resource such as execution time, the number of workers, and the type and quantity of material. The more accurate this information is, the more reliable the results will be. Weather data is based on historical data or weather forecast data. For making weather-impact decisions, the weather thresholds and a function—which together can determine the relationship between the productivity and weather parameters—are needed. Finally, the spatial data contains the required workspace types, the level of assignment, the orientation, and reference type. Furthermore, we define two strategies in order to resolve workspace conflicts between different workspaces: the spatial adjustment strategy and the productivity adjustment strategy. For the spatial adjustment strategy, we define additional attributes for the different types of workspace such as ability to rotate, resize, and/or relocate workspace. For the productivity adjustment strategy we identify different parameters, such as overlapping areas, quality and quantity of material, actual available number of resources, etc. that affect the performance of workers on a specific task. We integrate these parameters as additional functions for conflict resolution in the simulation framework. Furthermore, these parameters are considered cumulative and so we therefore gave a weight factor for each parameter.

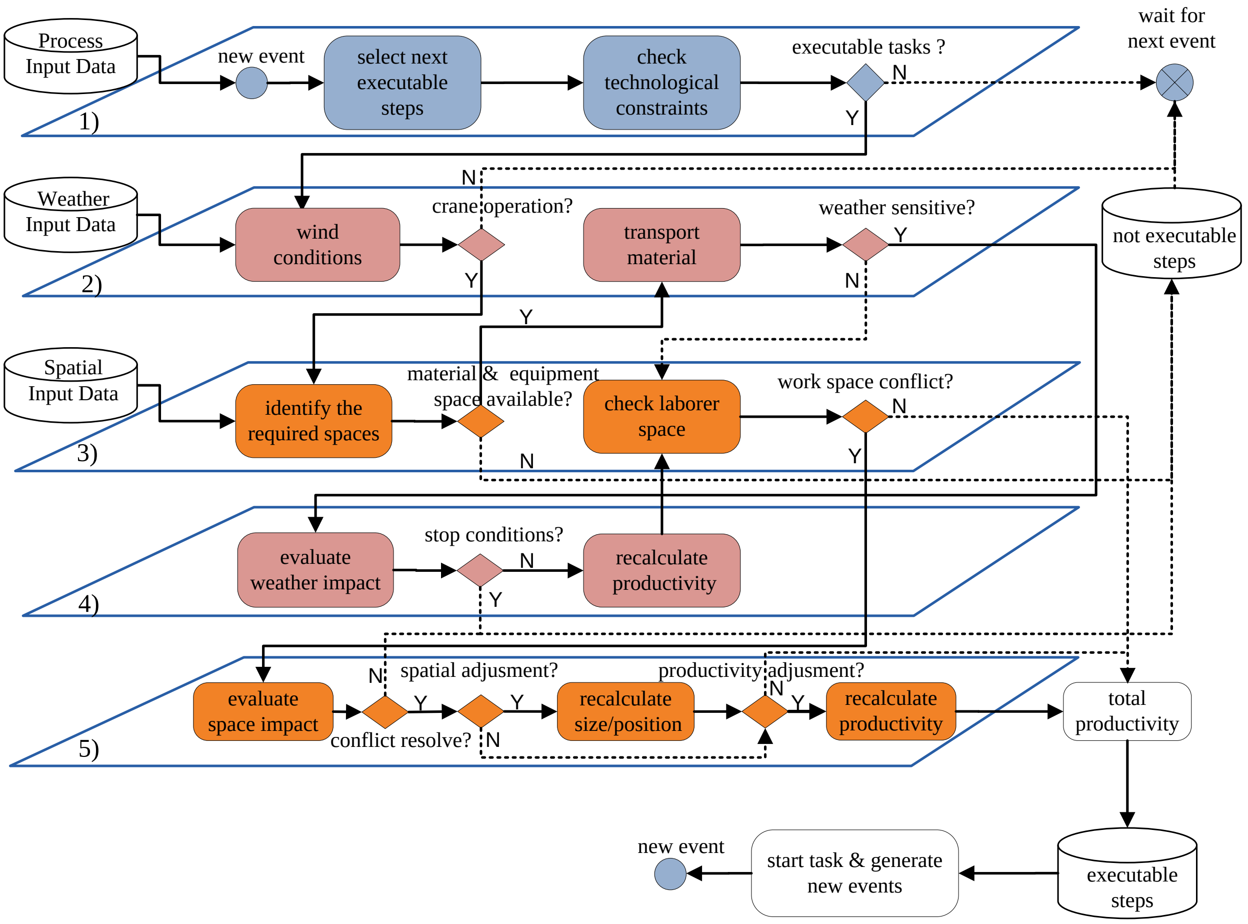


Figure 1. Multi-layer constraint and decision simulation framework

Activities are built within Plant Simulation in a hierarchical structure. The highest level of the activity description is the trade. The lowest level of the activity description is the element or section. For each element or section one or more work steps may be defined [3]. A work step has three states: “not started,” “started” and “finished,” and requires certain execution times, resources and space [19]. The procedure of the constraint-based simulation outlines the search for process steps that can be started for the current simulation time. Upon the occurrence of an event, all process steps do not start with the state of examining the performance of their technological constraints. Those process steps that meet their assigned constraints will be stored as the next executable steps. The next executable steps begin with their status “not started” and will be delivered to the developed weather constraints and spatial constraints. There, they will be checked up on the availability of workspaces and their sensitivity to weather hierarchically to verify their execution possibility to be started for the current simulation time as shown in Fig. 1.

The required resources and workspaces have to be locked during their execution. That means they cannot be used by other work steps. After locking the required resources and workspaces, the work step state changes from “not started” to “started” Subsequently, the set of “not started” work steps is checked to see if the required space and resources are available and if weather conditions are favorable by going to step one until no more work steps can be started at the current time. The simulation time is continuously checked during a simulation run. If the required time has expired, the work step status changes from "started" to "finished" and is marked as finished. Its locked resources and working spaces will be unlocked and can be used by other work steps.

V. Simple Case Study

We have validated the multi-layer simulation framework concept in a simple case study. The case study is a 5-story school building in Schwarzenberg, Germany. The construction site includes trades at the exterior construction phase. Our investigation is concentrated for trades to install the façade system and the windows.

The fiber cement façade construction is applied. Erecting a fiber cement façade system consists of the assembling of four main components: wall angle, insulation, aluminum profile and the fiber cement façade element. The installation of these four main components is decomposed into ten work steps: (1) measuring the position, (2) fixing wall angles to the building's structure; (3) measuring the position and dimension of elements, (4) cutting, (5) fixing insulation elements to wall angles and building structure; (6) measuring position and dimension, (7) cutting aluminum profiles, (8) installing aluminum profiles; (9) measuring position, (10) installing façade element. The installation of windows consists of two components: window sill and window frame, which includes four work steps: (1) measuring the position, (2) fixing the window sill; (3) leveling, and finally, (4) fixing the window frame. The execution duration of each type of work step was determined based on an expert's knowledge, which is shown in Fig. 2.

The weather input data used in this case study is the local 5-day weather forecast data. That is to say the model runs in every period of 5 days to check for the impact of weather on the construction process. The weather parameters are temperature (^{o}C), wind velocity (m/s), relative humidity (%), and precipitation (mm). The installation of the façade and window is performed through a scaffolding system. Thus the required workspace for laborer is driven due to the installation positions of the unique work steps of the elements and the width of the scaffolding platform. Since the work mainly focuses on assembling, the equipment workspace is modeled within the laborer workspace. Materials are transported from the storage area in small amounts to the installation through a crane. Aligned to the storage area a debris area is defined. An offset distance from the scaffolding is set as a hazard workspace.

In order to simulate the execution process, the constraints for assembling the façade system and windows need to be specified. The technological dependencies are represented in Fig. 2. Fixing the wall angles to the building structure, for example, needs to be finished before the measuring of the position or dimension of insulation elements can be performed. Generally, specific material and workers are required to execute a certain work step. For instance, to execute the work step "fixing wall angle", a highly skilled worker, a semi-skilled worker and a wall angle element are needed. The on-site working time of workers is defined in a simulation calendar. In this example, working days are from Monday to Friday and the daily working time is from 8:00 am to 5:30 pm including a pause of an hour at noon.

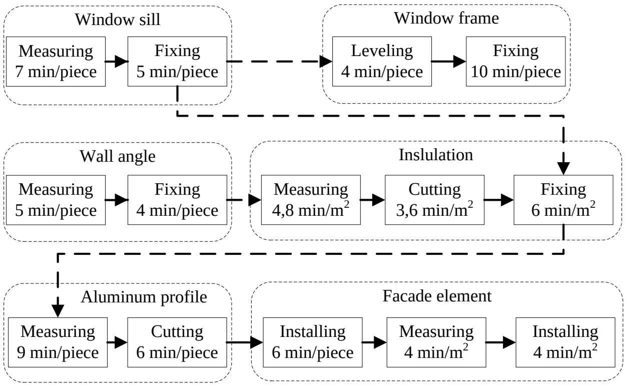


Figure 2. Technological dependencies between work steps and their non-weather-space-related execution duration

Based on the concept represented in Fig. 1, two schedules have been achieved from the simulation model: the as-built schedule and the as-possible schedule. The term "as-built schedule" is used to describe the actual schedule as constructed on site. The "as-possible schedule" expression is used to describe the schedule which considers the technological dependencies as well as the impact of weather conditions and workspace requirements.

In this example, the contract specifies a start construction date of February 12, 2008. The results of the model for the as-planned, as-built and as-possible schedule are shown in Fig. 3. The term "as-planned schedule" mentions the schedule which is prepared in the preconstruction stage.

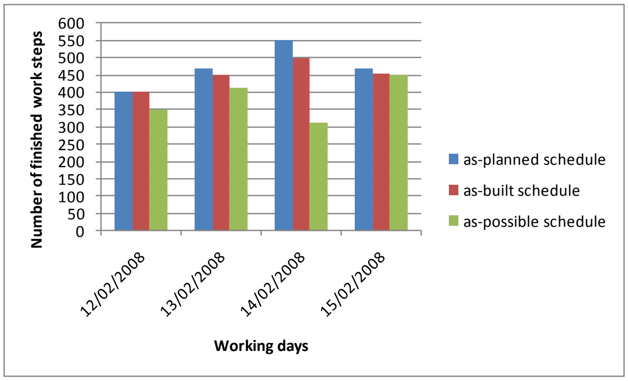
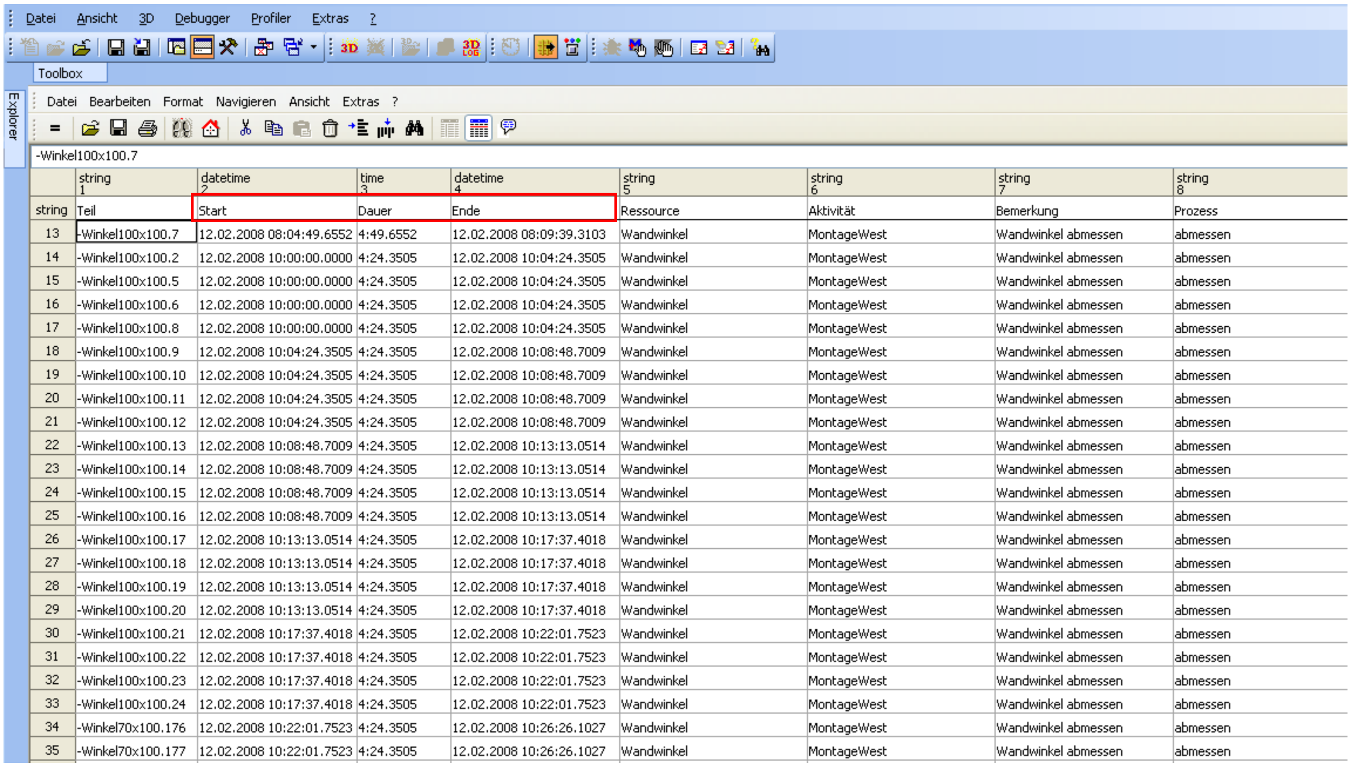


Figure 3. Output of one simulation run

The results consist of the number of work steps, which are finished in a period of 5 days. Because the first day (Feb 12) is Tuesday, the fifth day (Feb 16) is Saturday, which is a non-working day. Thus the schedules from Feb 12 to Feb 15 are achieved. Fig. 3 provides a daily comparison between the finished work steps of the as-possible, as-planned and as-built schedules for the first 4 working days. The number of finished work steps of the as-possible schedule is fewer than that of as-planned or as-built schedules. For example, on the first day, the number of finished work steps of as-planned, as-built, and as-possible schedule are 400, 400, 350 work steps respectively. During these four working days, construction operations experience delays and work steps' duration extended due to bad weather and spatial conflicts. On the first 3 days, there is heavy rain for 2 hours every day. Thus the work on the construction site is temporarily shut-down during the rain storm and the labor productivity is recalculated considering the uncomfortable conditions of temperature, humidity, wind velocity and unavailable workspace. Fig. 4 shows the work steps' execution table, including the start, duration, end point of time and specific information of the corresponding work steps.

One simulation run calculates exactly one practical execution schedule. Thus, project managers can develop different execution strategies and chose the optimal one to reduce the consequences caused by weather and space problems.



-Winkel100x100.7

	string 1	datetime 2	time 3	datetime 4	string 5	string 6	string 7	string 8
string	Teil	Start	Dauer	Ende	Ressource	Aktivität	Bemerkung	Prozess
13	-Winkel100x100.7	12.02.2008 08:04:49.6552	4:49.6552	12.02.2008 08:09:39.3103	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
14	-Winkel100x100.2	12.02.2008 10:00:00.0000	4:24.3505	12.02.2008 10:04:24.3505	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
15	-Winkel100x100.5	12.02.2008 10:00:00.0000	4:24.3505	12.02.2008 10:04:24.3505	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
16	-Winkel100x100.6	12.02.2008 10:00:00.0000	4:24.3505	12.02.2008 10:04:24.3505	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
17	-Winkel100x100.8	12.02.2008 10:00:00.0000	4:24.3505	12.02.2008 10:04:24.3505	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
18	-Winkel100x100.9	12.02.2008 10:04:24.3505	4:24.3505	12.02.2008 10:08:48.7009	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
19	-Winkel100x100.10	12.02.2008 10:04:24.3505	4:24.3505	12.02.2008 10:08:48.7009	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
20	-Winkel100x100.11	12.02.2008 10:04:24.3505	4:24.3505	12.02.2008 10:08:48.7009	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
21	-Winkel100x100.12	12.02.2008 10:04:24.3505	4:24.3505	12.02.2008 10:08:48.7009	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
22	-Winkel100x100.13	12.02.2008 10:08:48.7009	4:24.3505	12.02.2008 10:13:13.0514	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
23	-Winkel100x100.14	12.02.2008 10:08:48.7009	4:24.3505	12.02.2008 10:13:13.0514	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
24	-Winkel100x100.15	12.02.2008 10:08:48.7009	4:24.3505	12.02.2008 10:13:13.0514	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
25	-Winkel100x100.16	12.02.2008 10:08:48.7009	4:24.3505	12.02.2008 10:13:13.0514	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
26	-Winkel100x100.17	12.02.2008 10:13:13.0514	4:24.3505	12.02.2008 10:17:37.4018	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
27	-Winkel100x100.18	12.02.2008 10:13:13.0514	4:24.3505	12.02.2008 10:17:37.4018	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
28	-Winkel100x100.19	12.02.2008 10:13:13.0514	4:24.3505	12.02.2008 10:17:37.4018	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
29	-Winkel100x100.20	12.02.2008 10:13:13.0514	4:24.3505	12.02.2008 10:17:37.4018	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
30	-Winkel100x100.21	12.02.2008 10:17:37.4018	4:24.3505	12.02.2008 10:22:01.7523	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
31	-Winkel100x100.22	12.02.2008 10:17:37.4018	4:24.3505	12.02.2008 10:22:01.7523	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
32	-Winkel100x100.23	12.02.2008 10:17:37.4018	4:24.3505	12.02.2008 10:22:01.7523	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
33	-Winkel100x100.24	12.02.2008 10:17:37.4018	4:24.3505	12.02.2008 10:22:01.7523	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
34	-Winkel70x100.176	12.02.2008 10:22:01.7523	4:24.3505	12.02.2008 10:26:26.1027	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen
35	-Winkel70x100.177	12.02.2008 10:22:01.7523	4:24.3505	12.02.2008 10:26:26.1027	Wandwinkel	MontageWest	Wandwinkel abmessen	abmessen

Figure 4. Statistic of the execution process

VI. Conclusion and Future Work

Construction processes are complicated by nature and usually disturbed by many factors. These factors can affect progress individually or simultaneously, in the latter case they can create even worse consequences. The requirement to provide decision support systems to consider simultaneously different affected factors is necessary, which has not been researched enough in previous studies. Using simulation models to research construction problems is proven to be effective and convenient. This paper provides a multi-layer simulation framework which considers technological and spatial constraints as well as the impact of weather for construction planning. These aspects are described as hard and soft constraints in construction processes. This approach provides a flexible way to consider weather and spatial aspects where the model can easily be adapted by adding or removing constraints. Moreover, layers of this framework can interact flexibly with each others to ensure that the impact of spatial conflicts and bad weather conditions can be considered hierarchically. Based on this concept, more different influencing factors of construction processes can be integrated in the same model.

Furthermore, in the next research steps, based on weather-spatial-impacted results, alternative strategies to reduce the consequences can be provided and analyzed using this simulation model. Therefore, it is easier for managers to make the right decisions when encountering weather and spatial problems.

REFERENCES

[1] H.-J. Bargstädt and A. Elmahdi, "Simulation von Bauprozessen - ein Qualitätssprung für die Arbeitsvorbereitung," 8. Grazer Baubetriebs- und Bauwirtschaftssymposium, Tagungsband 2010 - Arbeitsvorbereitung für Bauprojekte, Nutzen der Arbeitsvorbereitung für den Projekterfolg, vol. 2010, pp. 131 - 146.

[2] S. Castro and N. Dawood, "RoadSim: Simulation Modelling and Visualisation in Road Construction," Conference on construction application of virtual reality, Lisbon, 2004, pp. 33-42.

[3] H.-J. Bargstädt and A. Elmahdi, "Automatic Generation of workspace Requirements Using Qualitative and Quantitative Description," 10th International Conference on Construction Applications of Virtual Reality CONVR2010 Organizing Committee, Japan, Sendai 2010, pp. 131-137.

[4] D.W. Halphin and L.S. Riggs, "Planning and Analysis of Construction Operations." 1992, Canada: Wiley-Interscience.

[5] G. Heinicke, "Technologie des Ausbaus," 2., durchges. ed. Lehrbuchreihe Technologie der Bauproduktion Technische Hochschule Leipzig. 1982, Berlin: Verl. für Bauwesen.

[6] N.B. Benjamin and T.W. Greenwald, "Simulating effects of weather on construction," Journal of Construction Engineering and Management, vol. 99, 1973, pp. 175-190.

[7] D.R. Riley and V.E. Sanvido, "Patterns of Construction-Space Use in Multistory Buildings," Journal of Construction Engineering and Management, vol. 121, 1995, pp. 464-473.

[8] D.R. Riley and V.E. Sanvido, "Space Planning Method for Multistory Building Construction," Journal of Construction Engineering and Management, vol. 123, 1997, pp. 171-180.

[9] B. Akinci, M. Fischer, and J. Kunz, "Automated Generation of Work Spaces Required by Construction Activities," Journal of Construction Engineering and Management, vol. 128, 2002c, pp. 306-315.

[10] R. Akbas, "Geometry-Based Modeling and Simulation of Construction Processes," 2004.

[11] E. Koehn and G. Brown, "Climatic Effects on Construction," Journal of Construction Engineering and Management, vol. 111, 1985, pp. 129-137.

[12] H.R. Thomas and I. Yiakoumis, "Factor Model of Construction Productivity," Journal of Construction Engineering and Management, vol. 113, 1987, pp. 623-639.

[13] O. Moselhi and K. El-Rayes, "Analyzing Weather-Related Construction Claims," Cost Engineering vol. 44, 2002, pp. 12-19.

[14] H.-S. Lee, H.-G. Ryu, J.-H. Yu, and J.-J. Kim, "Method for Calculating Schedule Delay Considering Lost Productivity," Journal of Construction Engineering and Management, vol. 131, 2005, pp. 1147-1154.

[15] A. Shahin, S. AbouRizk, Y. Mohamed, and S. Fernando, "A simulation-based framework for quantifying the cold regions weather impacts on construction schedules," Proceedings of the 39th conference on Winter simulation, IEEE Press, Washington D.C., 2007, pp. 1798-1804.

[16] S. Kenner, R.L. Johnson, J.R. Miller, J.A. Salmen, and S.A. Matt, "Development of Working Day Weather Charts for Transportation Construction in South Dakota," Study SD97-07, Technical Report, South Dakota Department of Transportation, Pierre, SD., 1998.

[17] D. Atef, H. Osman, M. Ibrahim, and K. Nassar, "A simulation-based planning system for wind turbine construction," Proceedings of the 2010 Winter Simulation Conference, Baltimore, Maryland, 2010, pp. 3283-3294.

[18] ForBau, "Modellierung der Wettereinflüsse auf den Erdbau," in Forschungsverbund: Virtuelle baustelle-Digitale Werkzeuge für die Bauplanung und -abwicklung, Abschlussbericht. 2010. pp. 117-120.

[19] U. Beißert, M. König, and H.-J. Bargstädt, "Constraint-Based Simulation of outfitting processes in Building Engineering," CIB 24th W78 Conference, Maribor, Slovenia 2007, pp. 491-497.

[20] U. Beißert, M. König, and H.-J. Bargstädt, "Considering quality aspects for construction scheduling using Constraint-Based Simulation," CIB W078 & MC4T - Managing IT in Construction., A.A. Balkema, Istanbul, Turkey, 2009, pp. 1-9.

[21] J.K. Voigtmann and H.J. Bargstädt, "Logistic Strategies for Construction Processes," Proceeding of the IABSE ICT Conference, Helsinki, Findland, 2008a, pp. 1-8.

[22] J.K. Voigtmann and H.J. Bargstädt, "Simulation of construction logistics in outfitting processes," EWork and EBusiness in Architecture, Engineering and Construction: ECPPM, 2008b, pp. 195-203.

[23] H.H. Le and H.-J. Bargstädt, "A simulation approach to integrate weather impact into the execution planning," Proceedings of the 27th CIBW78 - Applications of IT in the AEC Industry, Cairo, Egypt, 2010a, pp. 1-10.

[24] H.H. Le and H.-J. Bargstädt, "Simulation des Einflusses von Witterungsbedingungen auf die Bauausführung," 14th ASIM - Integration Aspects of Simulation: Equiptment, Organization and Personnel, KIT Scientific Publishing, Karlsruhe, Germany, 2010b, pp. 125-132.

[25] E. Sriprasert and N. Dawood, "Requirements identification for 4D constraint-based construction planning and control system," Proceedings of CIB w78 Conference, Aarhus, Denmark, 2002, pp. 1-8.

[26] M. König, U. Beißert, D. Steinhauer, and H.-J. Bargstädt, "Constraint-Based Simulation of outfitting processes in Shipbuilding and Civil Engineering," 6th EUROSIM Congress in Modelling and Simulation, Ljubljana, Slovenia, 2007, pp. 1-11.

Agent-based simulation validation: A case study in demographic simulation

Cristina Montañola-Sales*‡, Bhakti S. S. Onggo† and Josep Casanovas-Garcia‡
*Computer Applications in Science and Engineering
Barcelona Supercomputing Center, Barcelona, Spain
Email:cristina.montanola@bsc.es
†Department of Management Science
Lancaster University Management School, Lancaster, UK
Email: s.onggo@lancaster.ac.uk
‡Department d'Estadística i Investigació Operativa (DEIO)
Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
Email: josep.casanovas@upc.edu

***Abstract*—Two of the crucial parts in the process of performing a simulation study are validation and verification. The reason is these techniques help on increasing the confidence in the model, since it is not possible to demonstrate its absolute validity in all contexts. This paper presents the results of a white-box validation performed in an agent-based simulator for population dynamics. The tool provides a way to simulate the demographic evolution of large populations in a parallel environment. The purpose is to obtain population projections that can be used afterwards for policy analysis. Although the tool has been studied in terms of performance and scalability, its validation hasn't been addressed. With a white-box validation we expect to increase the confidence of policy analysers and social scientists in our simulation model.**

***Keywords*-White-box Validation; Agent-based Simulation; Parallel Simulation; Demography**

I. Introduction

Agent-based modelling is a model that is formed by a set of autonomous agents that interact with their environment (including other agents) through a set of internal rules to achieve their objectives [1]. An agent-based model, just like other types of model, is used to represent a real world system and to help us understand the system and to make decisions. An agent-based model is commonly implemented as a piece of computer code and run using a simulator. Agent-based simulation is the computer implementation of an agent-based model. Agent-based simulation has been applied in the physical sciences as well as the social sciences [2]. Many agent-based simulation tools have been developed in the last years to explore the complexity of social systems. Social phenomena are unpredictable and changing (dynamic). For this reason, agent-based simulation allows us to carry out experiments and studies that would not be feasible otherwise [3].

Agent-based simulation is recognised as one of the techniques which could contribute more in understanding complex social systems [4]. One of the application areas is demography. Onggo [5][6] has developed a parallel simulation tool for demography. Demography is often used as one of the important considerations in policy analysis and planning. The parallel simulation tool (Yades) was built for discrete-event simulation modelling paradigm [5][6]. In this paper, we have refactored the tool to separate the modelling component from the simulation execution component. The objective is to allow users to model population dynamics using agent-based simulation modelling paradigm. The agent-based model will be run on top of the parallel discrete-event engine. This paper reports the work that we have carried out to evaluate the correctness of the refactored simulation tool.

Validation and verification (V & V) is a significant element of any simulation study. As pointed by Robinson [7], "without V & V there are no grounds on which to place confidence in a (simulation) study's results". In simulation, we often differentiate between verification and validation. Verification is a process to determine whether a conceptual model has been implemented correctly in its computerized form. To borrow the computer programming term, we debug the model. Validation is a process to determine whether the model is an accurate representation of the system being studied for a given set of modelling objectives. Robinson states that it is not possible to prove that a model is valid in all contexts, because a model is only a simplified version of a real system. Consequently, a model cannot describe all aspects of a real system. Hence, the main objective of validation is to prove that a model is *sufficiently* accurate for parts of reality that is being studied. Indeed, one of the key aspects of validation is to assess whether the outcomes of a model can explain the real phenomenon under study [8]. This can be fulfilled by performing as many validation methods as possible during a simulation study until we (and users) can gain enough confidence in the model and accept its results. Edmonds [9] describes validation as a continuous process. Validation should also take into account the domain of the system under study [10]. Therefore, a validated model may not be valid for a set of different experimental conditions outside

ISBN: 978-1-61208-169-4

its domain.

Robinson identifies four different forms of validation in simulation modelling: conceptual model validation, data validation, white-box validation and black-box validation [7]. Conceptual model validation deals with issues such as the level of detail of the model and determines if it is enough for the purpose it was developed. Data validation is needed to determine whether the data used in the simulation study is sufficiently accurate. The black-box validation concerns with the relationship between inputs to the model and its outputs, ignoring the elements inside a model. The objective is to determine if the output of the model reflects the real world observation for the same set of inputs. Finally, white-box validation tries to answer the question *does each element of the model and the structure of the model elements represent the real world with sufficient accuracy?*

This paper reports our work in the validation of the agent-based simulation tool which has not been reported in our previous work. The validation is based on the white-box validation methods described in Pidd [11]. The rest of the paper is organised as follows. Section II presents an overview of the demographic simulation model that is used in this paper. Section III describes the simulation tool and Section IV presents the verification and validation work. Finally, our concluding remarks and lines of further work are described in Section V.

II. Demographic agent-based model

In demography the most commonly used paradigms are microsimulation, system dynamics and discrete-event simulation. In microsimulation, we need to specify a random sampling process for each individual at every simulation time point. On the other hand, in system dynamics, we do not keep track changes in the state of each individual but focuses on the population of individuals and the rates of individuals moving from one state to another. Similar to microsimulation, in discrete-event simulation, we keep track the individuals starting from their arrival in the system (through births and migrations) until they leave the system (through deaths and migrations). However, discrete-event simulation does not inspect each individual at every simulation time point. It inspects an individual only when the state of the individual changes.

It is commonly accepted that agent-based simulation can help to better understand a complex system where there is a need to model behaviours of many interacting individuals [12]. Agent-based modelling paradigm allow us to explicitly include human behavioural aspects into a model. This is one of the main reasons that motivates us to support the use of agent-based modelling paradigm in our demographic simulation tool. At the very core of a demographic model, we need to model key demographic components that represent basic population dynamics, such as: fertility (births), marital status, migrations, and mortality. On top of this, we can add components depending on the intended application of the demographic model. To take one example, for the application in tax and benefit systems, we may need to add another component that represents the change in economic status. With the agent-based modelling paradigm, we can evaluate the effect of a certain behaviour at the individual level on the population. This will make our tool more useful to wider potential users.

In Yades modellers can specify a model for each demographic component which will form a bigger model that represents the interactions between all components in a population. The detailed explanation on each demographic component can be read from our previous reported works [5] and [6]. The summary is as follows. The fertility model concerns with the representation of the number of children that a female individual may have, the age of the female individual when her first child is born, and the time between two consecutive births for female individuals who will have more than one child. The mortality model is used to represent the lifetime of an individual. The migrations model represents the mobility of individuals in a population. The marital status model is used to model the change in the marital status of individuals. Similarly, the economic status model represents the change in the economic status of individuals in the population.

III. Yades: A parallel simulator for population dynamics

Our parallel demographic simulation tool is called Yades. We have refactored the tool to allow modellers to model the individuals in a society using the agent-based simulation modelling paradigm and to run the model on top of a DES engine. The idea of running an agent-based model on top of a discrete-event simulator has been proposed by some writers [1][2][13]. It is one of the approaches that has been proposed to tackle the scalability issue of large-scale agent-based simulation models. The main advantage of this approach is that modellers who prefer to use agent-based modelling paradigm, do not need to change their modelling paradigm, and at the same time, a scalable parallel discrete-event simulation engine can be used to improve the overall simulation performance.

The simulation engine in Yades is implemented using μsik parallel simulation library. μsik is a parallel discrete-event simulation library that supports multiple synchronisation algorithms such as: lookahead-based conservative protocol and rollback-based optimistic protocol [14]. This library adopts the process interaction world-view in which a simulation model is formed by a set of interacting (logical) processes. Logical processes (LPs) communicate through events. Multiple LPs can be mapped onto a physical

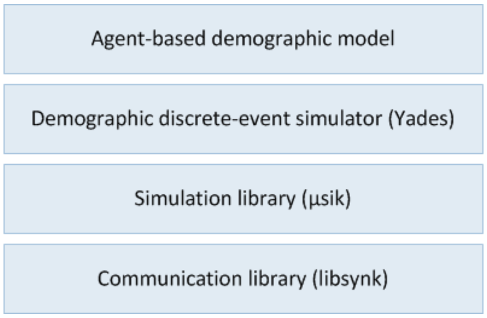

Agent-based demographic model
Demographic discrete-event simulator (Yades)
Simulation library (μsik)
Communication library (libsynk)

Figure 1. Software architecture

process (PP) that is run on top of a processing element (PE). A machine can have more than one PE (e.g., in multi-core architecture). The detailed description of the implementation can be read from [5] and [6].

Yades maps each agent in the model onto an LP. There are two types of agents in Yades. The first type of agent represents a family unit. In FRS data, a family unit is defined as a single independent individual or two independent individuals living together (as married or in cohabitation) and any dependent individuals (children). Hence, a family unit may represent an independent individual, a single parent, a childless couple, a nuclear family or an orphan. The main advantage of representing a family unit as an agent is that many public policies may apply to individuals as well as groups of related individuals, such as households and single parents. The behaviour of a family unit is defined by the five demographic components, i.e. fertility, mortality, change in economic status, change in marital status, and migration. Yades provides the placeholders for each demographic components where detailed behaviour can be specified by modellers. The second type of agent represents an administrative area where a number of families live. This agent will handle migrations and changes in simulation parameters and produce periodic reports. Yades allows users to model administrative area with different population characteristics. The main limitation of the current version is that it only allows one processing element to run one administrative area. The architecture of our tool is shown in Figure 1.

Yades allows users to provide data for the initial population. The data used in the model in this paper follows the structure of the UK Family Resources Survey (FRS) data. FRS is sponsored by the UK Department for Work and Pensions. It has been running since 1992 which provides useful cross-sectional and longitudinal data for the simulation. Hence, we can set the initial population parameters from a readily available data.

IV. Model verification and validation

A. Model verification

Although conceptually simple, verification can be challenging, especially when we are dealing with a relatively complex computer program. Law [15] and Banks et al. [16] lists a number of techniques that can be used in a verification process.The main technique that we use is the structured walkthrough of the program. This includes dividing the model into smaller components and test the correctness of each component. It is suggested that we start with the simplest possible behaviour so that the simulation output can easily be understood. Hence, errors can be easily spotted. This is the approach that we have used. We have tested the implementation of a simple model for each demographic component in our model. The explanation is given in the following subsections.

Another technique that we have used is the structured walkthrough of the program. The program developer (second author) presented the code line-by-line to the first author. It was done in a small meeting room with a projector showing the computer code. Before the structured walkthrough session started, the first author had been briefed with the conceptual model. We have found this technique very effective to uncover flaws in the computer implementation of the conceptual model. The most obvious explanation is because the non-developer(s) are independent and they may view the implementation from different perspectives. Hence, they can challenge the developer(s) on various implementation issues, such as the effectiveness of an implementation, the possible settings that can cause errors, and the correctness of the implementation. In the context of agent-based simulation, we have found this method especially effective, because of the many possible combination of interactions between agents in the model. When a conceptual model document was given to the non-developer(s) before a structured walkthrough session started, it gives a top level view of the model to the non-developer(s). Hence, the non-developer(s) focus more on the top-level view of the model and less distracted with the implementation detail. This has the potential to uncover the possible errors due to the combinations of interactions between agents that might have been missed by the model developer(s). We use Yades' facility to produce a trace to be examined to check any possible mistakes. This was done by the first author to minimize the developer's bias.

B. Model validation

From the 1990s, agent-based simulation has become increasingly popular [17]. However, according to the survey conducted by Heath et al. [17] on the articles related to agent-based models published between 1998 and 2008, 29% of the articles did not discuss the validation of their models.

They further divide the validation reported in the articles into two categories: conceptual (i.e. conceptual model validation) and operational (i.e. comparing the simulation result with the real observation). They found that 17% of the articles used the conceptual validation only, 19% used the operational validation only, and 35% used both. They also noted the dominance of qualitative validation methods in the validation of agent-based models. They provide a conjecture that this might be because many agent-based models are not conducive for quantitative validation methods. Klügl noted that agent-based models often exhibit behaviour that can be problematic for validation purposes, such as non-linearities and multi-level properties [18]. In addition, agent-based models often use significantly more assumptions which make the assessment of the validity of assumptions more difficult. Agent-based models also require the finer level of model detail in which data at that level of detail may be difficult to obtained.

Duong [19] also examines this issue and suggests that the greater uncertainty in social sciences compare to others, the lack of consensus on how to represent social environment, and the lack of experimental controls in data collection might contribute to the difficulties in the validation of agent-based models. Windrum et al. [20] examines a set of methodological problems in the empirial validation of agent-based models. The problems seem to have arisen due to, among other reasons, the lack of techniques to build and analyse these models and the lack of comparability between the ones which have already been developed. A number of validation techniques have been proposed for agent-based simulation modelling. Klügl [18] proposes a validation process for ABS models combining face validation and statistical methods. Moss et al. [21] use a declarative formalism to address the validation and verification of ABM with cognitive agents. However, there seems to be a general concern on the lack of validation framework or methodology in agent-based simulation.

In this paper we present a white-box validation of our simulation model based on the methods described in Pidd [11], especially on the static logic and the dynamic logic of the model. A white-box method focuses on the correctness of the internal workings of a model. This includes the correctness of the components and the interaction between components. A white-box method assumes that we know (and have access to) the components inside the model. Balci [22] defines white-box method as a technique that is intended to evaluate a model based on its execution behaviour. It can be applied to the programmed model (verification) or to the experimental model (validation) of the life-cycle process of a simulation study. We have applied this method to asess the correctness of the internal working of each component in the model introduced in Section III.

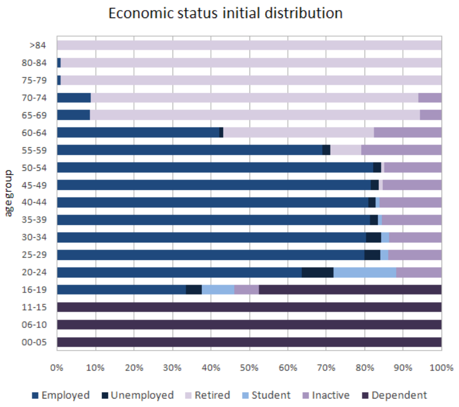


Figure 2. Initial distribution of economic status in the population

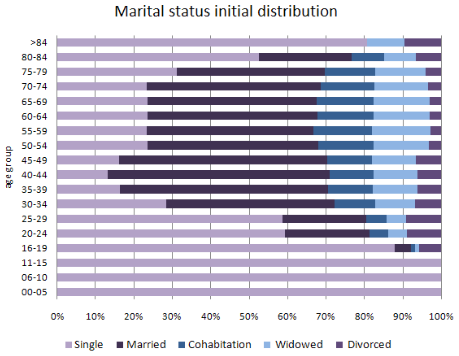


Figure 3. Initial distribution of marital status in the population

We divide the model into smaller components and test the correctness of each component. It is suggested that we start with the simplest possible behaviour so that the simulation output can easily be understood. Hence, errors can be easily spotted. This is the approach that we have used. We have tested the implementation of a simple model for each demographic component in our model.

In the following subsections, we present the result of our evaluation on each model component using the white-box method. In the evaluation, we use a population of 110,000 family units. In each test, we run the simulation five times and report the average results . The explanation on the validation of each demographic components is as follows. Figure 2 shows the proportion of different economic statuses in the initial population by age group. Similarly, Figure 3 shows the proportion of different marital statuses in the initial population by age group.

1) Mortality: Yades allows modellers to sample the lifetime of individuals using two commonly used methods: life table and survival function. In order to evaluate this component, we disable all other demographic components. This helps us to detect any error and to isolate the root cause of the error easily. We vary the life tables. One of

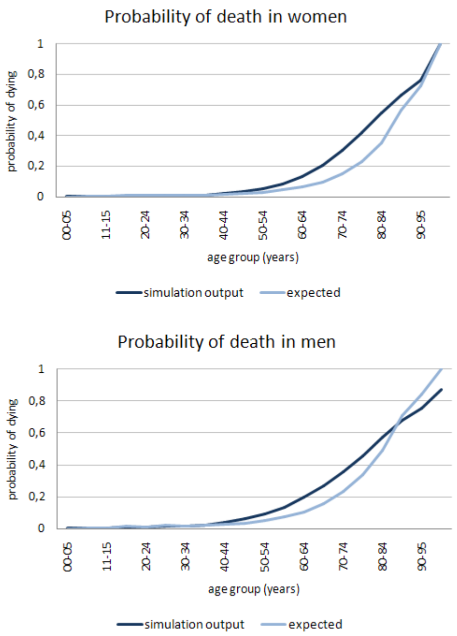


Figure 4. Simulation output and expected output for mortality model

the results is shown in Figure 4. In both cases, the Pearson product moment correlation coefficient of the original distribution and the outputs is very high, 0.9870 for women and 0.9783 for men. The same evaluation is repeated using various life tables. They also produce high correlation values. This has increased our confidence that the mortality component can produce the intended behaviour.

2) Fertility: Modellers can specify a number of fertility models in Yades using age-specific fertility model, parity-specific fertility model, birth spacing model and their combinations. To test the fertility model component, the rest of demographic components are disabled in order to isolate fertility results. To simplify the model, birth function is set to follow a Poisson distribution with parameter $\lambda = 2$ in women from 16 to 49 years old (assumed to be the reproductive age). Birth function is calculated for every woman regardless their marital status and birthspacing is uniformly distributed. After running the simulation the accumulated number of births by age group is obtained. In Figure 5 the percentages of births are represented according to the number of children's group. As we can see, the simulator is producing the expected number of births in the fertility interval. We repeat this experiment with different parameters and all of them produced the expected results.

3) Marital status: Yades recognizes the following marital statuses: single, married, cohabitation, separated, divorced and widowed. Individuals will move from one marital status to another during their lifetime. The transitions from one status to another can be specified based on a simple probability function, a regression function or a set of logical rules. Likewise, time spent in one status can be sampled using a distribution function, a regression function or a set of logical rules. In the formation of a family unit (e.g., marriage and cohabitations), we need to specify a function that matches a pair of individuals.

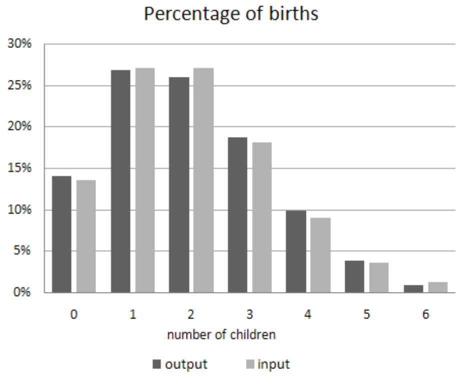


Figure 5. Simulation output and expected output for fertility model

In the following evaluation we use a probability function for the state transition and apply a simple matching criteria where we choose the first person that we find in the list regardless of his/her characteristics. Transitions are uniformly distributed between 1 and 10 years. First, we want to test the correctness of family unit formations. Hence, we disable all other transitions. We also disable all other life events (as fertility and mortality). The top chart in Figure 6 shows the result. As expected, the number of marriages and cohabitations increases with time while the number of singles decreases progressively.

The second test has the same settings as before but we enable the mortality in the model. The result can be seen from the bottom chart in Figure 6. The figure shows that the number of widowed increases with time. This is the behaviour that we expect.

4) Economic status: As in the marital status, an individual may move from one economic status to another during his/her lifetime. Yades recognizes the following economic statuses: dependent, in employment, unemployed, in full-time higher education, pension and economically inactive. Modellers will need to model the transitions from one status to another and the time spent in any of the status. In the evaluations, the transitions are scheduled using a uniform distribution between 1 and 5 years. The top chart in Figure 7 shows the changes in economic status. As expected, without mortality, the number of pensioners increases steadily. At the same time, the number of dependent individuals, individuals in higher education, working individuals, and unemployed individuals decreases over time.

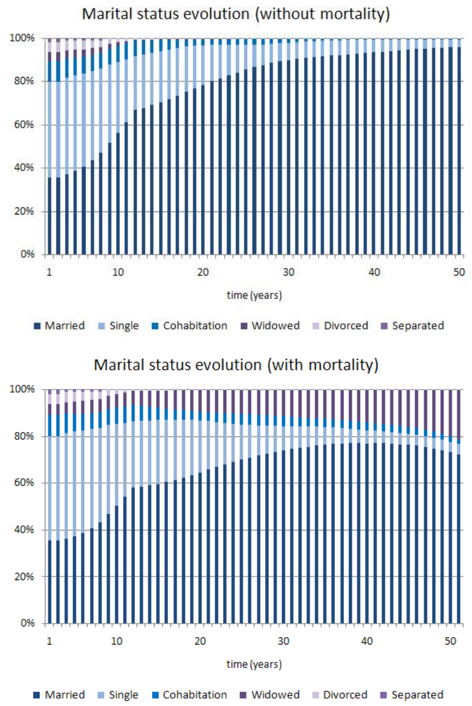


Figure 6. Validation results for marital status component model

In the second test, we use exactly the same setting but we enable the mortality. This should produce a similar behaviour as before but the proportion of retired individuals will be less because some of them will die. The result can be seen from the bottom chart in Figure 7.

5) *Migrations:* Yades provides a functionality for a modeller to define a model that determines whether a family unit is going to migrate. There are two types of migration: domestic migration and international migration (emigration and immigration). These models can be specified using a constant probability, regression or a set of logical rules. The destination region is determined using a probability matrix where each row represents the originating region and each column represents the destination. To validate this component we tried two different scenarios using 4 regions. In the first scenario, we set the probability to migrate in each regions to be the same. As expected, the number of population in each regions are relatively the same [6]. In the second scenario we set one of the regions (i.e., region 4) to be the most attractive, such that once people have move to that region they will never leave the region. In this example, we expect an increase in the population of region 4 while the rest of the regions experience a decrease in their population. The result is shown in Figure 8.

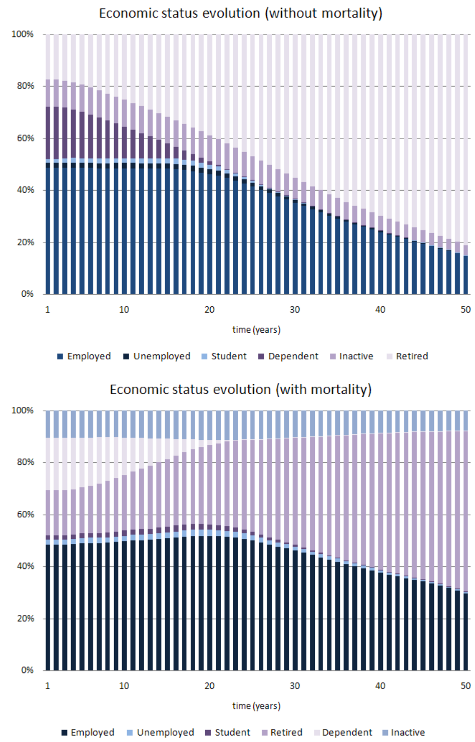


Figure 7. Validation results for economic status component model

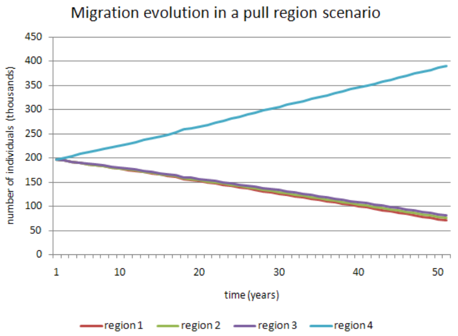


Figure 8. Evolution of population in a migration scenario

V. Conclusion and Future Work

The validation of a complex agent-based model is challenging. This is partly due to the quality of the available data that are needed to calibrate and to validate the model. The large number of model parameters makes it even more challenging. In this paper, we have presented the verification and validation of an agent-based demographic simulation model implemented using Yades using white-box method. This method allows us to assess the correctness of the model components and their interactions. The results obtained in this paper show that the five components of the simulator are behaving correctly in terms of what the modellers should expect from them for the given scenarios. To increase our confident in the model, we need to conduct more testing using different validation methods. At the moment, we are implementing a graphical user interface to help users specify the model more easily without having to write the codes. This would help potential users who do not have any programming experience to test their models and provide feedback on the tool.

Acknowledgment

This research is supported by the Royal Society International Joint Project 2009/R2 grant number JP090402.

References

[1] B. Onggo, "Running agent-based models on a discrete-event simulator," in *Proceedings of the 24th European Simulation and Modelling Conference*. Hasselt, Belgium: Eurosis-ETI, October 25-27 2010, pp. 51–55.

[2] C. M. Macal and M. J. North, "Tutorial on agent-based modeling and simulation," in *Proceedings of the 37th Conference on Winter Simulation*. IEEE, Inc., 2005, pp. 2–15.

[3] J. Pavon, M. Arroyo, S. Hassan, and C. Sansores, "Agent-based modelling and simulation for the analysis of social patterns," *Pattern Recognition Letters*, vol. 29, no. 8, pp. 1039–1048, 2008.

[4] G. Gilbert, *Agent-based models*, ser. Quantitative Applications in the Social Sciences. Sage Publications, Inc., 2008.

[5] B. Onggo, "Parallel discrete-event simulation of population dynamics," in *Proceedings of the 40th Conference on Winter Simulation*. IEEE, Inc., 2008, pp. 1047–1054.

[6] B. Onggo, C. Montañola-Sales, and J. Casanovas-Garcia, "Performance analysis of parallel demographic simulation," in *Proceedings of the 24th European Simulation and Modelling Conference*. Hasselt, Belgium: Eurosis-ETI, 2010, pp. 142–148.

[7] S. Robinson, "Simulation model verification and validation: increasing the users' confidence," in *Proceedings of the 29th Conference on Winter Simulation*. IEEE Computer Society, 1997, pp. 53–59.

[8] P. Ormerod and B. Rosewell, "Validation and Verification of Agent-Based Models in the Social Sciences," in *EPISTEMOLOGICAL ASPECTS OF COMPUTER SIMULATION IN THE SOCIAL SCIENCES - SECOND INTERNATIONAL WORKSHOP, EPOS 2006* , ser. Lecture Notes in Artificial Intelligence, Squazzoni, F, Ed., vol. 5466, Lucchini Fdn; European Social Simulat Assoc. HEIDELBERGER PLATZ 3, D-14197 BERLIN, GERMANY: SPRINGER-VERLAG BERLIN, 2009, Proceedings Paper, pp. 130–140, 2nd International Workshop on Epistemological Perspectives on Simulation, Brescia, ITALY, OCT 05-06, 2006.

[9] B. Edmonds, "The use of models - making MABS more informative," in *Second International Workshop, MABS 2000*. SpringerVerlag, 2000, pp. 15–31.

[10] R. Sargent, "Verification and validation of simulation models," in *Proceedings of the 37th Winter Simulation Conference*. IEEE, Inc., 2005, pp. 130–143.

[11] M. Pidd, *Computer simulation in management science*, 5th ed. John Wiley & Sons, Inc. Chichester, England, 2004, chapter 10.

[12] P. Siebers, C. Macal, J. Garnett, D. Buxton, and M. Pidd, "Discrete-event simulation is dead, long live agent-based simulation!" *Journal of Simulation*, vol. 4, no. 3, pp. 204–210, 2010.

[13] M. Hybinette, E. Kraemer, Y. Xiong, G. Matthews, and J. Ahmed, "Sassy: A design for a scalable agent- based simulation system using a distributed discrete event infrastructure," in *Proceedings of the 36th Conference on Winter Simulation (Monterey, California)*, 2006, pp. 926–933.

[14] K. Perumalla, "μsik: A Micro-Kernel for Parallel/Distributed Simulation Systems," in *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*. IEEE Computer Society Washington, DC, USA, 2005, pp. 59–68.

[15] A. Law and W. Kelton, *Simulation modeling and analysis*, 4th ed. McGraw-Hill New York, 2007, chapter 5.

[16] J. Banks, J. Carson, B. Nelson, and D. Nicol, *Discrete-event simulation*. Prentice-Hall, 1999, chapter 10.

[17] B. Heath, R. Hill, and F. Ciarallo, "A survey of agent-based modeling practices (january 1998 to july 2008)," *Journal of Artificial Societies and Social Simulation*, vol. 12, no. 4, p. 9, 2009.

[18] F. Klügl, "A Validation Methodology for Agent-Based Simulations," in *Proceedings of the 23rd Annual ACM Symposium on Applied Computing*, SIGSAC. NY,USA: ACM, 2008, Proceedings Paper, pp. 39–43, 23rd Annual ACM Symposium on Applied Computing, Fortaleza, Brazil, Mar 16-20.

[19] D. Duong, "Verification, Validation, and Accreditation (VV&A) of Social Simulations," in *Spring Simulation Interoperability Workshop, Orlando, April 12-16 2010*, 2010.

[20] P. Windrum, G. Fagiolo, and A. Moneta, "Empirical Validation of Agent-Based Models: Alternatives and Prospects," *The Journal of Artificial Societies and Social Simulation*, vol. 10, no. 2, 2007.

[21] S. Moss, B. Edmonds, and S. Wallis, "Validation and Verification of Computational Models with Multiple Cognitive Agents," Manchester Metropolitan University, Centre for Policy Modelling, Discussion Papers 97-25, 1997.

[22] O. Balci, "Validation, verification, and testing techniques throughout the life cycle of a simulation study," *Annals of Operations Research*, vol. 53, no. 1, pp. 121–173, 1994.

Analysis and Simulation of Power Law Distribution of File Types in File Sharing Systems

Yuya Dan
Faculty of Business Administration
Matsuyama University
Bunkyo 4-2, Matsuyama, Ehime 790-8578, Japan
Email: dan@cc.matsuyama-u.ac.jp

Takehiro Moriya
CTO and R&D Headquarters
Branddialog, Inc.
Minato 3-5-10, Chuo-ku, Tokyo 104-0043, Japan
Email: moriya@branddialog.co.jp

***Abstract*—We study the distribution of file types classified by file extensions in usual file systems. In this paper, we report that the power-law distribution is observed in a certain file system and try to give the answer to the mechanism of that formation. In order to recognize the phenomena, we construct mathematical models and compare them to the results of Monte Carlo simulation. Then, we propose that file operation of creation and copy would form the distribution at the conclusion. This paper focuses on the formation of the power-law distribution by mathematical analysis and computer simulation.**

***Keywords*-power-law distribution; mathematical modeling; Monte Carlo simulation; file operation; scale-free structure**

I. Introduction

In a variety of sciences, we observe that the value distributes a typical value around the average which individual measurements are centered. Gaussian distributions are often obtained when scientists measure their targets. Despite of Gaussian distributions, there are binomial, Poisson and power-law distributions in observed scientific data. In particular, power-law distributions in the observation has no typical value as averages, so that we also call scale-free structure.

There are many examples of distributions that obey power-law in natural, social, and other sciences. We know Gutenberg-Richter's law as the sizes of earthquakes [13], Zipf's law as the frequency of use of words in any human language [27], the numbers of papers scientists write [15], the number of citations received by papers [20], the number of hits on web pages [1], structure of WWW traffic [7], people's annual incomes [19], the sales of music recordings [5], the frequency of opening moves in chess [4], and so on. See also city populations and the property of power-law phenomena [16] more in detail. Clauset et. al. [6] gave a concise statistical method for analysis of power-law phenomena.

Mathematicians and physicists would believe that comparative simple principles form complex structure in these fields, and try to recognize the essential framework of models they proposed. In fact, We know that chaotic phenomena often occur even in the simple system. It is natural that complex systems are made from the simple principles.

In this paper, we propose a model for file operation process as a complex system, then provide a simulation result based on the model. File operation process is one of human-computer interaction in our ordinary computer life, that we unwittingly create, copy, move and deleted the files in our storages. It seems to be a random process that we do such file operations, however, we can obtain the highlight data which occur in the file operation process.

This paper is organized into five sections. Section II gives a brief review for the result of observation in a certain file sharing system. After motivated, Section III describes the construction of the mathematical model for file operations. The simulation of the proposed model is presented in section IV. Finally, Section V concludes the paper.

II. Observation

We found out the distribution of file types in the file sharing system of social groupware ”GRIDY” [12] that is used by over 10,000 registered companies in Japan. They share files on the cloud, that is a virtual storage on the Internet. File types can be classified by their file extensions, so that we have statistics of file types. Figure 1 shows the doubly logarithmic plot of the frequency of each file type by descending order. $p(k)$ is defined the number of files in the same extension at the k-th order. It is easy to see from regression that the distribution of file types seem to follow power-law distribution;

$$p(k) = Ck^{-\gamma} \tag{1}$$

with $C = 780,359$ and $\gamma = 2.438$ at the comparative high coefficient of determination in $R^2 = 0.9864$. This is our motivation of discussion why power-law distribution forms in file types.

In more detail, the data fit near the regression curve (1) although the data at small k are far from the law of power. There are 264 file types and the largest number of files at $k = 1$ is 63,392.

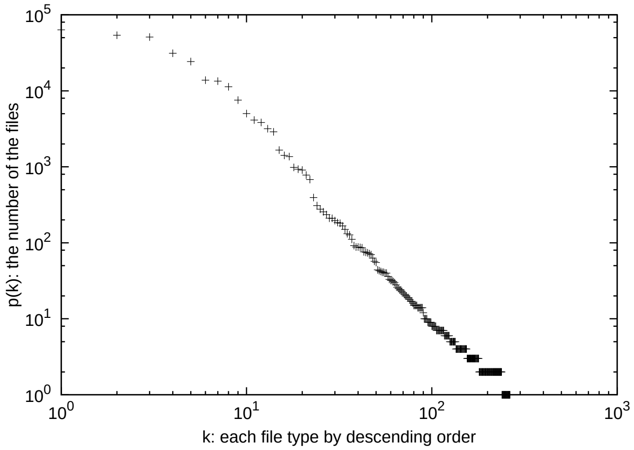


Figure 1. Observed distribution of file types in a file sharing system

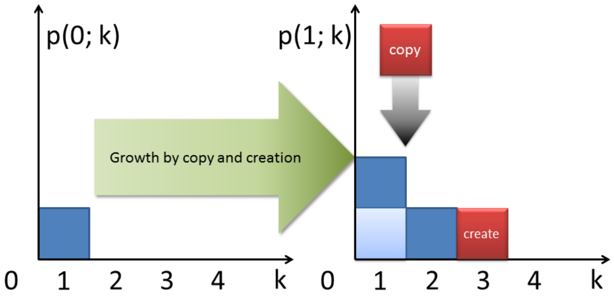


Figure 2. A model of file operation

III. Mathematical Model for File Operation

In this section, we construct a mathematical model for file operation, and show a fine result of power-law distribution of file types.

First of all, we define $p(t;k)$ as an integer-valued discrete function of t and k. The variable t means the time step which may take $\{0, 1, 2, \ldots\}$ and each k represents a sort of file types which may take $\{1, 2, \ldots\}$. There is one file in the system when $t = 0$. At the next step, we copy a file from the existing file to the system which type is $k = 1$ and we create a file at $k = 2$. When we copy a file from the existing files, we select a type of files at the probability proportional to the number $p(t;k)$ of the existing files in the system. Adding to this, we create a file at $p(t;k)$. See also Figure 2 in detail.

Table I
Estimation of Stirling's approximation

n	$n!$	Stirling	ratio
1	1	0.92	0.922
2	2	1.92	0.960
5	120	118	0.983
10	3628800	3.60×10^6	0.992
20	2.43×10^{18}	2.42×10^{18}	0.996
50	3.04×10^{64}	3.04×10^{64}	0.998
100	9.33×10^{157}	9.32×10^{157}	0.999

Mathematical analysis for the model indicates

$$p(t+1;k) = p(t;k) + \frac{p(t;k)}{2t+1} \cdot 1, \tag{2}$$

where we used

$$\sum_k p(t;k) = 2t + 1 \tag{3}$$

as the sum of all possible k. The boundary condition for file creation, we can write

$$p(t;t+1) = 1 \tag{4}$$

for all t and

$$p(t;k) = 0 \tag{5}$$

for every k with $k \geq t + 2$. Since the recurrence relation

$$\frac{p(t;k)}{p(t-1;k)} = \frac{2t}{2t-1} \tag{6}$$

and initial value

$$p(k-1;k) = 1, \tag{7}$$

we obtain

$$\begin{aligned} p(t;k) &= \frac{2t}{2t-1} \cdot \frac{2t-2}{2t-3} \cdot \cdots \cdot \frac{2k}{2k-1} \\ &= \prod_{j=k}^{t} \frac{2j}{2j-1} \\ &= \prod_{j=1}^{t} \frac{2j}{2j-1} \Bigg/ \prod_{j=1}^{k-1} \frac{2j}{2j-1} . \end{aligned} \tag{8}$$

In our discussion, we use Stirling's approximation

$$n! \sim \sqrt{2\pi} n^{n+\frac{1}{2}} e^{-n} \tag{9}$$

for sufficiently large n. See also Table I for accuracy of Stirling's approximation.

Applying Stirling's approximation to the previous expression, we have

$$\begin{aligned} \prod_{j=1}^{t} \frac{2j}{2j-1} &= \frac{2^t t!}{\frac{(2t)!}{2^t t!}} \\ &= \frac{2^{2t}(t!)^2}{(2t)!} \\ &\sim \frac{2^{2t}(2\pi)t^{2t+1}e^{-2t}}{\sqrt{2\pi}(2t)^{2t+\frac{1}{2}}e^{-2t}} \\ &= \sqrt{2\pi}t^{\frac{1}{2}}, \end{aligned} \tag{10}$$

so that it is concluded that the limit of the expression around t converges to $\sqrt{2\pi}$;

$$\lim_{t\to\infty} t^{-\frac{1}{2}} \prod_{j=1}^{t} \frac{2j}{2j-1} = \sqrt{2\pi}. \tag{11}$$

Similarly, we have

$$\begin{aligned}\prod_{j=1}^{k-1} \frac{2j}{2j-1} &= \frac{2^{k-1}(k-1)!}{\frac{(2k-2)!}{2^{k-1}(k-1)!}} \\ &= \frac{2^{2k-2}((k-1)!)^2}{(2k-2)!} \\ &\sim \sqrt{\pi}(k-1)^{\frac{1}{2}}\end{aligned} \tag{12}$$

for sufficient large k.

Summing up these calculations, we conclude

$$\lim_{t\to\infty} t^{-\frac{1}{2}} \prod_{j=1}^{t} \frac{2j}{2j-1} \Bigg/ \prod_{j=1}^{k-1} \frac{2j}{2j-1} = \sqrt{2}(k-1)^{-\frac{1}{2}}\ , \tag{13}$$

which indicates the power-law distribution.

IV. Computational Experiment

In order to investigate power-law distribution, we construct a network on the computer.

A. Simulation

There is a vertex at the beginning of the simulation. We construct the network by adding vertices according to the probabilities proportional to the number of edges that the candidate vertex have. In other words, a person who have many friends is tend to have new friends. In the simulation, we can see evolution of networks that have at most $8,048$ vertices.

Figure 3 summarized the flow chart of procedure in our simulation. First of all, every element of the array are initialized, and put the first vertex on the network. Next, the program loops it until the number of vertices is 8,084 that a new vertex comes to the network and select a vertex to be connected by the application of preference selection. The number $8,084$ can be extended to $8,084^2$. We restricted the maximum number of the array for the reason of analysis the network structure, however, the restriction is not necessary for the case that we see the number of edges in the network. At the last stage, the program make the histogram for the number of edges, then we can estimate the distribution by regression in statistical analysis.

The source code of the simulation is written in Java which is shown in the appendix at the last part of the paper, and the program ran on Intel Core 2 Duo CPU (T9600 @2.80GHz x2) with Microsoft Windows Vista 32bit version.

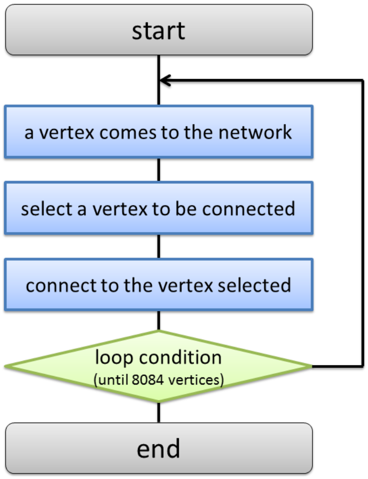

start
a vertex comes to the network
select a vertex to be connected
connect to the vertex selected
loop condition
(until 8084 vertices)
end

Figure 3. Flow chart of the simulation

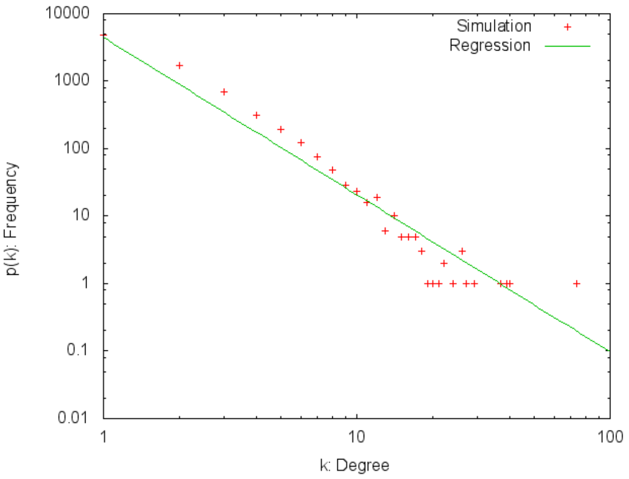


Figure 4. Distribution of degrees of vertices and regression curve

B. Results

Figure 4 shows the distribution of degrees of each vertex with regression curve. We use logarithmic scale both in axis. The result indicates

$$p(k) = 4.47 \times 10^3 k^{-2.33}, \tag{14}$$

which is characterized by scale-free structure of networks with $\gamma = 2.33$. The result of a trial is outputted as follows:

```
# === Simulation ===
# a = -2.3317937664992416
# b = 8.40563662841364
# Rˆ2 = 0.8875206382378562
1 4757
2 1702
3 701
```

V. Conclusion

According to the relevant results [16], we expect the power-law distribution of $\gamma = 3$ by the application of preference selection. In our result, we obtain $\gamma = 0.5$ from mathematical estimate, and $\gamma = 2.33$ from computer simulation. There is still a gap between mathematical analysis and simulation results. In addition to this result, we have the property of γ to converse to 3 if we give a large number of vertices to the network.

We have studied construction of scale-free networks in stochastic process. According to the model proposed by Barabási and Albert, we can construct scale-free networks using connecting probability that is proportional to the number of edges each vertex has.

In the problem of file types, we assume to copy files from old ones at random, so that we can conclude there is a similar effect to construct scale-free networks and power-law distribution emerges in file operations.

Acknowledgment

The authors would like to thank the referees for their useful comments on the previous manuscript of the paper.

References

[1] L. A. Adamic and B. A. Huberman, ”The Nature of Markets in the World Wide Web,” *Q. J. Electron. Commerce*, Vol. 1, pp. 5–12. (2000)

[2] R. Albert and A.-L. Barabási, ”Statistical mechanics of complex networks,” *Rev. Mod. Phys.*, Vol. 74, No. 1, pp. 47–97. (2002)

[3] A.-L. Barabási and R. Albert, ”Emergence of scaling in random networks,” *Science* 286, pp. 509–512. (1999)

[4] B. Blasius and R. Tönjes, ”Zipf’s Law in the Popularity Distribution of Chess Openings,” *Phys. Rev. Lett.* Vol. 103, pp. 218701. (2009).

[5] R. A. K. Cox, J. M. Felton, and K. H. Chung, ”The Concentration of Commercial Success in Popular Music: An Analysis of the Distribution of Gold Records,” *Journal of Cultural Economics*, Vol. 19, pp. 333–340. (1995)

[6] A. Clauset, C. R. Shalizi, and M. E. J. Newman, ”Power-law distributions in empirical data,” *SIAM Review* 51 (4), pp. 661–703. (2009)

[7] M. E. Crovella and A. Bestavros, ”Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes,” in *Proceedings of the 1996 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pp. 148–159. (1996)

[8] Y. Dan, ”Modeling and Simulation of Diffusion Phenomena on Social Networks,” *The Proceedings of 2011 Third International Conference on Computer Modeling and Simulation (ICCMS 2011)*, pp. 139–146. (2011)

[9] Y. Dan, ”Mathematical Analysis and Simulation of Information Diffusion on Networks,” *The Proceedings of The 11th IEEE/IPSJ International Symposium on Applications and the Internet (SAINT 2011)*, pp. 550–555. (2011)

[10] S. N. Dorogovtsev, *Lectures on Complex Networks*, Oxford University Press. (2010)

[11] Dorogovtsev, S. N., J. F. F. Mendes, and A. N. Samukhin, ”Structure of Growing Networks with Preferential Linking,” *Phys. Rev. Lett.* 85, pp. 4633–4636. (2000)

[12] GRIDY, http://gridy.jp/

[13] B. Gutenberg and R. F. Richter, ”Frequency of Earthquakes in California,” *Bull. Seismol. Soc. Am. 34 185*, Vol. 34, no. 4, pp. 185–188. (1944)

[14] L. Kullmann and J. Kertész, ”Preferencial Growth: Exact solution of the time dependent distributions,” *Phys. Rev. E 63*, 051112. (2001)

[15] A. J. Lotka, ”The Frequency Distribution of Scientific Productivity,” *J. Wash. Acad. Sci.* Vol. 16, pp. 317–323. (1926)

[16] M. E. J. Newman, ”Power laws, Pareto distributions and Zipf’s law,” *Contemporary Physics*, Vol. 46, no. 5, pp. 323–351. (2005)

[17] M. E. J. Newman, *Networks*, Oxford University Press. (2010)

[18] M. E. J. Newman, A. L. Barabási, and D. J. Watts, *The Structure and Dynamics of Networks*, Princeton University Press. (2006)

[19] V. Pareto, *Cours d’Économie Politique*, Droz, Geneva. (1896)

[20] D. J. de S. Price, ”Networks of Scientific Papers,” *Science*, 149 pp. 510–515. (1965)

[21] W. Reed and B. D. Hughes, ”From gene families and genera to incomes and internet file sizes: Why power laws are so common in nature,” *Physical Review E*, Vol. 66, Issue 6, id. 067103. (2002)

[22] E. M. Rogers, *Diffusion of Innovations, 5th ed.*, Free Press, New York. (2003)

[23] H. A. Simon, ”On a class of skew distribution function,” *Biometrika*, Vol. 42, pp. 425–440. (1955)

[24] A. Vázquez, R. Pastor-Satorras, and A. Vespignani, ”Large-scale topological and dynamical propertes of the Internet,” *Physical Review E*, Vol. 65, No. 066130. (2002)

[25] P.-F. Verhulst, ”Notice sur la loi que la population poursuit dans son accroissement,” *Correspondance mathématique et physique* 10, pp. 113–121. (1838)

[26] D. J. Watts and S. H. Strogatz, ”Collective dynamics of ’small-world’ networks,” *Nature* 393, pp. 440–442. (1998)

[27] G. K. Zipf, *Human Behaviour and the Principle of Least Effort*, Addison-Wesley, Cambridge. (1949)

Appendix

Here is the Java source code used for the numerical simulation in our research:

```
/*
  Simulation for Emergence in Complex Networks
  Copyright (C) 2011 Yuya Dan, Matsuyama University */

import java.util.Random;

public class NetworkEmergence{
  public static void main( String[] args ){
    final int NUM = 8048;  // Max{NUM} on memory = 8048
    Random r = new Random( 0 );

    // Start the Simulation
    System.out.println( "# === Emergence Simulation in Networks ===" );

    // Initialization
    int n = 1;
    int[] a = new int[NUM];

    for( int i = 0; i < a.length; i++ ){
      a[i] = 1;
    }

    // Construction of a Network
    for( int i = 0; i < NUM - 1; i++ ){
      int sum = 0;
      for( int j = 0; j < n; j++ ){
        sum += a[j];
      }
      int x = r.nextInt( sum );
      int s = 0;
      int j;
      for( j = 0; j < n && s <= x; j++ ){
        s += a[j];
      }
      a[--j]++;
      a[n]++;
      n++;
    }

    // Make the Histogram
    int max = 0;
    for( int i = 0; i < a.length; i++ ){
      if( max < a[i] ){
        max = a[i];
      }
    }
    int[] histogram = new int[max];
    for( int i = 0; i < histogram.length; i++ ){
      histogram[i] = 0;
    }
    for( int i = 0; i < a.length; i++ ){
      histogram[a[i] - 1]++;
    }

    // Statistical Analysis
    int nn = 0;
    double ax = 0.0, ay = 0.0;
    for( int i = 0; i < histogram.length; i++ ){
      if( histogram[i] > 0 ){
        ax += Math.log( (double)i );
        ay += Math.log( (double)histogram[i] );
        nn++;
      }
    }
    ax /= nn;
    ay /= nn;

    double Sxx = 0.0, Syy = 0.0, Sxy = 0.0;
    nn = 0;
    for( int i = 0; i < histogram.length; i++ ){
      if( histogram[i] > 0 ){
        Sxx += ( Math.log( (double)i ) - ax )
               * ( Math.log( (double)i ) - ax );
        Sxy += ( Math.log( (double)histogram[i] ) - ay )
               * ( Math.log( (double)i ) - ax );
        Syy += ( Math.log( (double)histogram[i] ) - ay )
               * ( Math.log( (double)histogram[i] ) - ay );
        nn++;
      }
    }
    Sxx /= nn;
    Sxy /= nn;
    Syy /= nn;
    System.out.println( "# a = " + ( Sxy / Sxx ) );
    System.out.println( "# b = " + ( ay - Sxy / Sxx * ax ) );
    System.out.println( "# Rˆ2 = " + ( Sxy / Sxx ) * ( Sxy  / Syy ) );

    // Result
    for( int i = 1; i < histogram.length; i++ ){
      System.out.println( i + "\t" + histogram[i] );
    }
  }
}
```

Simulation hypotheses

A proposed taxonomy for the hypotheses used in a simulation model

Pau Fonseca i Casas
Universitat Politècnica de Catalunya
Statistics and Operations Research department
Barcelona, Spain
pau@fib.upc.edu

***Abstract*— Defining a simulation model implies the use of different knowledge of the system we are going to model. Often, this knowledge is not complete or lacks in the needed detail in order to fully explain the behavior and the structure of the system. In that case, different hypotheses must be used in order to constrain the reality, or allow the needed complete and unambiguous definition of the model. However, all the hypotheses used do not lie in the same category. In this paper, we propose taxonomy for the hypotheses used in a simulation model in order to detect, previously to any implementation or model definition the possible lacks in the model construction.**

Keywords-simulation hypotheses; validation; verification; formal languages.

I. Defining a simulation model

Building a simulation model is an iterative process where usually different personnel are involved. This process starts always describing the system we want to represent, and what are the key elements that must be taken in consideration in order to define the model. Once we have the definition of what is "my system" we can go further to describe what is the problem we try to solve. As is stated by Professor George Box, "*all models are wrong, but some models are useful.*" It is interesting to keep this present in order to assure that we are performing a good validation process of the model as we can see next. Our model, although maybe is not correct, must be useful to our purposes, a good cite of this can be found on [1]: "*A simulation model should always be developed for a particular set of objectives. In fact, a model that is valid for one objective may not be for another.*"

Regarding to the process of Validation, Verification and Accreditation (VV&A) of a simulation model, the phases are based on the definition of "my system", the conceptual model, and the implementation of the model. Sargent [2] proposes the Figure 1 as the cycle that a simulation project follows until its completion. As we can see in Figure 1, data validation is assumed to be a central point in the whole simulation process. In the next sections we explore how the different model hypotheses work in each one of the different stages of the model validation that accomplishes the "Conceptual Model Validation", the "Operational Validation" and the "Data Validity".

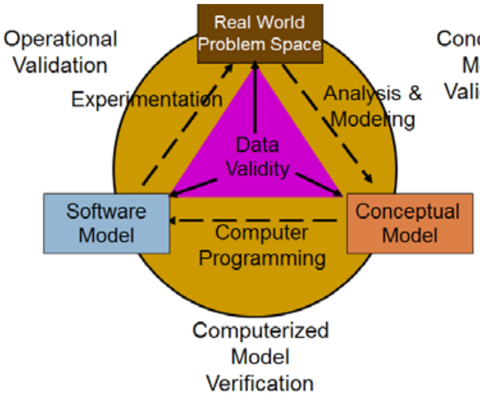


Figure 1. Validation and Verification process in a simulation model [2].

II. Working with the hypothesis

Often the system is a complex reality. Even though we can work with simple systems, we need to use hypotheses in order to constrain what is "my system". From this definition of "my system", the scope of my experimental framework, we need to go further defining a model that represents the hypotheses, the structure and the behavior of the elements that compose "my system".

This model must be defined using a formal representation, independent of the tool selected to perform any implementation. The need of define a formal representation of the system is widely exposed in several books and papers, but maybe three key aspects must convince us of the need of use a formal representation of the model.

First, the formal representation of a model, as is stated on [3], can be considered a product by itself. This is quite interesting since sometimes the representation of the knowledge that rules the different processes in a system (for example an industry) can be more interesting than the simulation by itself. The formal representation of a model helps to understand how the model behaves, and consequently how the system is constructed. Second, the formal representation of the model simplifies its implementation and enhances its maintainability. And third, a formal representation of a model simplifies the understanding of the model by all the different actors that are involved in the simulation project, improves the communication.

ISBN: 978-1-61208-169-4

The formal representation of any simulation model starts by the definition of a hypotheses document that represents how I understand the behavior and the structure of "my system". The formal representation of a simulation model is the formal representation of the simulation model hypotheses, hence the correspondence between the model and the hypotheses must be cleared understood, and this is often unusual. The main concern here is that in simulation literature, usually do not exist any classification between all the different hypotheses that can take part in the definition of the formal model. Not all the hypotheses have the same effect on the simulation model, and not all the hypotheses are needed on all the stages of the simulation model development. For that reason we propose to classify the hypotheses in three different categories as is explained next. This helps the model validation, because the modelers can focus its attention on the hypotheses related to each stage of validation. Also this taxonomy helps to understand the implications on the modifications on the knowledge we have of the system, or in the technology we use to implement the simulator.

III. Hypotheses taxonomy

First are those hypotheses that allow defining how the system behaves. Those hypotheses usually describe the system. In some papers related to VV&A are known as structural hypotheses. Mainly we want to add as many as we can of these hypotheses, since helps us in the description of the model. If we have a deeper knowledge of the system we can use a lot of these hypotheses describing its behavior. We know (or believe) that these hypotheses are true. We propose to name these hypotheses **Systemic**, since they describe the behavior of the system. These hypotheses can also be divided in two categories, those who are related with the data of the model, representing the flow of the elements of the system, and those related with the structure of the model, representing the underlying elements of the system.

Systemic Data hypotheses are those related with the data assumptions, which define the different probability distributions that rule the behavior of the model elements. **Systemic Structural hypotheses** can be those that represent the relations between the different elements that compose the model, the model behavior.

The other category is composed by the hypotheses that simplify the model we are going to build, named **Simplification** hypotheses. These hypotheses are useful in order to reduce the complexity of the model. Always, because the resources, the time we have to implement the model, and the knowledge we have of the system are limited, we must use this kind of hypotheses.

These three categories of hypotheses encompass all the hypotheses that can be used in a simulation project, with the main objective of simplify and help in the VV&A process and to help in the understanding of the nature the decisions taken during the modeling process.

IV. Validating a simulation model

Validated means that the hypotheses are assumed as true by all the parts involved in the simulation project. However, we cannot assure that a model obtained from a set of hypotheses is true; we can only assure that a model is false. Also modeling can be useful for many other reasons but predict [4], we can assume that a model is valid, and for that the applied hypotheses are valid, for a specific purpose. In order to conduct the Validation process in a simulation project, Naylor and Finger [5] proposed combining the three historical methods of rationalism, empiricism, and positive economics into a multistage process of validation. This validation method consists of:

1. Developing the model's assumptions on theory, observations, and general knowledge.
2. Validating the model's assumptions where possible by empirically testing them.
3. Comparing (testing) the input-output relationships of the model to the real system.

Figure 1 mainly shows this process, and as we can see on it we must validate three aspects:

1. Data validation.
2. Conceptual model validation: logical structure and hypothesis. Conceptual model validity is determining that (i) the theories and assumptions underlying the conceptual model are correct and (ii) the model's representation of the problem entity and the model's structure, logic, and mathematical and causal relationships are "reasonable" for the intended purpose of the model.
3. Operational validity: In this step, see if the outputs of the model have the accuracy required in accordance with the problem.

Also we can validate according [6].

1. Experimental validation: analyze if the experimental procedures used to obtaining the results are sufficient accurate.
2. Solution validation: in this validation the focus is on the accuracy of the results obtained from the model of the proposed solution. This validation is useful for the modellers in order to learn.

In this paper we are focused on the three first aspects and we avoid analyzing the experimental validation and the solution validation.

Since the validation process is the process of comparing the behavior of the model and the behavior of the real system in order to assure that we build the correct model, we focus in the formal representation of the model, although some techniques to test the model validity (as we can see next) uses a specific implementation of the model. This implies that we accept that the implementation of the model is correct (verification is done correctly). We avoid in this paper of talking about the verification of the tool that implements the simulation model.

Also, since the formal representation of the model lies on the hypotheses, the process of validate a model is the process of validate the different hypotheses used to build the formal representation of the simulation model.

In the next table, we define what kind of hypotheses must be validated for each one of the different aspects that must be validated.

TABLE I. HYPOTHESES VALIDATED IN EACH ASPECT.

	Systemic / Structural	Systemic / Data	Simplification
Validation of data		X	
Validation of the conceptual model	X		X
Operational validity	X	X	X

In order to validate the data, we must focus our efforts in the Systemic Data hypotheses. This implies that the nature of the tests that can be performed must focus on analyze the data we are going to use in the model. Here, we propose to distinguish between two aspects related to the data, the **structure** and the **nature**. The structure represents the shape that the data follows, while the nature means the source of this data, where the data lives. In that case, we can perform Chi-square tests in order to assure that the structure of the data is correct [7], [8]. To analyze the nature of the data, we must detect if this data will be always up to date, often assuring that an institution or research center take care of this work. This is usually a key aspect in order to perform environment simulation models. As an example a wildfire model needs information regarding of the vegetation, the digital terrain model (DTM), the winds, etc. All the validations tests proposed, related with the Systemic Data hypotheses, focuses on the data structure. The validity of the nature of the data is based on assure that the institutions (structures, enterprises, etc.) needed in order to obtain the data and keep this data up to date during the life of the simulator, exists. The validation of data structure is the validation of the Systemic Data hypotheses.

The conceptual model represents the structural relations of the different model elements and the behavior of the different element that compose the model. To validate the conceptual model we must focus on the formal representation of the model or in the implementation of this formal representation in order to validate mainly, the Structural Systemic and the Simplification hypotheses.

In brief, to validate the conceptual model we must validate Structural Systemic hypotheses and Simplification hypotheses. This implies that we must define techniques in order to assure that (i) the Structural Systemic hypotheses are correct for our purposes, and (ii) the Simplifications hypotheses do not transform the model in a caricaturing of the reality.

The Operational validity is focused in the results that we can obtain from an implementation of the model. Some of the methodologies presented here, like Black Box validation among many others, imply the use of all the model hypotheses, since the modeler try to validate the whole behavior of the computer program that implements the model. In that case we often cannot distinguish if the results (in case that the results are wrong) are due to an incorrect (Data or Structural) Systemic hypotheses, due to a wrong Simplifying hypotheses, or due to an incorrect implementation of the model introducing the verification phase into account.

In order to detect the sort of hypotheses that is validated depending on the validation methodology used, we propose a classification of some of well-known validation methods, and what are the hypotheses that they try to validate. In that case we are focused on the validation of a specific class of hypotheses, instead to try to validate the whole model.

V. VALIDATING THE HYPOTHESES

Different papers and books describe several tests related to the model validation [1] [2] [7] [8] [9]. Taking some of the tests described on [2] we are going a little further trying to define what are the hypotheses that are tested on each test. This will help us to define what are the batteries of tests we must use in order to build a good model, since as we can see next, not all the tests are focused on the same kind of hypotheses. The tests we analyze are (i) Validation face (ii) Black Box validation, (iii) Turing tests, (iv) Comparing with other models, (v) Degenerate tests, (vi) Extreme Condition tests, (vii) Event Validity and (viii) Variability of the Parameters and Sensitivity Analysis. For each one of these test we propose a classification depending on the hypotheses that we argue that mainly check.

In **Face** validation, the experts analyze the results obtained from the simulation model. From this analysis they can recognize the correctness of a model. One example could be to test if a simulation model of a specific machine behaves similar to the system machine. In this case, like in **Black Box** validation, the model is seen as a whole, implying that the validation is done over the complete set of hypotheses. Other similar case is **Turing** tests; in that case, the simulator generates fake documentation that is merged with real documentation. Again, the experts determine, examining the documentation that contains real and fake documents, what are the fake documents generated by the simulator. Finally, in this category, the comparison of the model outputs with **Historical Data**, allows to understand if the model is behaving as expected, at least for a scenario that is reproducing the behavior of an existing system. Looking these tests we can argue that the model is tested as a whole, for that the hypotheses tested are all, the Structural and the Simplification hypotheses. This family of validation tests is related with the Operation validity.

On the tests based on the **comparison with other models**, the underline idea is that if other models work fine, its outputs must be similar. As an example, if we have an analytical model, we can compare the outputs of this model with a new simulation model. On this kind of test we can chance the input data used and the model parameters in order to validate if both models follows the same patterns for the results. This allows determining if the structural relations of the models are correct. This test focuses on the structure of the model, we modify the data and we assume that the data is correct. For that in this kind of tests often only Systemic Structural hypotheses are tested, although the Simplification hypotheses can be tested too, since some of the decisions on the structure of the model rely on them.

On the **Degenerate Tests** is analyzed the model's behavior modifying the values of input and some selected internal values. The objective is to test if the modification of these parameters is coherent with the expected result. As an example if we increase the service time of a server we expect that the number of elements in the queue increases. Similar to this on **Extreme Condition Tests** is supposed that the model structure and outputs should be credible although using any extreme and unlikely combination of values for the variables. On **Fixed Values** tests we analyze the outputs for a well know values for the parameters of the model. In this test we look the outputs in order to compare them with the expected results. On these tests we are focused on understand if the relation between the elements are correctly described, for that these tests focuses on the Systemic Structural hypotheses. Note that in these tests we assume that the model is valid and we look the model to understand if the relations between the model elements are correct, no Simplification hypotheses are tested here.

Comparison with other models, Degenerate, Extreme Condition and Fixed Values tests are related with the validation of the conceptual model.

Variability of the parameters and **sensitivity analysis** allows analyzing the factors that have greatest impact on the performance measures. This allows determining what elements must be modeled carefully and detecting possible errors on the definition of the relations of the model elements. In this sort of tests we are focused on the Systemic Structural and Data hypotheses. We can detect if the probability distributions are correctly represented and if the relations between the different elements are correctly implemented. As an example, if we add between two model elements a causal relation when in the system only a correlative relation exists, we are introducing an error that can be detected with this test. This test is related with the validation of the conceptual model and the validation of the data.

On the **Event Validity** we compare the occurrences of some events with the real occurrence of those events in the system. As an example, the number of "broken" event occurrences in a specific machine of the model. This kind of test can be useful to test the Systemic Data hypotheses since usually the events that rule the behavior of a simulation model are defined using known probability distribution or an empirical distribution obtained from a database. This test is related with the data validation.

Other methods exists to validate the model, like Internal Validation, Predictive Validation, the use of Traces or the use of the Animation to understand if the model behaves as expected [2]. In Table II the description of the hypotheses tested on each one of the tests is shown. Subsequent to this table, if we want to validate our simulation model, at least is needed to test once all the hypotheses. This implies that we must select the tests that allow doing this, for instance selecting Compare with other models and Events tests, or Degenerative, Variability of the parameters and Black Box tests.

TABLE II. HYPOTHESES VALIDATED ON SOME TYPICAL TESTS

	Systemic / Structural	Systemic / Data	Simplify
Validation "Face"	+	+	+
Turing tests	+	+	+
Black box	+	+	+
Historical data	+	+	+
Compare with other models	+		+
Degenerative	+		
Extreme conditions	+		
Fixed values	+		
Variability of the parameters, sensitivity analysis	+	+	
Events		+	

Following the approach proposed by Naylor and Finger, and understanding that we need to validate all the hypotheses, we can start with the validation related to the Data, then continue with the conceptual model and finally perform an operational validity once a preliminary version of the model is constructed. We can use Table I to understand the hypotheses that must be validated on each validation, and Table II to select the appropriate test. We can start performing the goodness tests for the distributions we are going to use in our simulation model, and selecting some test that allows validating the Structural Data hypotheses, like the Events test. These tests are focused on the **structure** of the data. It is also needed to assure the validity of the **nature** of the data, or at least that is enough for our project purpose. As we said previously that means that an institution or enterprise assures that we have the data up to date in order to use it in our simulation model. Once we have the validation of the data we can perform the validation of the conceptual model. We can select some of the test that verifies this. Since we have the Systemic Data hypotheses validated, we can focus our efforts in the validation of the Systemic Structural hypotheses, using as an example the Degenerative test. Also, we can use the compare with other models test to test again the Systemic Data hypotheses (note that comparing with other models can be time demanding due to we need to have other models to perform this comparison).

Finally, we can perform the Operational validation. Note that in this stage, if we have all the hypotheses tested (we previously have been performed an Events and Compare with other models tests) we can argue that our model have all the hypotheses validated, hence the Operational validation is done. However, since the validation process never assure that we have a model correct (the validation can only assure that we have an invalid model) we can perform here some of the tests that works with all the hypotheses, like the Turing tests to improve our confidence in the model. Remark that since not all the tests are focused on the same typology of hypotheses we can argue that it is interesting to test first the model with tests that are focused on certain hypotheses in order to detect possible mistakes. It is more difficult to find an error in our model if we test all the hypotheses using

Black box test that if we are testing only the Structural hypotheses using the Extreme conditions test.

VI. USING NOT VALIDATED HYPOTHESES

Validate a simulation model is a time demanding task, and often we need to work with models that have some of the hypotheses not validated (as an example to analyze extreme conditions or to validate Systemic Structural hypotheses). Since not all the hypotheses have the same effect on the model, we can select what are the more interesting tests to be performed first in order to validate the more critical hypotheses first, always depending on the purpose on the model. Again, remark that we are looking for a useful model, often we can assume to work with non-validated hypotheses. In the table III we show if usually is desirable or not working with no validated hypotheses because the effects that this can imply to the model, again regarding to achieve a specific result. *Wanted* means the desired state of the hypotheses, *Useful* states means that, although it is not a desirable state, can be useful for the model construction, as an example to perform the validation of some model hypotheses, or to obtain some values from a hypothetical data. It is interesting to remark here that the use of Simplification hypotheses can be *Useful*, but never is desirable. The final objective of a simulation model is to work without simplifications. We must note that the simplification hypotheses are always false. That means that we know for sure that the reality is more complex that the structure that we are depicting on the model. Lastly *Unwanted* means that this state of the hypotheses is undesired for any purpose of the simulation model.

TABLE III. EFFECTS OF USING NO VALIDATED HYPOTHESES.

	Systemic / Structural	Systemic / Data	Simplification
Validated	Wanted	Wanted	Useful
Non validated	Unwanted	Useful	Unwanted

As we said previously, Systemic Structural hypotheses depict the relations between the different elements that compose the model. If these relations are not well defined, the model is not correct. For that, using no validated systemic structural hypotheses is an unwanted state, since we need to incorporate the knowledge of the client, and the client must assume that the relations depicted in the model are the relations that exist in the system, assuming the Simplification hypotheses as true.

On the case of Systemic Data hypotheses, using no validated data can be useful for testing purposes. As we said previously, Systemic Data hypotheses are those related with the data assumptions, which often define the different probability distributions that rule the behavior of the model elements. In some cases it is needed to use no validated data in order to analyze the behavior of the model in some specific circumstances, or as we see in Table II to test that the Systemic Structural hypotheses are correct. For this, using no validated Systemic Data hypotheses can be useful.

Finally, if we are using validated Simplification hypotheses on our model, we are assuming that they are useful in order to achieve our expected result with the project constrains (technology, time, resources, knowledge, etc.). Like in the case of the Systemic Structural hypotheses, if these simplification hypotheses are not validated, often implies that we are using some simplifications in or model that the client maybe cannot assume. This is dangerous for the project, and often reflects a bad communication with the client. As is stated in [1], the communication with the client from the beginning of the project, and the definition of a good hypotheses document is a key element for the success of a simulation project. Again, note that the desired state (all in wanted) implies to avoid the use of simplification hypotheses.

VII. WORKING WITH THIS TAXONOMY, WRITING THE HYPOTHESES DOCUMENT

As is stated on [1], the hypotheses document is a key element in the success of a real simulation project. Starting with some initial meetings, it is needed to start the redaction of this document that describes in detail the model assumptions and main objectives. This document is simple but clear, and we propose to use the template shown next. In this template we categorize, for each one of the different elements of the model the hypotheses used. Also, since we need to describe the Systemic Structural hypotheses we can use a formal language to describe the structure and the behavior of the model in a complete and unambiguous manner. A formal language like SDL [10] [11], DEVS [12] or Petri Nets [13] [14] [15] among others, becomes a powerful tool to represent the Systemic Structural hypotheses. In the diagrams of the model, we show the elements we are going to represent and the relation between all the elements. Remember that using a formal language to represent the model allows using some static methods to validate the correctness of the Structural Systemic hypotheses [2]. The proposed outline of the document has the next sections:

1. Description of the system.
2. Purpose of the model.
3. Simplification hypotheses for the external view of the model. Showing for each one if has been validated by the client or not.
4. Systemic Data hypotheses for the external view of the model, again showing if have been validated each one of them by the client.
5. Systemic Structural hypotheses for the external view of the model, using a formal language. This helps to the understanding of what are the key elements of the model that we are going to simulate.
6. For each one of the different elements of the model we detail its hypotheses. Again the Systemic Structural hypotheses can be represented (and we support this) using a formal language.

In our projects, we write in red the hypotheses that have not been validated. This simplifies the understanding by the client and by the modeler teams of the need to validate the hypotheses in order to achieve the desired result, and clearly shows what the state of the model construction is. In an iterative construction of a simulation model, once all the hypotheses of the document have been validated by the client and by the modelers, we have a simulation model that can be used to take decisions and can be prepared for its final step; believe in the model, the accreditation.

VIII. CONCLUSIONS AND FUTURE WORKS

The hypotheses are the key element that rules the definition of the model. However, not all the hypotheses used in a simulation model have the same effect on the model definition. Also the tests used to prove the validity of a simulation project not are focused on the same typology of hypotheses, for that is needed a taxonomy in order to focus our efforts in a selected subset of the tests that validates those hypotheses. In our taxonomy, three classes of hypotheses exist, Systemic Structural hypotheses, Systemic Data hypotheses and Simplification hypotheses. Regarding the data, we note that two aspects must be validated, the nature, that means that the data will be correct during the life of the simulation model, and the structure, that means the usual validation process for the data (for example, test if the inputs follow an exponential distribution). The Systemic Data hypotheses are focused on the structure of the data, since the nature can be assured if an institution take care of this data or we have the knowledge that the nature of the data do not change during the life of our simulation model, this is usual in an industrial simulation model, but unusual in an environmental model where the climatic data can change day to day and we need an institution that take care of this data.

We showed in this paper how this taxonomy can help in the validation process of a simulation model, thanks to improve the selection mechanism of the tests in order to achieve a complete (if needed) validation of the model.

Also we show the implications of work with no validated hypotheses. Sometimes it could be desirable to work with no validated Systemic Data hypotheses in order to validate the Systemic Structural hypotheses, or to obtain data related to extreme conditions situations. From this taxonomy, we can clearly understand that the Systemic hypotheses must grow in order to represent better and with more detail the relations and the data assumptions of the system, and the simplification hypotheses must decrease in order to represent the deeper understanding of the system.

The improvement on the perception of the system, or the improvement on the tools we can use to implement the model can modify the hypotheses. Often an improvement on the tools imply the use of less Simplification hypotheses, but an improvement on the system knowledge implies the use of more Systemic hypotheses, implying a detailed description of the model. This taxonomy helps to understand the implications on the modification in the system knowledge, or on the tools used to implement the model or in our needs, in order to define faster a new model and perform a new implementation.

The future work is focused in develop a methodology to systematize not only the validation but also the verification of the hypotheses, combining some existing methods to define the appropriate tools to implement a simulation model [16]. This can help us to understand the limitations of our simulation model, due to the hypotheses used, before any implementation.

BIBLIOGRAPHY

[1] Averill M. Law, "How to build valid and credible simulation models," in *Proceedings of the 2009 Winter Simulation Conference*, 2009, pp. 24-33. [Online]. http://www.informs-sim.org/wsc09papers/003.pdf (accessed 2011/09/21)

[2] Robert G. Sargent, "Verification and Validation of simulation models," in *Proceedings of the 2009 Winter Simulation Conference*, 2009, pp. 162-176. [Online]. http://www.informs-sim.org/wsc09papers/014.pdf (accessed 2011/09/21)

[3] Dirk Brade, "Enhancing modeling and simulation accreditation by structuring verification and validation results," in *Winter Simulation Conference*, 2000, pp. 840 - 848.

[4] Joshua M. Epstein, "Why Model?'," *Journal of Artificial Societies and Social Simulation*, vol. 11, no. 4, pp. , 2008.

[5] Thomas. H. Naylor and J. M. Finger., "Verification of computer simulation models," *Management Science*, vol. 14, no. 2, 1967. [Online]. http://www.jstor.org/pss/2628207 (accessed 2011/09/21)

[6] Stewart Robinson, "Simulation Verification,Validation and Confidence: A Tutorial," *TRANSACTIONS of The Society for Computer Simulation International*, vol. 16, no. 2, pp. 63-69, 1999.

[7] Antoni Guasch, Miquel Àngel Piera, Josep Casanovas, and Jaume Figueras, *Modelado y simulación*. Barcelona, Catalunya/Spain: Edicions UPC, 2002.

[8] Averill M. Law and W. David Kelton, *Simulation Modeling and Analysis*.: McGraw-Hill, 2000.

[9] Robert G. Sargent, "Verification, validation and accreditation of simulation models," in *Proceedings of the 2000 Winter Simulation Conference*, 2000, pp. 50 - 59.

[10] Lauren Doldi, *Validation of Communications Systems with SDL: The Art of SDL Simulation and Reachability Analysis*.: John Wiley & Sons, Inc., 2003.

[11] Telecommunication standardization sector of ITU. (2002) Series Z: Languages and general software aspects for telecommunication systems. [Online]. http://www.itu.int/ITU-T/studygroups/com17/languages/Z100.pdf (accessed 2011/09/11)

[12] Bernard P. Zeigler, Herbert Praehofer, and Tag Gon Kim, *Theory of Modeling and Simulation*.: Academic Press, 2000.

[13] Carl A. Petri, *Kommunikation mit Automaten*. Bonn: University of Bonn, 1962.

[14] Manuel Silva Suárez, *Las Redes de Petri: en la Automática y la Informática*. Madrid: Editorial AC, D.L., 1985.

[15] James Lyle Peterson, *Petri Net Theory and the Modeling of Systems*.: Prentice-Hall, 1981.

[16] Gladys Rincon, Marinelly Alvarez, Maria Perez, and Sara Hernandez, "A discrete-event simulation and continuous software evaluation on a systemic quality model: An oil industry case," *Information & Management*, vol. 42, pp. 1051-1066, 2005.

SimARC: An Ontology-driven Behavioural Model of Alcohol Abuse

Francois Lamy
CRICS
Charles Sturt University
Bathurst, Australia
flamy1978@gmail.com

Pascal Perez
SMART Infrastructure
UOW
Wollongong, Australia
pascal.perez@uow.edu.au

Alison Ritter
NDARC
UNSW
Sydney, Australia
alison.ritter@unsw.edu.au

Michael Livingston
TPADC
Uni. of Melbourne
Melbourne, Australia
michaell@turningpoint.org.au

***Abstract*—Alcohol-related problems (assaults, accidents and/or crimes) and alcohol abuse are recurrent societal problems leading to high social costs. Finding adapted policies to tackle this issue isn't a trivial task due to the highly complex nature of alcohol consumption as many interrelated risk factors interact in a hardly predictable way. This paper describes an agent-based simulation model, called SimARC (Simulation of Alcohol-Related Consequences), aiming at exploring the complex interplay of these factors following a generative process whereby theory and model co-evolve within iterative loops. To explore the complexity of alcohol use and abuse, we need not only to include the aforementioned risk factors but also their evolution and highly dynamical interactions across scales. Therefore, our agent-based model aims to encapsulate several levels of reality. Considering an ontology as catalog of elements and relation amongst those elements, our ontology-driven behavioral model includes: neuro-biological responses to alcohol use (individual level), peer influence channeled through various social networks (meso-level) and societal responses to alcohol-related problems (meta-level). This ontological framework aims to establish a robust test-bed to analyze – in silico – the plausible consequences of various public policies related to alcohol abuse in public venues. After a brief review of the literature, we present SimARC's core structure and preliminary results.**

Keywords-agent-based model; ontology; alcohol; social simulation; public health.

I. INTRODUCTION

In its «Global Status Report on Alcohol and Health 2011», World Health Organization (WHO) points that alcohol «is a causal factor in more than 60 major types of diseases and injuries and results in approximately 2.5 million deaths each year […] Thus, 4% of all deaths worldwide are attributable to alcohol» [1]. Furthermore, a recent report from the Independent Scientific Committee on Drugs (ISCD) indicates via that alcohol, in term of social cost, is more dangerous than heroin and crack [2]. In the same vein, Collins and Lapsley have estimated at 15.3 billions AUS$ (11.6 billions €) the social cost of alcohol [3].

Moreover, both in Europe and Australia, «binge drinking» (heavy drinking session leading to intoxication) is on the rise inducing greater chances of individual harms (i.e., falls, pedestrian/car accidents) as well as greater risks of violence (i.e., brawl, degradation, violent assaults) [4][5].

Due to its legal status and large availability alcohol has become a major health problem for governments [6][7] who generally attempt to solve this problem by different combinations of public policies, such as alcohol taxation, prevention campaigns or reduction in availability [8][9]. Beside, net revenues associated with alcohol consumption largely make up for subsequent expenditures in the Australian federal budget [3].

Hence, alcohol-related social harm remains a difficult research topic [10] as consumption patterns adapt quickly to new policies. For example, individual change their drinking habits (i.e., "preloading" episode, shifting from one type of alcohol to another one) or license premises adapting their marketing to remain competitive. As well as all the others drug uses, alcohol consumption and its aftermaths are complex social phenomena: they result from the interaction of many risk and protective factors that dynamically evolve through time [11].

These factors belong to distinct levels of analysis: genetic predispositions; neurophysiology and neuro-pharmacology of alcohol; individual psychology; social and environmental conditions; current laws; economical constrains or cultural norms [12]. We consider here three levels of analysis: a micro-level (the individual, his neurologic, physiologic and social characteristics), a meso-level (groups, peer influence and significant others) and a macro-level (public policies, urban geography and societal responses).

Our work aims to create a social simulation, which integrates three levels of analysis in order to get a better understanding of alcohol use and misuse. Once calibrated and validated, this type of simulation model could be used to inform policy-making debate on alcohol [13]. To describe this simulation, we will review the different components of the model, then we will discuss the need for new technologies to capture alcohol-related problems, and finally, we will describe the different components of SimARC and show some preliminary results.

II. Alcohol Use: a Multi-Factorial System

Our three different levels represent three levels of interactions between five components: alcohol, individual, group, context and society. We consider the relation between the alcohol and individual components constitutes the *prima causa* of alcohol-related harms.

A. *Micro-level: Alcohol/Individual*

Alcohol is a powerful psychoactive substance highly addictogen. BAC (Blood Alcohol Concentration) is the main indicator of alcohol intoxication and impairment. BAC gives good indications concerning cognitive and motor impairment: the following figure (cf. Figure 1) illustrates the relation between accident and BAC [14].

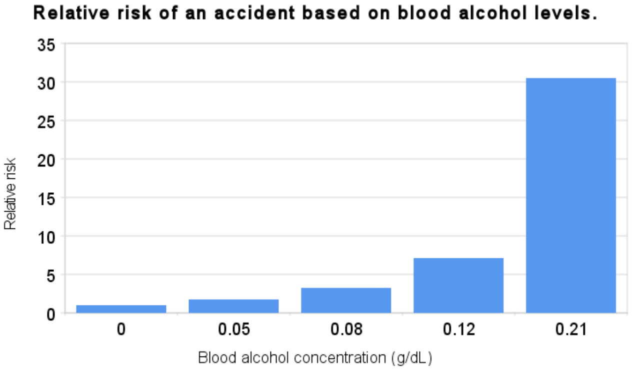


Figure 1. Risk of traffic accident/BAC

However, aggressive behaviors are consequences of the neural action of alcohol on the brain. Ethanol, the active principle of alcohol, has an impact on many neurotransmitters, but Dopamine, GABA, Glutamate and Serotonin (5-HT) are strongly related to behavioral changes [15]. As most of the other drugs, alcohol generates release of Dopamine, the neurotransmitter of reward and pleasure. From behavioral viewpoint, dopamine increases self-confidence, and happiness; however, it is also considered as one of the key-factors that trigger craving, addiction and schizophrenia [16].

Alcohol also acts on GABA (Gamma-Amino-Butyric Acid) the principal inhibitory neurotransmitter in the brain. In standard volume, GABA has a relaxant effect, but higher doses cause drowsiness and motor impairments [17]. Another role of GABA is to balance the excitatory action of Glutamate. At normal dose, this neurotransmitter is implied in learning and memorization [18], inversely, higher concentration of Glutamate in the brain leads to "excitotoxicity", impairing or killing neurons.

Finally, ethanol interacts strongly with Serotonin. Also known as 5-hydroxytryptamine (5-HT), this neurotransmitter is responsible for mood regulation, sleeping cycles and thermoregulation. Mild dose brings euphoria and a sentiment of happiness. Conversely, depleted serotonin level is generally correlated with feeling of depression and aggressive behaviors [19].

From a neurophysiologic viewpoint, during alcohol consumption the level of Glutamate in the brain decreases with a correlative increase in GABA concentrations, giving a mild relaxant feeling and desinhibition. At the same time, the agonist effect of alcohol on Dopamine and Serotonin neurotransmitters induces euphoria as well as feelings of happiness and self-confidence in the user [20]. Inversely, excessive amounts of GABA and low rates of Glutamate lead to motor impairment, lack of concentration and potentially induce sleep, increasing dramatically the chance of being involve in a car or pedestrian accident.

Once alcohol slowly disappears from the bloodstream and "stocks" of those neurotransmitters have been partially or completely depleted, a "calm down" period starts and users sense the different side effects following their consumption. Individuals will start to feel depress, get moody or exhibit violent behaviors (due to dopamine and serotonin depletions). Furthermore, some neurons may be damaged due to the rise of glutamate (excitotoxicity) after drinking [21].

This short review of the neuro-pharmacology of alcohol gives a partial explanation of behavioral changes but remains insufficient if it is not linked to social reactions and especially peer influence.

B. *Meso-level: Individual/Group*

Sociology has studied the impact of peer perception and influence regarding alcohol and drug use. Making reference to social learning theory, Kilpatrick et al. [22] and Flay et al. [23] have shown that children witnessing drug consumption from "significant others" (parents, sibling or tutors) have an increased risk of substance abuse. Obviously this influence can be extended to other elements of personal interactions, “peer pressure” has an important influence on experimental alcohol and drug use. On this subject, a vast literature exists about the social influence of friendship groups [24] indicating that individuals are influenced (positively or negatively) by their friends but also select which peers they have to mix with in order to find and use any drugs [25].

If peer-pressure has been the object of many studies as a risk factor, "social control" coming from members of the family, friends or community consist a solid protective factor [26]. Drug users compare their behaviors and consumption to other consumer comportments: irrational or erratic behaviors are generally banned and stigmatized [27]. However, repeated public misbehaviors around a particular location and/or generalization of alcohol-fuelled violence and disorders call to societal and political responses.

C. *Macro-level: Social Environment/all components*

As pointed by Livingston, the density of alcohol-related venues is directly related to violence in all neighborhood type, but bars and nightclubs are associated with violence in the inner city while packaged alcohol outlets were associated with violence in suburban zone [28][29]. Similarly, the social capital of neighborhood seems to decrease with the density of alcohol outlet leading to more incivilities and to a possible social segregation [30].

Inside those venues, measures as closing times [31], limitation of crowding and a coordinated staff [32] have significant positive impact on alcohol-related violence.

Alcohol price taxation is accurately associated to alcohol-consumption: in their review of the different studies done on the subject, Chaloupka et al. indicated that increasing the monetary prices of alcoholic beverages reduces significantly alcohol consumption and alcohol-related problems [33].

Having reviewed those different factors, we need to find a robust framework able to encapsulate these components and capture their inter-evolutions over time. Hence, we propose to employ computer simulation to mimic this social phenomenon.

III. Agent-Based Model and Social Simulation concerning alcohol

Computer simulation models have attracted an increasing number of researchers and practitioners over the last decade. As a matter of fact, social simulations can be used as artificial social experiments (*in-silico*) to explore the consequences of pre-defined conditions on a range of specific social and environmental indicators. In his seminal book '*Generative Social Science*', Epstein argues that computer simulations provide new tools for integrative and empirical research in social sciences [34].

A particular instance of computer simulation, called Agent-Based Modelling (ABM), allows building artificial societies from the bottom-up; whereby individual autonomous agents interact, communicate and pursue personal goals while societal norms and regulations constrain their freedom [35]. ABM is also very helpful for collecting and making sense of dynamical (spatial movements, time series) or heterogeneous information (qualitative, quantitative, ill-defined or aggregated).

Finally, ABM is largely used in environmental, health or defence studies to explore intervention scenarios with policy makers [36]. According to Liu and Eck, "*crime simulation is [also] an emerging research area that has the potential of revealing hidden processes behind urban crime patterns and criminal justice system operations*" [37]. Again, the analytical value of the approach doesn't rely on its capacity to describe spatiotemporal dynamics, but – more importantly – on its ability to assess different hypothesis about social causality [38].

In the field of alcohol and other drugs use, ABM has been successfully used to explore mechanisms of drug use initiation [39], and impacts of different policing interventions on street-based illicit drug markets [40]. Agent-based simulations concerning alcohol experiences gossiping amongst student [41], interactions agent-environment [42] or movement of alcohol user in the city [43] have mainly studied agent/group or agent/environment interactions.

Our aim is to encapsulate both neurologic physiology, impact of the network on decision, geographic data and societal response in a single model. Computer science concept of ontology seems to tally with our objectives. Originally, ontology was a philosophical concept which, a branch of metaphysics: coming from *ontos* (being) and *logos* (discourse), ontology aims to describe general properties of things. For our purpose, we will consider ontology as *a description of a particular domain defined by its objects, concepts, and their properties and relations* [44]. This framework enables the description of the previous data and concepts in a common language, Unified Modeling Language (UML) (cf. Figure 2):

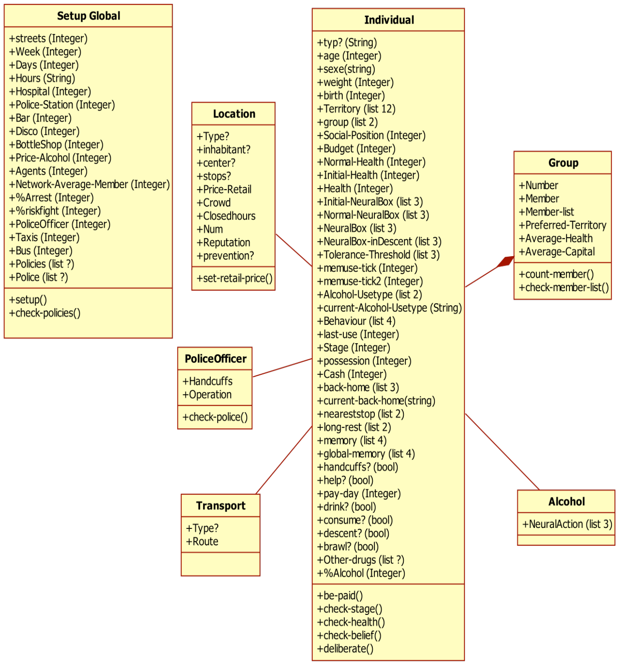


Figure 2. SimARC Class Diagram

IV. SimARC: Global Functionning

As indicated by Ferber [45], a Multi-Agent System (MAS) comprises of the following elements:

- An environment (E), a space that generally has a volume;
- A set of passive objects (O) which can be perceived, created, destroyed and modified by the agents;
- An assembly of agents (A) representing the active set of objects;
- An assembly of relations (R) that link active or passive agents to each other;
- An assembly of operations (Op) making it possible for the agents of A to act on objects from O.

SimARC aims to encompass all those different components. Besides that some part of this simulation are still under construction, we coded the UML structure in Netlogo 4.1.3 [46]. SimARC interface allows the experimenter to choose the number of Streets, Bar, Disco, Bottle-shop, Hospital and Police station. Sliders help to choose how many Agents and Constables will be created.

Finally, the simulation user can select Alcohol Price, Police Operations and Public Policies (those two latter are pre-implemented by the programmer). He can also select via sliders the percentage for a constable to arrest an alcohol user and the "brawl risk" percentage.

In the two next sections, we will describe the manner in which we have implemented most of our algorithms and the last section will give some preliminary results.

A. *SimARC Urban Environment and Interface*

The visual interface is a drastic simplification of an urban area, the grid includes the following features: street (here in black), house (green), bar (blue), disco (purple), bottle-shop (orange), police station (red), hospital and a rehab centre. The Figure 3 gives an outline of the urban environment.

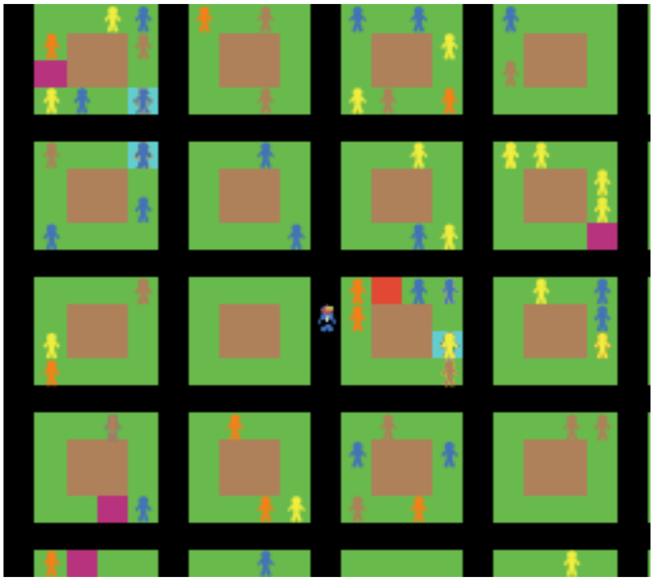


Figure 3. SimARC Urban Grid

Licensed premises (*venues*) have different Retail Prices and characteristics (Happy-Hours, Lock-out, Curfew, Crowding) as well as ‘Reputation’. Every step (tick in Netlogo) represents 2 hours time, 12 ticks a real weekday and weekend (Monday, Saturday etc.).

The Retail Price varies according to the type of venues: bottle-shops have their retail price equal to the price chooses by the experimenter; bar sees this price increases by 2 and discos have a retail price multiply by 2. According to the Reputation of the venue, constables may be more incline to patrol in that neighborhood and some agents can just avoid this venue. Actually and according to the implemented Public Policies, Retail Prices can increase for every alcohol-venues, but Happy-Hours can also be suppressed and/or Curfews or Lock-outs can be imposed. These different policies influence the choice and drinking patterns of the virtual alcohol users.

B. *SimARC Agents and Networks*

Each agent is characterized by the following attributes:

- Physical attributes (*Health*, *Age*, *Gender*, *BAC*);
- Status of neurotransmitters (*Serotonin*, *GABA*, *Glutamate* and *Dopamine*);
- A *Stage* representing its frequency of alcohol use and a correlated *Alcohol-routine*;
- Behavioral tendencies (aggressive, neutral or elusive)
- Memories of past experiences (past consumption, accidents, violence and sickness);
- Strategy to "*get-back-home*" once the night-out is over (private or public transport);
- Social characteristics (*Income, Friends, Address, Favorite Venues*).

An agent acts according to a series of heuristics based on an hourly schedule. All agents have a routine "daily-life": they go to work (part of their *Address* data), earn virtual money every fortnight (*Income*), eventually, decide to have a drink and finally, come back home to rest (restore their *Health* and *Status of neurotransmitters*). Agents earn ten times their Income (normal-distribution) every fortnight (randomly predefined). This fortnight income constitutes "pocket money" for non-essential expenses. The average amount of this pocket money is equal to 180.

Alcohol consumption varies according to each agent’s *Alcohol-routine* and *Stage*: some agents may have a few drinks in their favorite bar during the weekend while others can have several binge-drinking sessions at home during weekdays. Some agents are just staying home and sober the whole week, resulting in no individual harm or social trouble. Other agents consuming large quantities of alcohol can display violent or dangerous comportments (brawl, accident and having been sick are counted and memorized) and these heavy drinking decrease their *Health* attribute.

Each "Drink" represents a Standard Drink (10 g of alcohol) and each intake increases the BAC of male agent by 10/(Weight x 0.7) and by 10/(Weight x 0.6) for female agent. BAC is reduced by 0.15 every tick (2 hours time).

Each consumption modifies the levels of neuro-transmitters and the relative balance of neurotransmitters governs changes in behavioral patterns. An agent might change his opinion about alcohol consumption based on its cumulative experience of negative consequences (personal or witnessed) during or after successive night-outs. In turn, these updated opinions might change an agent’s *Alcohol-routine* and *Stage*.

Furthermore, agents can interact through physical co-location in the spatial environment or through messages amongst friendship networks. Therefore, agents of a same network move all together in *Favorite Venues*. Those *Friends* of the network can also "ask" an agent with a low *Health* or frequent dangerous behaviors to "slow down": if it accepts, the agent will not drink for some weeks, recovering from its past consumptions, otherwise, it will change its primary network to find new drinking mates.

C. *Preliminary Results*

In this section, we present some preliminary results from SimARC. However, this simulation hasn't been entirely calibrated yet. Therefore, these experimentations intend to test the internal consistency of the model. To do so, we examine the consequences of alcohol taxation policy. Four different prices have been tested 1, 5, 10 and 15 (relatively to the average 180 "pocket money").

For this experiment, the virtual population is composed of 750 agents and 2 constables. Those latter have 1% chance to arrest users with a BAC > 0.5. In order to initiate the simulation, we consider that 70% of the agents start with a Stage = 1, 15% with a Stage 2, 10% with a Stage 3 and 5% with Stage 4 (those data have to be calibrated).

For each scenario, we have run 50 replicates of 4367 time steps (one year simulation time). We have measured quantities of alcohol consumed for each scenario as well as the number of accidents and fatal accidents.

Figure 4 summarizes our results on Standard Drink consumption depending of the price of alcohol:

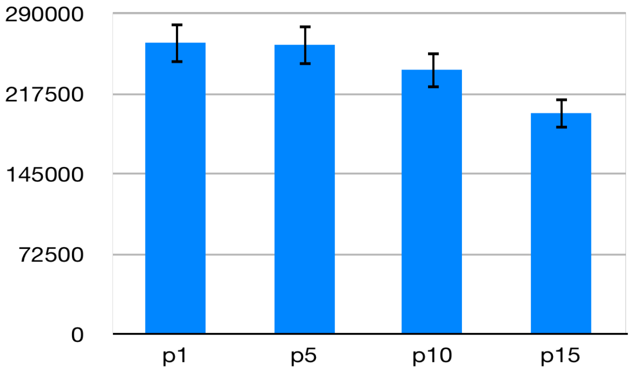


Figure 4. Alcohol Consumption (SD)/Price of Alcohol

Alcohol consumption decreases of 8.6% between P5 and P10, and decreases of 23.6% between P5 and P15. Those results seem to be concordant with economic studies [33]. However, it seems surprising that the amount of alcohol consumed for P1 and P5 are quasi-equivalent: we attribute this proximity of SD consumed to the "social control" operates by peers and to evolution of individual opinions in response to bad experiences during heavy drinking sessions (see IV.B).

Concerning Accidents with have implemented an algorithm matching the relative risk to be involved in an accident shown in Figure 1. Risk increases with BAC according to the following equation:

$$p(crash|BAC) = 1 / (1 + 0.2 \exp (5 - 2 * \%BAC)).$$

According to *MUARC* (Monash University Accident Research Centre) on alcohol-related car crashes provoke fatal accidents in approximately 1% of cases, and cause serious wounds in 34,5% and 64,5% constitute minor trauma [47]. Our experiments display the following results (Figure 5):

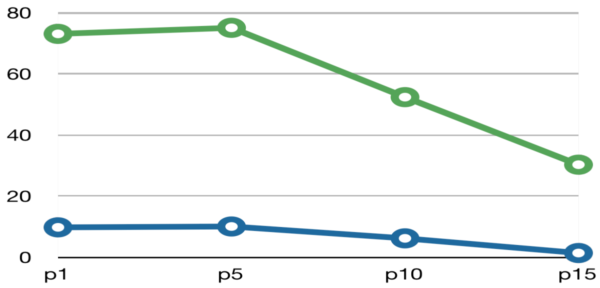


Figure 5. Accidents (green) and Fatal Accidents (blue)/Price of Alcohol

As expected, the number of accidents and related lethal injuries decreases with the increase of alcohol price.

V. Conclusion and Further Works

From reviewing different factors involved in alcohol use, we have underscored the necessity of a multidisciplinary perspective to understand the complexity of this phenomenon. This complexity leads us to consider an ontologic ABM as a suitable method to mimic alcohol consumption and alcohol-related social problems. This ontologic model is implemented and simulated with Netlogo in order to run multiple simulations and so achieve public policies testing.

At this stage, most of the algorithms are based on empirical heuristics calibrated against existing quantitative and qualitative data. While the simulation of behavioral patterns linked to alcohol consumption and driven by the neurobiological status of an agent is well advanced, the research team is still seeking complementary information to represent the consequences of these behavioral patterns. Both quantitative and qualitative complementary data are needed.

We plan to realize in-depth interviews with different categories of alcohol users in order to obtain a better understanding of alcohol users behaviors (how their habits change, what are the different reasons for such changes, how users evolve through life....). As proposed by Moore and colleagues, SimARC aims to integrate ethnographical and epidemiological information in an iterative way [48]. Later on we intend to integrate real urban information (GIS) in order to display an accurate geographical context and to give a more accurate representation of public policies implications and results.

REFERENCES

[1] D.J. Nutt, L.A. King & L.D. Phillips, Drug harms in the UK: a multicriteria decision analysis. The Lancet. vol. 376 (9752). 2010.

[2] WHO, «Global Status Report on Alcohol and Health». 2010.

[3] D.J Collins & H.M Lapsley, The costs of tobacco, alcohol and illicit drug abuse to Australian society in 2004/2005. 2008.

[4] Drugs and Crime Prevention Committee, Report on Inquiry into Strategies to Reduce Harmful Alcohol Consumption. Melbourne: Parliament of Victoria. 2006.

[5] OFDT, Tendances (76), Les niveaux d'usage de drogues en France en 2010. 2011.

[6] H. Parker, F. Measham & J. Aldridge, Illegal Leisure: the Normalisation of Adolescent Recreational Drug Use. Routledge. 1998

[7] J. Grace, D. Moore & J. Northcote, Alcohol, Risk and Harm Reduction: Drinking Amongst Young Adults in Recreational Settings in Perth, NDRI. 2009.

[8] WHO, European Alcohol Action Plan 2012-2020. 2011

[9] Australia Commonwealth, National Alcohol Strategy 2006-2009. Ministerial Council on Drug Strategy. 2006.

[10] R. Nicholas. Understanding and responding to alcohol-related social harms in Australia. Options for Policing. NDLERF. 2008.

[11] D.M. Gorman et al. Implications of Systems of Dynamic Models and Control Theory for Environmental Approaches to the Prevention of Alcohol-and other Drug-use related Problems. Substance Use & Misuse. vol. 39 (10-12). 2004.

[12] J. Unger et al. What are the implications of structural/cultural theory for drug abuse prevention? Sub. Use & Misuse. vol. 39 (10-12). 2004.

[13] P. Gruenewald. Why do alcohol outlets matter anyway? A look into the future. Addiction. vol. 103. pp. 1585-1587. 2008.

[14] www.infrastructure.gov.au

[15] R.M. Julien, C.D. Advokat & J.E. Comaty. A Primer in Drug Action: a comprehensive guide to the actions, uses, and side effects of psychoactive drugs. Worth Publishers. 2008.

[16] H.J. Hanchar, P.D. Dodson, R.W. Olsen, T.S Otis & M.Wallner. Alcohol-induced motor impairment caused by increased extrasynaptic GABA A receptor activity. Nature Neurosciences, vol. 8 (3). 2005.

[17] M.K. Ticku & A.K. Mehta. Effects of alcohol on GABA-mediated Neurotransmission. Handbook of Experimental Pharmacology. vol. 114 (6). pp. 103-119. 1995.

[18] W. McEntee & T. Crook.Glutamate: its role in learning, memory, and the aging brain. Psychopharmacology vol. 111 (4). 1993.

[19] D.M. Lovinger. The Role of Serotonin in Alcohol's Effects on the Brain. Current Separations. vol. 18 (1). 1999.

[20] K. Yoshimoto et al. Alcohol stimulates the Realease of Dopamine and Serotonin in the Nucleus Accumbens. Alcohol. vol. 9 (1). 1992.

[21] P.L. Hoffman. Glutamate receptors in Alcohol Withdrawal-Induced Neurotoxicity. Metabolic Brain Disease. vol. 10 (1). pp. 73-79. 1995.

[22] D.G. Kilpatrick, et al. Risk Factors for Adolescent Substance Abuse and Dependence Data from a National Sample. Journal of Consulting and Clinical Psychology, vol. 68 (1). 2000.

[23] B.R. Flay et al. Differential Influence of Parental Smoking and Friends' Smoking on Adolescent Initiation and Escalation and Smoking, Journal of Health and Social Behavior, vol. 35(3). 1994.

[24] M. Pearson & L. Michell. Smoke Rings: social network analysis of friendship groups, smoking and drug-taking, Drugs: education, prevention and policy, vol. 7 (1). 2000.

[25] K.E. Bauman & S.T. Ennet. On the importance of peer influence for adolescent drug use: commonly neglected considerations, Addiction, vol. 91 (2). 1996.

[26] S. Sussman et al. Adolescent peer group identification and characteristics: A review of the literature. Addictive Behaviors. vol. 32. pp. 1602-1627. 2007.

[27] T. Decorte. Drug users' perceptions of 'controlled' and 'uncontrolled' use. International Journal of Drug Policy, vol.12, pp.297-320. 2001.

[28] L. Zhu, D.M. Gorman & S. Horel. Alcohol Outlet Density and Violence: A Geospatial Analysis. Alcohol & Alcoholism. vol. 39 (4). pp. 396-375. 2004.

[29] M. Livingston. A Longitudinal Analysis of alcohol Outlets Density and Assault. Alcoholism: Clinical and Experimental Research. vol. 32 (6). 2008.

[30] K.P. Theall et al. Social Capital and the Neighborhood Alcohol Environment. Health & Place. vol. 15. pp. 323-332. 2009.

[31] K. Kypri et al. Effects of Restricting Pub closing Times on Night-Time Assault in an Australian City. Addiction. vol. 106 (2). pp. 303-310. 2011.

[32] K. Graham et al. Bad Nights or Bad Bars? Multi-level analysis of Environmental Predictors of Aggression on Late-Night Large-Capacity Bars and Clubs. Addiction. vol. 101. pp. 1569-1580. 2006.

[33] F.J. Chaloupka et al. The Effects of Price on Alcohol Consumption and Alcohol-related Problems. National Instituve on Alcohol Abuse and Alcoholism. 2002.

[34] J. Epstein. Generative Social Science: Studies in Agent-Based Computation. Princeton University Press. 2007.

[35] R.K.Sawyer. Social Emergence: Societies As Complex Systems. Cambridge University Press. 2005.

[36] P. Perez & D. Batten. Complex Science for a Complex World: Exploring Human Ecosystems with Agents. ANU Press. 2006.

[37] L. Liu & J. Eck. Artificial Crime Analysis Systems: Using Computer Simulations and Geographic Information Systems. Information Science Reference. 2008.

[38] P.j Bratimgham et al. A Statistical Model of Criminal Behavior. Math. Models and Methods in Applied Sceiences vol. 18. 2008.

[39] N.H. Agar et al. Epidemiology or Marketing? The Paradigm Busting Use of Complexity and Ethnography. Proceedings of Agent. 2004.

[40] P. Perez et al.. SimDrug: Exploring the Complexity of Heroin Use in Melbourne. DPMP. Monograph 11. 2005.

[41] L.A. Garrison & D.S. Babcock. Alcohol Consumption among College Students: An Agent-baded Computational Simulation. Complexity. vol. 14 (6). 2009.

[42] D.M. Gorman et al. ABM of drinking Behavior: A Preliminary Model and Potential Applications to Theory and Practice. American Journal of Public Health. vol .96 (11). pp. 2055-2060. 2007.

[43] J.E. Rowe & R. Gomez. El Botellon: Modelling the Movement of Crowds in a City. Complex Systems vol. 14. pp. 363-370. 2003.

[44] F. Arvidsson and A. Flycht-Eriksson, Ontologies I, Retrieved 26, 2008.

[45] J. Ferber, Multi-agent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley. 1990.

[46] http://ccl.northwestern.edu/netlogo/. U. Wilensky. NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. 1999.

[47] M. Symmons & N. Hawoth. Safety Attitudes and Behaviours in Work-Related Driving, Stage 1: Analysis of Crash Data. MUARC. Report no. 232. 2005.

[48] D. Moore et al. Extending drug ethno-epidemiology using agent-based modelling. Addiction. vol. 104 (12). pp. 1991-1997. 2009.

Agent-Based Model (ABM) Validation Considerations

Philip Cooley, Eric Solano
RTI International
Research Triangle Park, NC, USA
e-mail: solano@rti.org

***Abstract*—This paper describes the use of validation methods in model building. We address issues associated with the increasing complexity of models that is in part a response to the growing popularity of Agent-Based Models (ABM), commonly used to study cognitive, natural, and social phenomena. The first section of this manuscript discusses model categories and attributes. The second section discusses the stages of validating a simulation model: verification, validity, and sensitivity analysis. The third section presents specific validation approaches, with an emphasis on six specific tests that are described in detail. The final section summarizes the goals of model validation and modeling.**

Keywords-Agent-Based Models, Validation, Verification, Infectious Disease Models.

I. INTRODUCTION

A number of global events point out the need for effective modeling. These include the H1N1 pandemic of 2009 and most recently, the Chilean earthquake tragedy, in which observers used modeling to issue tsunami warnings to Hawaii. The tsunami warnings overestimated the effect of the waves that would ultimately reach Hawaii, and “scientists will pore over reams of data” [1] as they work to understand what happened. However, some scientists say that “there should be a rigorous examination of long-standing assumptions within computer-generated models that are used to estimate the strength and impact of tsunamis,” and that the “main problem right now is that we have unsubstantiated assumptions built into our warning system and we really have to check those [1].”

Due to significant reductions in the cost of computational resources and the increasing power of those resources, the nature and type of computer models used in a number of areas including disease transmission processes are changing. In particular, Agent-Based Models (ABM) are a relatively new technology growing in use. One reason is that ABM are an important method for representing and describing interacting heterogeneous agents. Recently, they have been applied to H1N1 infectious disease applications [2-7]. The heterogeneous property of agents enables ABM to describe more sophisticated and complex environments. Many researchers believe that human systems are complex processes that are poorly described by existing/alternative equation-based models (EBM) and it is easier to incorporate existing knowledge about human interactions and decisions into an ABM than into a model described by analytical equations [8]. The downside of this enhanced flexibility is that validating ABM may be more complicated because the processes they describe are more complicated; consequently, rigor is more difficult to achieve because of the complex environment.

A. Validation Definitions

Various definitions of validation appear in the literature. Schlesinger et al. [9] define validation as “substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model.” Midgley et al. [10] define validation as demonstrating that the “correct” equations have been solved by referencing an external and independent test. Macal [11] defines validation as the process of determining the extent to which a model or simulation accurately represents the “real” world from the perspective of its intended use. The final definition of validity presented here is from Ziegler [12], who distinguishes three types of validity:

- replicative validity—the model matches externally available data that has been generated by the modeled system (retrodiction).
- predictive validity—the model matches data that can be acquired from the modeled system, and
- structural validity—the model reflects observed behavior and matches the process inherent to the process to produce the behavior.

B. Model Characteristics

The type of model used to describe the phenomena of interest depends on the nature of the phenomena, the available supporting information about the phenomena, and the purpose of the model. A major issue that affects the type and quality of the validation method that can be applied is the degree of heterogeneity required to describe model elements. In many cases, the level of detail that is incorporated into the model architecture is dictated by the model’s purpose. For example, if intervention strategies to prevent disease spread depend on individual agent characteristics, those characteristics have to be included in the description of the agents. A review of the important categories of models and their characteristics is presented below. These categories are not mutually exclusive.

1) Agent-Based Models (ABM)

ABM have been used to describe phenomena such as social systems and immune systems, which are distributed collections of interacting entities (agents) that function without a leader. Simple agents interact locally according to simple rules of behavior, responding in appropriate ways to environmental cues and not necessarily striving to achieve an overall goal. An ABM consists of a set of agents that encapsulate the behaviors of the individuals that make up the system, and model execution consists of emulating these behaviors [13].

2) Equation-based Models (EBM)

EBM describe the modeled phenomena using a set of equations that interconnect the behavior of individuals or groups of individuals to the environment they inhabit. Manipulating the model’s interconnections allows assessing control scenarios through evaluation of the equations. Historically, an important category of EBM is system dynamics, an approach based on describing simulation processes using ordinary differential equations (ODE) [15].

3) Social Network Models

The structure and dynamics of social networks are critically important to many social phenomena. There are a number of important questions in social networks research, but a lack of data does not allow them to be answered. For example, one of these questions is how social networks change over time.

Social network models are built around two basic entities of a directed graph: the node and the edge. Networks are a form of relational data and arise in many fields, and graphs are a natural method for representing the structure of these relationships. In these applications, nodes usually represent people or agents, and edges represent a specified relationship between them. This framework has many applications, such as assessing the influence of the structure of social networks on the spread of epidemics, assessing the interconnectedness of the World Wide Web, and examining long-distance telephone calling patterns.

4) Deterministic Models

A deterministic model is a mathematical model that employs parameters and variables that are not subject to random fluctuations. Therefore, the system is at any time entirely defined by the initial conditions, in that the assumptions and equations the user selects "determine" the results. The only way the outputs change is if the user changes an assumption (or an equation).

5) Stochastic Models

In many real-life situations, observations are influenced by random effects throughout an entire interval of time or sequence of times. A stochastic model includes elements of randomness that can be introduced at one or many points of the model. Thus, every time the model is applied, a different result is produced even if the parameters and logic are unaltered. Running the model many times provides a measure of the variability in the process that can be captured by the model. In many cases, stochastic models are used to simulate deterministic systems that include smaller-scale phenomena that cannot be accurately observed. The stochastic nature of these types of models is caused by at least three sources: noise in the parameter realization; the representation of a truly random process, and/or a deterministic process that is measured with imprecise tools. The last scenario, though not truly random, produces random-type behavior. In complex systems such as hybrid ABM/EBM, all three sources of randomness could be present. Thus, comparing individual trajectories/outcomes is not straightforward because an infinite number of outcomes are possible. Therefore, a comparison of two stochastic processes should be based on trajectory/outcome generalizations.

6) Monte Carlo Simulation Methods

Monte Carlo models are a class of computational approaches that rely on repeated random sampling to compute results [16]. Monte Carlo methods are often used in simulating physical and mathematical systems. Because of their reliance on repeated computation of random numbers, these methods tend to be used when it is unfeasible or impossible to compute an exact result with a deterministic model. These methods are useful in studying systems with a large number of coupled degrees of freedom and for modeling phenomena with uncertainty in inputs. It is a successful method in risk analysis when compared with alternative methods or human intuition.

II. VALIDATION STAGES

There are three steps in the validation process: (A) verification, which assesses the accuracy of the programmed model; (B) validation, which assesses the accuracy of the phenomena (as described by the model assumptions) against external criteria such as data or other factual information; and (C) sensitivity analysis, which determines the robustness of model estimates with respect to changes in model assumptions.

A. Model Verification

With a complicated computer program, programming errors can result in output that is the result of a mistake rather than a surprising consequence of the model. Verification is the process of checking that a program does what it was planned to do. In the case of simulation, the difficulties of verification are complicated by many simulations being based on a stream of random numbers—meaning every run is different—and it is only the distribution of results that can be anticipated by the theory. Therefore, it is essential to debug the simulation using a set of test cases, perhaps of extreme situations in which the outcomes are easily predicted. Setting up a suite of such test cases and re-running the simulation against them—each time a major change is made—can help ensure that more errors have not been introduced. This process can be made easier by using a version control system that automatically records and tracks model results from each version of the simulation program.

B. Model Validation

Validation processes attempt to demonstrate whether the simulation is a good model of the target phenomena. A model that can be relied on to reflect the behavior of the phenomena is valid. One way to ascertain its validity is by comparing the model's output to data collected from the target. However, a few caveats are warranted:

- Both the model and the target processes are likely to be stochastic, so exact correspondence would not be expected on every occasion. Whether the difference is large enough to cast doubt on the model depends in part on the expected statistical distribution of the output measures. Unfortunately, with simulations, these distributions are rarely known and are not easy to estimate.
- Some simulations are path-dependent and early random number choices can greatly influence outcomes. Outcomes may also depend on the initial conditions chosen, which will affect the paths taken by the simulation.
- Even if the results obtained from the simulation match those from the target, there may be some aspects of the target that the model cannot reproduce.
- A model may be correct but the target data available for validation is either incorrect or not known.
- Data accuracy issues also arise when a model is intentionally highly abstract. Relating the conclusions drawn from the model to specific data from the target may be difficult. In highly abstract models, it is unclear what data could be used for direct validation. This issue arises with models that employ synthetic populations, in which the population is either intentionally remote from the simulation or does not exist at all. For these models, questions of validity are difficult to assess.

C. Model Sensitivity Analysis

Sensitivity analysis investigates how projected performance varies along with changes in the key assumptions on which the projections are based. Once a model appears to be valid, at least for the initial conditions and parameter values for which a simulation has been run, a modeler is likely to consider a sensitivity analysis to answer questions about the extent to which the behavior of the simulation is sensitive to assumptions that have been made. Sensitivity analysis is also used to investigate the robustness of a model [10, 14]. If the behavior is very sensitive to small differences in the value of one or more parameters, a modeler should be concerned about getting accurate estimates for those sensitive parameters.

The principle behind sensitivity analysis is to vary the initial conditions and parameters of the model by a small amount, re-run the simulation, and observe differences in the outcome. This is done repeatedly while systematically changing the parameters. Unfortunately, even a small set of parameters can quickly result in a very large number of combinations of variations in parameter values, and the resources required to perform a thorough analysis can be prohibitive.

Randomization of parameters to obtain a sample of conditions is one of several uses of random numbers in simulation. Random numbers can also be used to: vary exogenous factors (all the external and environmental processes that are not being modeled); model the effects of agents' innate attributes; and address simulation techniques that yield different results, depending on the order in which the actions of agents in the model are simulated.

Results from the simulation will need to be presented as distributions, or as means with confidence intervals. Once a random element is included, the simulation must be analyzed using the same statistical methods that have been developed for experimental research: analysis of variance to assess qualitative changes (e.g., whether clusters have or have not formed), and regression to assess quantitative changes.

III. METHODS

A. The General Process

A model is usually developed to examine a specific set of issues; therefore, model validity should be examined with respect to them. For example, if a disease transmission model is focused on a single epidemic period, and if the pathogen generating the epidemic confers immunity, having the model discriminate between agents that are susceptible to disease and agents that have contracted disease is important. However, if the focus of the study is to determine effective intervention strategies, the outcome of persons contracting disease is unimportant.

Model validation is difficult to make into a structured task. As a model develops, modelers should conduct formal theory predictions (analytical validity) and empirical data comparisons (historical data validity). These tests can be done with varying levels of sophistication. In some cases, looking for simple equivalence is possible. In other cases, running the model hundreds of times is necessary to ensure that the results are robust across a variety of parameter settings.

After designing the model, researchers should spend a substantial amount of time testing model performance under a variety of conditions. Model components can be validated with historical data. Subject area experts can examine the face validity of the predictions to confirm the similarity of model output to their perceptions of how the modeled events should have developed and progressed. Modelers should examine their results to test the implications of the core model assumptions. If possible, they should use real data from external sources and compare model results with the external data.

Modelers should also conduct a set of experiments to set model parameters to their extreme values. Model results using extreme parameter settings should have obvious outcomes.

Once the logical boundaries of the parameter settings are determined, a sensitivity analysis can be performed on all model parameters. In this analysis, model results are generated across a wide range of theoretically feasible parameter settings. This allows the effect of each model parameter on the dependent (outcome) variables to be quantified by generating a numerical estimate of the partial derivative of the outcome variables with respect to changes in the parameter variables.

Simulation models based on ABM use more details to represent a specific model than do those based on EBM. This introduces greater opportunities for validation. Also, using the partial derivative sensitivity estimates as a criterion identifies those parameters that require accurate estimates. Validation of simulation models based on ABM in general should be judged by fidelity, realism, and resolution. These models should be validated on empirical data, as is commonly done for empirical models. Validation is possible through prediction and retrodiction. The quality of the data should be an important criterion for determining the weight of individual validation components. Sensitivity analysis is also necessary for simulations in which parameters are imperfectly measured. Finally, sensitivity analysis should be performed not only on model parameters but also on rules used by the simulation to specify the agent's interaction mechanisms.

B. General Validation Approaches

Many validation approaches have been described in the literature. In general, we will follow the procedures reported in [17]. Schreiber describes four sets of validation tests. We have added sensitivity analysis as a fifth test to assess model robustness. The five tests are defined as follows:

1. Theory-Model Tests determine whether the model describes the conceptions in the minds of the modelers.
2. Model-Model Tests connect the developed model to other pertinent models that describe the same or similar phenomena.
3. Model-Phenomena Tests connect the programmed model to the phenomena that are observed via available data.
4. Theory-Model-Phenomena Tests simultaneously examine the model in the context of both theory and phenomena. Because models, theories, and phenomena often overlap, these categories are more constructed conveniences than concrete truth.
5. Global Sensitivity Tests assess model parameter sensitivity.

C. Validation Tests

A number of validation tests are derived from the general approaches cited above. Note that these validation tests begin after the model has been verified, but in many instances they supplement the model verification processes. Examples of these tests are described below.

1) Calibration

Calibration is the process of tuning a model to fit detailed real data. This is a multi-step, often iterative process in which the model's processes are altered so that the model's predictions come to fit, with reasonable tolerance, a set of detailed real data. This approach is generally used for establishing the feasibility of the computational model; it shows that the model can generate results to match the real data. Calibrating a model may require the researcher to both set and reset parameters and to alter the fundamental programming, procedures, algorithms, or rules in the computational model. To an extent, calibrating establishes the validity of the internal workings of the model and its results.

2) Theory-Model Tests

In Theory-Model tests, the central problem is whether the model matches the theory. As programmed thought experiments, models can have a transparency (assuming the code is written clearly and assumptions are described clearly) that raw theories may lack. Theory-Model tests are also called Cross-Model Validity tests, which emphasize the connectedness of the epistemological framework.

Docking Validity Tests are standard tests of Theory-Model validity. Docking tests use a second model (developed independently) to investigate whether the index model and the second model proceed in like manner or yield similar results. Analytical Validity Tests are similar to Docking Validity Tests, except they compare results from the index model with results from published accounts about the second process and/or the inputs and outputs that are connected to this process [18].

The Face Validity Tests uses the broad knowledge and experience of substantive experts as the source of the data. A model is presented to persons who are knowledgeable about the source problem, and they are asked whether this model is reasonably compatible with their knowledge and experience. The Narrative Validity Test is similar to Face Validity, but it relies on published accounts about the process usually presented by observers of the phenomena. The Narrative Validity Test is amenable to consensus from a team of scholars. Within the context of a group discussion, the group will more likely disagree about whether a model fits their experience than whether it fits a narrative description.

The Turing Test, named for mathematician Alan Turing, examines whether a group of experts can tell the difference between data generated by a model and data generated by the real world. Extreme Point Tests are useful Theory-Model approaches from two perspectives. First, they are an important debugging tool in that they frequently identify subtle code problems. Second, these tests can be used to check model behavior on extreme scenarios.

3) Model-Model Tests

Model-Model tests have a number of variations. In general, these tests involve comparing the index model with other similar models or with theoretical models. In this scenario, a commonly used test is the Cross-Model validity test [19], which validates computational models by investigating whether several models can produce the same results after changing an element/variable in the agent architecture.

Comparing two models allows modelers to recognize significant differences between model results. Identifying the assumptions that caused the differences is an important outcome because it often defines a difference in model assumption or a parameter that is imperfectly known.

4) Model-Phenomena Tests

This category of tests compares the occurrence of specific events represented in the model with the occurrence of the event as represented by real-world data. Comparing model-time series results with the results of previously collected data is one example. Some models forecast results of specific events that follow other events, or alternatively forecast the duration of a specific event. Results from these models can be compared with the actual occurrence of the sequence of the phenomena in the data.

5) Theory-Model-Phenomena Tests

These tests examine the model and the phenomena simultaneously and compare the occurrence of particular events in the model with the occurrence of the events in the source data. Historical Data tests compare model results against previously collected data of some part of the simulated scenario.

6) Global Sensitivity Analysis

Global Sensitivity Analysis tests adjust the parameter settings of the model to determine how sensitive the model predictions are when small changes are made in model parameters. If particular results, such as control strategy predictions, change as a consequence of slightly altered parameter values, then modelers should exercise caution when making claims about model outcomes. Running a comprehensive set of sensitivity analysis tests is not a trivial issue. For example, scientists are confronted with a huge parameter space and very little notion of reasonable parameter values. This requires running many simulations to determine feasible model outcomes. Given a large parameter space, enumerating every possible combination of parameters may be out of the question. This suggests a need for an adaptive process that can steer a search of the parameter space toward more useful/realistic model outcomes.

D. Component Validation Issues

So far, we have discussed tests designed to examine the entire model as a single entity. Testing individual components can also be useful, especially if the social network and agent state change driving force (e.g. disease transmission) components are disjoint entities. In this situation, validating model components allows examining the performance of the model’s individual components; degenerate tests may interrupt some elements of the model and examine the impact on overall results, and trace testing examines individual agents as they work through the modeling environment. Animation methods can support this test to compare the visually displayed qualities of the model with the qualities observed in source data. Trace testing combines our theoretical expectations of the model and our observations about the model and real-world phenomena.

ABM have been criticized because of the large number of assumptions used to implement them. This increases the number of components requiring authentication in the model. However, proponents of ABM might argue that even though detailed models increase the number of component assumptions that have to be reconciled, the assumptions are presented explicitly. Most of us generally understand explicit assumptions and can therefore attempt to validate them. Consequently, they form the basis for judging the validity of one component of a model. However, implicit assumptions are often buried in the logic of EBM and are therefore hidden. In some instances, when implicit assumptions are identified, they are recognized as crude and a necessary evil, with the basic assumptions behind them unchangeable as a part of the fabric of the approach.

ABM and EBM use distinct approaches to describe the same process. They both make a judgment about an identical set of assumptions. ABM represent the assumptions explicitly, while EBM represent assumptions about the same set of processes implicitly, hidden within the fabric of the methodology.

Overall, representing assumptions explicitly allows ABM to expose the weaknesses of the assumptions and define new knowledge requirements for improving model performance.

IV. SUMMARY

Overall, model validation is a common problem in computational modeling of cognitive, natural, or social phenomena: Determining whether the model is the right one and if it captures the essential mechanisms behind the modeled empirical phenomenon is important. As we have seen above, model predictions can be compared with the empirical data to draw conclusions about the plausibility of the model's assumptions. However, this approach does not measure the model's accuracy with respect to unseen data or alternative models designed to explain the same phenomenon. As noted above, there are other methods of validation that can help the modeler, including drawing on the knowledge and experience of subject matter experts.

A related issue is model selection and determining whether a particular model most accurately explains the target phenomenon. Comparing several models and reporting their relative predictions is one way, but this approach often attributes superior performance to inherent model complexity or ad hoc assumptions included in the model.

The goal of modeling is to increase understanding of the underlying mechanisms of the phenomenon; a model that fits the data perfectly does not necessarily capture the essential mechanisms behind the modeled phenomenon. Instead, the model may simply be flexible enough (i.e., over parameterized) to account for the random noise introduced into the model by various means [20].

REFERENCES

[1] Sample H.A. Scientists say tsunami models should be tested. Boston.com. 2010. March 2; News/Science/Articles (col 1). September 12, 2011 <http://www.boston.com/news/science/articles/2010/03/02/scientists_say_tsunami_models_should_be_tested/>.

[2] Longini I. Jr., Nizam A., Xu S., et al. Containing pandemic influenza at the source. Science, 2005; (309):1083-1087.

[3] Ferguson N.M., Cummings D., Fraser C., Wheaton W.D., Cooley, P.C., & Burke, D.S. 2006. Strategies for mitigating an influenza pandemic. 2006. Nature. Jul 27;442(7101):448-52.

[4] Lee B.Y., Brown S.T., Cooley P.C., Zimmerman R.K., Wheaton W.D., Zimmer S.M., Grefenstette J.J., Potter M.A., Assi T., Furphy T., Wagener D.K., Burke D.S. A computer simulation of employee vaccination to mitigate an influenza epidemic. 2010. Am J Prev Med. 38(3):247-257.

[5] Lee B.Y., Brown S.T., Cooley P.C., Potter M.A., Wheaton W.D., Voorhees R.E., Lando J., Stebbins S., Grefenstette J.J., Zimmer Cooley, P., Zimmerman R.K., Assi T., Bailey R.R., Wagener D.K., Burke D.S. Simulating school closure strategies to mitigate an influenza epidemic. 2009 Dec. J Public Health Manag Pract. [Epub ahead of print].

[6] Cooley P.C., Lee B.Y., Brown S., Cajka J., Chasteen B., Ganapathi L., Stark J.H., Wheaton W.D., Wagener D.K., Burke D.S. Protecting health care workers: a pandemic simulation based on Allegheny County. 2010 Feb. Influenza and Other Viruses. 4(2), 61–72

[7] Halloran E.M., Eubank S., Ferguson M.N., Longini, M.I., Barrett C., Beckman R., Burke S.D., Cummings A.D., Fraser C., Germann C.T., Kadau, K., Lewis, B., Macken A.C., Vullikanti A., Wagener K.D., & Cooley P.C. Modeling targeted layered containment of an influenza pandemic in the USA. 2008 Mar. PNAS;105(12): 4639-4644.

[8] Van Dyke Parunak, H., Savit, R., Riolo R.L. Agent-based modeling vs. equation-based modeling: A case study and users' guide.

[9] Schlesinger, S., Crosbie, R.E., Gagne R.E., Innis, G.S., Lalwani, C.S., Loch J., Sylvester R.J., Wright R.D., Kheir N., and Bartos D. Terminology for model credibility. Simulation. 1979. 34(3):103-104

[10] Midgley D., Marks R., Kunchamwar D. The building and assurance of agent-based models: an example and challenge to the field. Journal of Business Research. 2007. Aug;60(8): 84-893. Complexities in Markets Special Issue. doi:10.1016/j.jbusres.2007.02.004.

[11] Macal C. Model Verification and Validation. The University of Chicago and Argonne National Laboratory. Workshop on Threat Anticipation: Social Science Methods and Models. 2005. April 7-9, Chicago, IL.

[12] Ziegler B.P. Theory of Modeling and Simulation. Krieger: Malabar; 1985.

[13] Boero R., Squazzoni F. Does empirical embeddedness matter? Methodological issues on agent-based models for analytical social science. Journal of Artificial Societies and Social Simulation. 2005. 8(4) 6. September 12, 2011<http://jasss.soc.surrey.ac.uk/8/4/6.html>.

[14] Rahmandad H., Sternman J. Heterogeneity and network structure in the dynamics of diffusion: comparing agent-based and differential equation models. MIT Sloan School of Management, System Dynamics Group. May 2005. September 12, 2011 <http://web.mit.edu/~jsterman/www/Rahmandad-Sterman070906.pdf>.

[15] Suter K. The Club of Rome Revisited. ABC Science. 1999. September 12, 2011 <http://www.abc.net.au/science/slab/rome/default.htm>.

[16] Hammersley J.M. Handscomb DC. Monte Carlo Methods. London: Methuen; 1975. ISBN 0416523404.

[17] Schreiber D. Validating agent-based models: From metaphysics to applications. Midwestern Political Science Association's Annual Conference in Chicago. Apr 2002.

[18] Gilbert N., Troitzsh, K.G. Simulation for the social scientist, second edition. Berkshire, UK: Open University Press; 2005.

[19] Sargent R.G. Verification and validation of simulation models. Proceedings of the Winter Simulation Conference; 1998 December 13-16; pp. 121-130.

[20] Laine, T. Methodology for comparing agent-based models of land-use decisions. Indiana University Computer Science Dep. and the Cognitive Science Program, Bloomington. Proc. of the Sixth Annual Int. Conf. on Cognitive Modeling. 2004; 410-411; Mahwah, New Jersey; Lawrence Earlbaum.

Detailed Input data Source for Construction Process Simulation

Approach to connect different data source for discrete-event simulation

Jürgen Melzner, Sebastian Hollermann, Hans-Joachim Bargstädt
Bauhaus-University Weimar, Germany
e-mails: {juergen.melzner, sebastian.hollermann, hans-joachim.bargstaedt}@uni-weimar.de

***Abstract* - Process simulation is an established tool to support planning process. In construction business, process simulation is not gained acceptance for work planning. The reason therefore is the complex development of simulation models because of the lag of comprehensive input data in a suitable form. For meaningful results of process simulation a reliable input data is very important. In construction business scheduling and cost calculation depends in different planning phases on different data source. Developments in object-orientated construction design software offering possibilities to take this information about the object and combine these with process data delivered by dynamic construction data source. This paper presents an approach to integrate the dynamic construction data for input data in a very high level-of-detail. The presented approach is implemented in a discrete-event simulation framework. The objective is a tool for decision support in scheduling and calculation in construction management.**

Keywords-simulation; level-of-detail; process modeling; input data; construction management; scheduling.

I. INTRODUCTION

Modeling and simulation of construction process supports construction planning and can help in reducing the risks concerning budget, time and quality on a construction project [1] [2]. Construction projects are complex because of the need considering different aspects in the planning phase [3]. The client creates a technical specification to define its expectations to the object. These product descriptions define the quality of the product, which has to be built. However, for a detailed job planning it is necessary to know the execution processes in detail. Detailed process chain analyses in construction management are not as well widespread as in stationary industry. Therefore process analysis and description have to done before. In complex process chains with different influences and boundary conditions process simulation is a well established [4]. Process simulation is rarely used for construction process planning because of different aspects. Experts name for the most important impediment for the lacking use of simulation the missing automatically generated input-data [5].

The most important aspect in the development of the methodology is to capture the major requirements that describe the real construction process [6]. This paper presents, how construction processes can be described to deliver valuable entry criteria and input conditions for a simulation model.

This short paper is structured as following: A short summary about the related work in 4D-animation in construction and scheduling with simulation support reproduced in Section II. Following by an explanation about the different planning phases in construction management with different level-of- detail. Section IV explains the two data sources: product data and dynamic construction data. The generated input data will implemented in a discrete-event simulation framework (Section V). Finally we complete with a summary and an outlook for further investigations.

II. LITERATURE REVIEW

A lot of investigations have been done in the area of 4D animation and process simulation. The generation of the input data is crucial according to the many researchers. Huhnt et al. [7] present an approach for data management for animation of construction processes. There modeling approach starts with a decomposition of a building into components. The manufacturing process is described in process-templates and sub-processes. By assigning the building components with the processes a 4D-animation will generate.

In construction projects it is common that drawing, scheduling and estimation are separated planning tasks. Several research projects are initiated to face these challenges. For instance, Kim et al. [8] propose a methodology of developing a 5D system. They link the schedule and costs, which develop in “EVMS” (Earned Value Management System) with the 3D model of a bridge.

Chahrour and Tulke [9] link the 3D model with the time schedule to a 4D Modell. With help of their IFC tool they are able to integrate the process, of time scheduling in a project network. Furthermore they improve a 4D editor and implement their concept to software.

Wu et al. [10] presented a pattern-based approach for facilitating schedule generation and cost analysis in bridge construction projects. Their approach shows a computational method to help project planner to generate the time schedules for bridge automatically. The method has been implemented in a discreet-event simulation environment. They developed a software tool called “Preparator”, which assists the scheduler to assign process patterns to individual elements of a 3D building model. The pattern represents a certain construction method including a number of work packages that the user has to decide to apply.

The presented research projects describe approaches for data management in special fields. On the one hand, the data is used for the visualization and the other for the simulation. The research approach at Bauhaus-University Weimar focus on a data base for input data for simulation. The simulation result can visualized as well. In construction business, process planning is not a static process. Process planning and construction planning will developed dynamically and parallel in different project development phases.

III. Level-of-Detail in different Planning Phases

Tendering and scheduling in construction management is a progress, which is developed over a large period of time and in different level-of detail [11].

A. *Primilinary planing*

The first phase in the development of a construction project is called “primilinary planning”. At the beginning of a project there is only an idea to build an object. There are less information about sizes and quantities. For project managers it is important in this early planning phase on a low level of detail to calculate the costs and construction time in an appropriate way. The figures in this phase are based on experience. For construction projects, they are called *construction cost index* (Baukostenindex-BKI). For different types of buildings the costs per unit are listed.

B. *Tendering*

The “tendering” phase, the construction company calculates the price for the object. The calculation is based on a bill of quantity with a description of the building performance. The bill of quantities defines detail construction quality for all objects. However, there is no virtual connection between processes and objects.

C. *Work planning*

The highest level-of-detail is necessary in the “work planning phase”. For robust work planning it is important to know much information about the object and processes as well. The project profit depends a lot on knowledge of the managers and the available tools for decision support.

Our research objective is tool for decision support in construction management. The tool allows what-if-scenarios and a forecast of resource requirements.

The bases for profound decisions are information. All decisions have influences to costs and time. But in the work planning phase the design of the building is almost completed. Therefore it is necessary to get and use more information for decision making in the early planning stage (Fig. 1).

IV. Input Data

Construction business is typically for the one-of-a-kind business. Every building is build at once. Distinguishing features of the unique processes in the construction industry are according to [12]:

- uniqueness of design and construction
- different materials
- resources availability

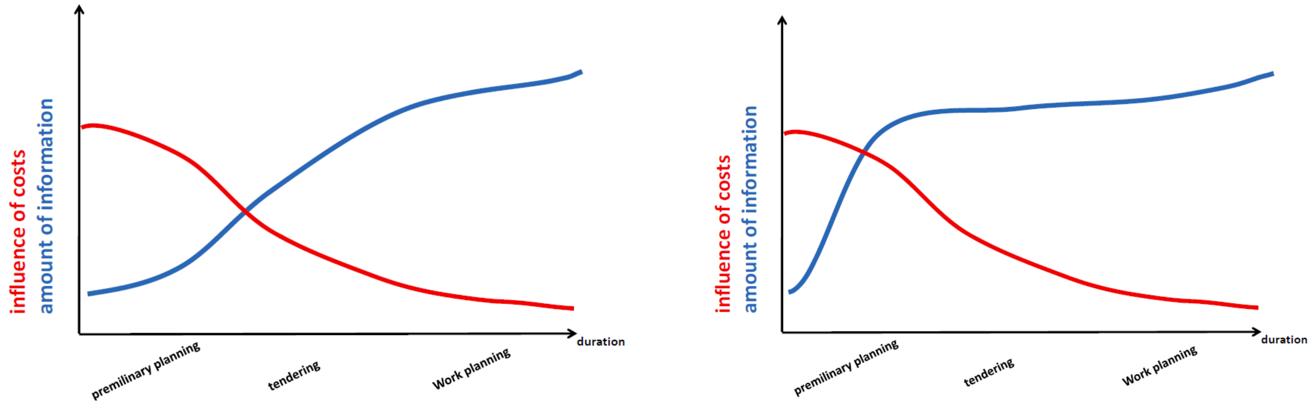


Figure 1. Quantity of information in relation to planning phases; tratitional (left), new apporach (right)

- many disturbances and parameters with stochastic properties
- many different participants
- variety of mandatory and useful dependencies in the processes

Therefore for simulation the construction processes the model, including the input data, has to be created from beginning on. For minimizing the effort for model generation standardized input data is necessary. For Construction simulation there are two main data sources needed. Information about the building including all objects and quantities will import via a Building Information Model (BIM). The construction data with the information about to time, cost, materials and labor recourses will delivered by a specific dynamic data base.

A. Product data

In construction business, 2-dimensional drawings are still usually. New developments in construction software support object-orientated Building Information Models (BIM).

Commercial software programs developed an object orientated modelling environment. The base is different intelligent objects of building types such as walls and windows. These intelligent objects are elements of the product model and can be linked to each other for example that the height of a wall depends on the level of the next ceiling (level). Planning of construction projects in an object orientated environment is helpful for construction management as well as for technical purposes such as clash detection. The objects like walls, columns or decks in an object orientated environment have more information like traditional 3D-Modells. The information is geometrical parameters such as lengths, height, orientation and global coordination. Furthermore, we added information about construction methods on each object. Such as, for constructing a concrete wall you can do this in different ways. The first decision is to build it on site or with prefabricated elements. Such construction relevant information is connected to all objects.

Our research does not replace other researches in the growing field of Building Information Modeling (BIM). It can be considered as a special contribution to additional performances of BIM.

B. Dynamic construction data

For more than 40 years a specific data base for construction work planning establish in Germany. The former standardized book of bill of quantities (Standardleistungsbuch-Bau STLB Bau) offering a vast amount of data about construction materials, construction regulations and construction practices [13]. The objective of this approach is the integration of these big data source for discrete-event simulation (Fig. 2).

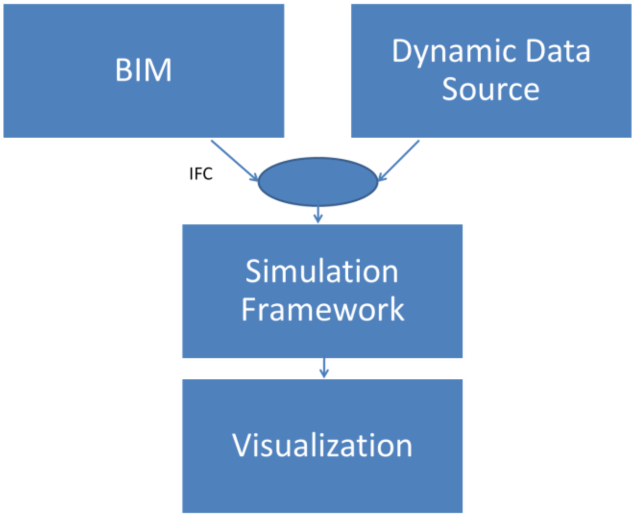


Figure 2. Integration approach

The dynamic construction data base offering a vast amount of structured information of construction processes. The data base structures all construction cost relevant trades. For the simulation studies the process time and costs are required. First each building object has connected with one process from the dynamic data base. The reunite of object and process are implemented in some commercial software as well. But, the process sequences have to create manual. Our approach aims to a definition of constraints in different level-of-detail.

The crucial point for an automatically generated process sequence is correct setting of constraints. This is main development for the simulation framework described in Chapter V. Different planning phases requires different Level-of-Constraints. Therefore we take the existing decomposition of planning phases and define for each level appropriate constraint (Fig. 3).

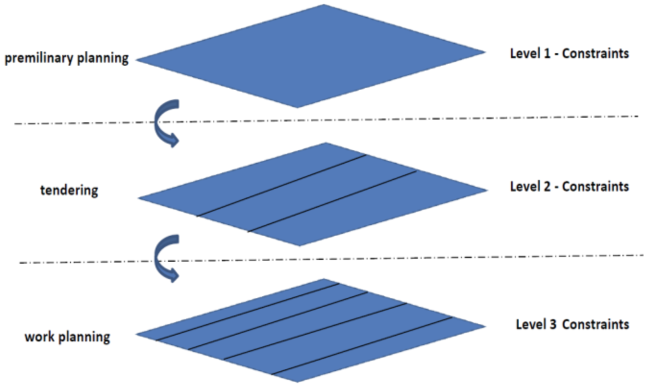


Figure 3. Different level of constraints

V. SIMULATION FRAMEWORK

In this approach, a constraint-based model is used, which has been developed by König et. al. [14] to analyze construction processes. Thereby, the construction tasks and their constraints for production such as technological dependencies, availability and capacity can be specified and valid execution schedules can be generated. The concept has been implemented using discrete-event simulation. That means, only one point of time, at which event occur, are inspected. Within the cooperation SIMoFIT (Simulation of Outfitting Processes in Shipbuilding and Civil Engineering) a constraint-based simulation approach is developed (e.g. [15]). The SIMoFIT cooperation was established between Bauhaus-University Weimar and Flensburger Shipyard, because construction processes in shipbuilding are comparable to building industry [16].

The constraint-based simulation approach is implemented by existing Simulation Toolkit Shipbuilding (STS) and uses the simulation program Plant Simulation provided by SIEMENS PLM. The Flensburger Shipyard and their partners are developing the STS since 2000.

The researcher group of Bauhaus-University Weimar developed and used the STS components for special aspects in construction management. They developed simulation models to investigate the impact of different influencing parameters on the performance of construction activities.

VI. Conclution and Outlook

The described approach shows how an existing huge data source can deliver important data for construction simulation. The data source is the base of high level simulation solutions. The level-of-detail these approach depends on the planning phase, which is to consider. Both the building objects as well as the constraints in early planning phase are on a low level-of-detail. The specification during the project development delivers further information. Thus simulation solution develops from rough to detail as well. For that reasons the solutions can help project manager by decision making.

Acknowledgment

The described approach is mainly developed in the research project MEFISTO, a lead project of the German Federal Ministry of Education and Research (BMWF). We express our thanks for the possibility to develop this approach.

References

[1] P. Song, S. Wang, and S. Abourizk, “A virtual shop modeling system for industrial fabrication shops”, Simulation Modelling Practice and Theory, Vol. 14, 2006, pp. 649-662.

[2] S. Abourizk, D. W. Halpin, and J. D. Lutz, “State of the art in construction simulation”, Proceedings of the 1992 Winter Simulation Conference, Arlington, VA, 1992, pp. 1271-1277.

[3] F. O. Faniran, P. E. D. Love, and H. Li, “Optimal allocation of construction planning resources”, Journal of Construction Engineering and Management, Vol. 125, 5, 1999, pp. 311–319.

[4] C. M. Tam, T. K. L. Tong, and H. Zhang, “Decision making and operations research techniques for construction management, City University of Hong Kong Press, Hong Kong, 2007.

[5] W. A. Günthner and A. Borrmann, „Digitale Baustelle-innovativer Planen, effizienter Ausführen: Werkzeuge und Methoden für das Bauen im 21. Jahrhundert“, Springer, Dordrecht, New York, 2011.

[6] D. W. Halpin and L. S. Riggs, “Planning and analysis of construction operations”, Wiley, New York, 1992.

[7] W. Huhnt, S. Richter, and S. Wallner, T. Habashi, T. Krämer, „Data management for animation of construction processes”, Advanced Engineering Informatics, Vol. 24, 2010, pp.404-416.

[8] H. Kim, C. Benghi, N. Dawood, D. Jung, J. Kim, and Y. Baek, “Developing 5D system connecting cost, schedule and 3D model”, CONVR 2010: Proc. of the 10th International Conference on Construction Applications of Virtual Reality, 2010, pp. 123-129.

[9] R. Chahrour and J. Tulke, “Anbindung der Simulation an eine BIM-Umgebung, Chancen und Anforderungen im Vergleich zur Terminplanung”, Proc. of Simulation von Unikatprozessen: Neue Anwendungen aus Forschung und Praxis (Ed.: Istitut für Bauwirtschaft an der Universität Kassel), 2011, pp. 63-80.

[10] I.-C. Wu, A. Borrmann, E. Rank, U. Beißert, and M. König, “A Pattern-Based Approach for Facilitating Schedule Generation and Cost Analysis in Bridge Construction Projects”, Proc. of the 26th International Conference on IT in Construction & 1st International Conference on Managing Construction for Tomorrow, 1-3 October 2009, Istanbul

[11] S. Hollermann, J. Melzner, and H.-J. Bargstädt, “Flexible Scheduling based on Construction Process Modeling“ Proc. 11th International Conference on Construction Application of Virtual Reality (CONVR 2011). unpublished.

[12] V. Franz, “Unikatprozesse und ASIM-Aktivitäten – Bericht von der Arbeitsgruppe – Unikatprozesse“ Proc. Modellierung von Prozessen zur Fertigung von Unikaten Forschungsworkshop zur Simulation von Bauprozessen, 25.03.2010, Weimar, 2010, pp. 5-16.

[13] StLB-Bau, StLB-Bau-Dynamische Baudaten, 2011, www.stlb-bau-online.de, Retrieved: 25.07.2011

[14] M. König, U. Beißert, and H.-J. Bargstädt, “Ereignis-diskrete Simulation von Trockenbauarbeiten - Konzept, Implementierung und Anwendung”, Proc. Simulation in der Bauwirtschaft: 13. September 2007 an der Universität Kassel (Ed.: V. Franz), Kassel Univ. Press. Kassel, 2007, p. 15

[15] M. König, U. Beißert, D. Steinhauer, and H.-J. Bargstädt, “Constraint-Based Simulation of Outfitting Processes in Shipbuilding and Civil Engineering”, Proc. 6th EUROSIM Congress on Modelling ans Simulation, 2007.

[16] SIMoFIT (Simulation of Outfitting Processes in Shipbuilding and Civil Engineering, http://www.simofit.com/, Retrieved: 25.07.2011.

Energy Simulation Supporting the Building Design Process

A Case Study at the Early Design Stage

Marco Massetti
ARC, Enginyeria i Arquitectura La Salle
Universitat Ramon Llull
Barcelona, Spain
massetmar@yahoo.it

Stefano Paolo Corgnati
TEBE, Department of Energetics
Politecnico di Torino
Torino, Italy
stefano.corgnati@polito.it

***Abstract*—This paper shows a case study proposed in a doctoral thesis currently in progress. The thesis investigates the application of energy calculations to support the design process, ranging from simple energy calculation methods to detailed simulations. In the case study, the design process of an apartment building block in Spain is proposed. Different energy calculation tools are applied at each stage of the project. This paper focuses on the early stages of the design process, in which a simple modelling and calculation approach is adopted. The paper shows, by means of the discussion of the case study, the importance of key factors identified by the authors for the choice of a suitable energy calculation method during the design process.**

Keywords-building design process; energy calculation; energy simulation; key factors; multiple design problems.

I. Introduction

Research and professional experiences of architects and energy specialists [1] reveal that ordinary professional activity rarely involves deep energy analysis and calculations to support the building design process, despite the expected potentials of these tools [1][2][3]. In the field of building energy simulation, McElroy [1] and Clarke [2] proposed to integrate energy simulation with a specific design methodology, but from their experiences they recognized that several barriers still exist. In order to address this issue, it is fundamental to understand which factors must be considered to integrate effectively calculation tools at the different phases of the design process. Few indications are provided by previous investigations in this field [4][5][6]: to face this issue, a doctoral thesis is on going on this topic. The thesis analyses this issue considering energy calculation in the specific context of the building design process: attention is paid to the evolution of the process through different stages, the integration of different competences and the interaction of multiple design problems, not merely quantifiable. In fact, it is fundamental to understand the complexity of the design process: extensive studies exist on the architecture design process and the application of design methodologies [7].

To illustrate this complex problem and to support the investigations, a case study is proposed in this paper.

In particular, in Section II, the design process of an apartment building block in Spain is presented. Different phases are identified: Conceptual and Development Design phases, followed by the Operational phase. In each design phase a suitable calculation tool is adopted - from simple calculation to detailed simulation tools.

In Section III, the paper focuses on Concept design phase. The software Archisun 3.0 [10] is adopted to evaluate building energy and indoor environmental performances. In Section III.A, the project constrains are identified including the constraints imposed by the indoor environmental requirements, by the regulation, by the building program, etc.. In Section III.B, the main decisions at stake in this project at Concept design phase are also identified. In Section III.C, with reference to the defined constraints and decisions, specific hypotheses for energy modelling are established, fixing the boundary conditions and letting the other variable free in order to represent design possibilities to be explored. Section III.D describes how different design alternatives are developed and modelled by the design team, and the corresponding energy performances are obtained by the tool. The obtained performances are considered together with other design aspect, to assess the design solution and take more conscious decisions about the solution to be further developed.

In Section IV, several factors influencing the choice of the suitable tools to support the building design process are identified: these factors are specifically discussed with reference to the case study.

In Section V, conclusion and future perspectives are presented.

II. Case study of a social housing design process

A social housing apartment building block recently completed in Spain is considered. In this paper, the authors wish to replicate retrospectively a possible design process that the design team could have followed using energy calculation to support the design decisions, see Madrazo et al. [8]. The case study intends to exemplify as far as possible an ordinary project, in terms of building use, size and budget. For this reason, at the concept design stage, a small design team working in close cooperation is considered.

A. *The existing building*

The building is a recently completed 24 apartment social housing block, in Cerdanyola del Vallès, Barcelona, Spain, which has been built by the public housing institute Incasol.

The rectangular block is aligned to the street, with the main expositions facing South and North. It is 64 meters long and 12 meters wide. It occupies the maximum surface permitted by the building codes and it has four stories, plus the underground parking. Lobbies and commercial areas are located in the ground floor. The first, second and third floors are addressed to residential use. The typical floor is organized around two cores serving four dwellings each one (Figure 1).

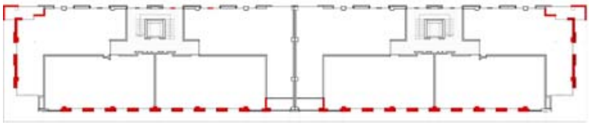


Figure 1. Cerdanyola residential building, typical floor. In red, solar wall in south/east/west façades.

The building design process has been recreated by means of a hypothetical scenario assuming that different energy calculation tools are used according to each specific design stage.

B. Scope and strategy of the case study

The case study, referring to one specific building design process, can support but not demonstrate the validity of theoretical hypotheses about the use of energy calculations in the design process. However, the case study can provide a useful example aimed at discussing the key factors influencing the choice of energy calculation method (see Section IV). Therefore, the research focuses on the application of energy calculation through the process more then analysing in depth energy performances of different solutions.

Moreover, the case study doesn't provide an exhaustive replication of the whole design process. In fact, (a) it focuses on the energy performance of the building, stressing only some (of the innumerable) relations with other design aspects. (b) At each design stage, only some representative solutions (among many ones) are taken into account from more extended design scenarios. (c) Only some specific moments through the different stages of the building life cycle are described.

In line with the arguments of different authors [7][9], it is here assumed that the design team methodology is flexible to the specificity of the project and to the evolution of the design circumstances. This methodology is not necessarily totally explicit and predefined, but it can be partially implicit and embedded in the design team experience, and at some extent improvised according to the project evolution. The main references used to reproduce a realistic design process based on different design stages are INTEND [9] and McElroy [1].

C. Design process overview

Different stages of building life are considered: the building Design – Conceptual Design and Design Development – and the building Operation stage. Energy calculation tools, which support the building Design, are considered for the investigation. Each Design stage is characterized by specific decisions associated with specific project constrains. Both influence the specific hypotheses for energy calculation at each design stage. For the analysed case study, at each design stage, different calculation tools are used to predict building energy performances:

- Archisun, based on a simple energy calculation method
- EnergyPlus (DesignBuilder interface), based on a detailed energy simulation.

At the conceptual design stage, two design alternatives (C1 and C2) are investigated among different solutions, where the second one is a variant of the first one. Archisun is used to predict energy performances.

At the next design stage, a solution from concept design is developed. Two design alternatives (D1 and D2) are considered, as in the previous stage, and EnergyPlus is used to predict energy performances.

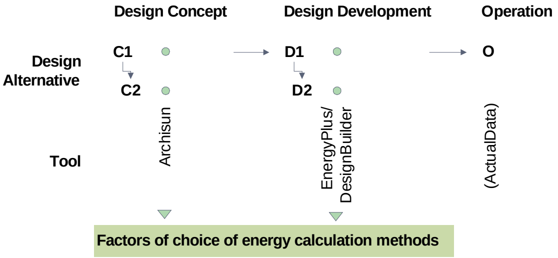


Figure 2. At each phases of design process a specific tool is used: two design solutions (C1-C2 and D1-D2, respectively) are generated through the process and modelled with the corresponding tool.

At the operational stage, the data obtained (by means of measurements, surveys and energy bills) from the real building are considered to verify the mach between predicted and actual energy performance [9].

III. Focus on the concept design stage

This section describes the specific constraints of the project at the concept stage, the decisions to take, and then how they are translated in modelling hypotheses for energy calculation. Finally concept design solution(s) is described. The section structure doesn't correspond to a predefined sequence of steps as it is assumed that different design tasks within the design process are strictly interrelated: consequently, they may occur simultaneously or in aleatory order [7].

A. Project constrains at concept design stage

The following conditions are mostly identified by the design team since the beginning of the concept design phase including: building program (description of the general requirements for the building), budget and specific goals, which are defined by the design team and the client, site conditions and applicable regulation, which depend on the context.

The building program consists of the following points:

- 24 social housing apartments for rent (one of them must be adaptable).

- 2 independent staircases and roofs
- 4 apartments at each floor for each staircase, with maximum useful floor area of 70m^2
- 3 rooms for each apartment, with all rooms visitable
- 5 people for each apartment
- Commercial areas at the ground floor
- 1 underground floor for the garage.

The client budget is 3.170.000 euro.

The main applicable regulation constraints are identified from the technical regulations about construction [11] and systems [12] and the urban regulation [13].

Site conditions are also considered, such as: environmental conditions (thermal, acoustic, lighting, etc.), social conditions (social composition, population density, etc.) and perceptive conditions (as surroundings' views).

Finally, the design team agrees specific project goals with the client, referred to different design aspects, including among the others energy performance goals:

1. Indoor environmental comfort indicators
2. Energy Demand for Heating, Cooling and Domestic Hot Water
3. Primary Energy Consumption
4. Energy Cost
5. Embedded Energy

Energy calculation with Archisun is used at this stage to assess the points 1 and 2, in order to highlight the trend of the performance indicators with the variation of design solutions and to support design decisions (as exemplified in Section D). The aim of the analysis is to compare the effect of different solutions on energy indicators.

B. *Object of design decision at conceptual design stage*

At conceptual design stage, the main decisions regard Building and Systems, while the building Use related factors are not yet considered. In particular the design team explores different options about:

- Building orientation
- Building shape
- Building envelope opening ratios
- Building envelope components performances
- Systems types
- Systems - in situ renewable generation

Section D provides an example of the different Building envelope opening ratios explored during the design process.

C. *Hypotheses for energy calculation at conceptual design stage*

Building program, site conditions and applicable regulation (Section A) are translated to hypotheses for energy calculation. These hypotheses apply to the conceptual design stage and are initially common to any building design alternative (including C1, C2), as they do not initially depend on the design solution.

The parameters affecting energy performances (Figure 3) may be design variables (e.g., U value of the envelope) or they may be design given inputs imposed by the context of the project (e.g., outdoor temperature of the site or input imposed by the regulations, etc.).

The design team defines the same energy calculation boundary conditions for all the building design alternatives. Boundary conditions include all design given inputs imposed by the context and those design variables the team decides to fix as boundary conditions. In fact, some design variables are fixed by the design team as they do not influence the design decisions taken at this stage (see Section B). In Figure 3, the main Building (E) and System (S) characteristics are open to design decisions. Instead, the building Use related factors (User behaviour - U, Indoor environmental quality - I, Operation - O) which are not object of decisions are fixed. Exterior environment data (C) on climate and building surroundings are also fixed, being given inputs imposed by the context.

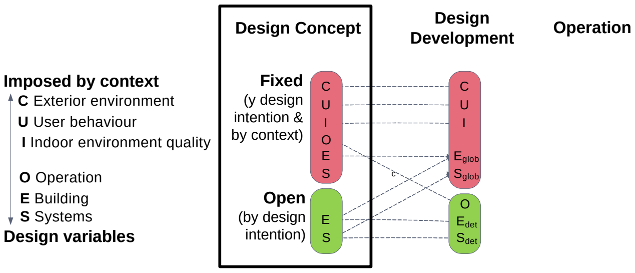


Figure 3. At the concept design stage, the design team fixes some to energy performances variables - red field. While they let main Building (E) and System (S) characteristics open to design decisions - green field.

The calculation tool as well imposes boundary conditions for some energy performance variables which can mach with the boundary conditions assigned by the designer. At this point, it is fundamental the choice of a suitable tool. The design team chooses Archisun (the choice is discussed in Section IV) in a way that the boundary conditions imposed by the tool are coherent, as far as possible, with the boundary conditions they want to define. This logic (outlined in Figure 3) applies to the specific modelling hypotheses of the design alternatives (C1, C2) simulated with Archisun, which are considered hereafter. Input data are summarized in Table I.

Archisun input deta	Value		Comments
Exterior environment (C)			limited options constrainted by the tool
Map position			imposed by project context
Height over sea level	105	m	imposed by project context
Urban density	0.9	-	imposed by project context
…climate data			imposed by context, taken from tool library
Bulding geometry and construction (E)			
Conditioned volume*	4536	m3	depending on program
…envelope data			
System (S)			
…system efficiency data			
User related factors (U, I, O)			limited options constrainted by the tool
Maximum occupancy	120	p	depending on program (=5people*24flets)
Building use	perm.nt		depending on program
temperature set point	f(t)	ºC	imposed by the tool Archisun
...ventilation settings data			imposed by the tool Archisun
...other building use data			imposed by the tool Archisun
*: it includes only appartments (1680 m2 useful floor, 2.7m useful internal height)			

TABLE I. SUMMARY OF ARCHISUN INPUT FOR CONCEPT DESIGN OF THE CASE STUDY. CONSTRAINED INPUT ARE IN RED, UN-CONSTRAINED INPUT ARE IN BLACK

Archisun allows to model one single zone, accordingly with design team intentions. The building is modelled as a single zone representing all apartments, excluding the building areas with different uses and common spaces. The tool imposes the indoor environmental conditions of the adjacent spaces. The values are between the conditioned space and outdoor conditions. The modeller assigns the useful volume value (4536 m3), deduced from the building program, to the "volume" input. The other building data are not fixed as they depend on the explored design possibilities.

Tool inputs data like map position, altitude and urban density are fixed by the context. Inputs, as urban density, not directly available from any source, must be assessed in advance by the modeller. After map position, altitude, sea distance and urban density are given by the modeller, the tool assigns default climate data for four seasonal sequences of typical reference days [10]. Mean external temperature is 9.6ºC in winter (7.5ºC daily variation) and 28.1ºC in summer (8.8ºC variation).

Input data related to building use, imposed by the building program (Section A), are translated to specific values by the modeller: building use is set as "permanent". Then the tool, based on this single input value, assigns a set of default values for each day of a representative week. The tool constraints on detailed use parameters are coherent with the design team intention to constrain building control strategy. At this stage in fact, control strategy is not object of design decisions.

D. Generation and evaluation of concept design alternatives

After the definition of the input parameters (Section A), alternative conceptual solutions are outlined. A large number of alternatives of the design solution are admitted to be explored by the team during concept design. Among them, we describe here only one design solution, C1, and its further modification in another solution, C2, already identified in Figure 2.

The design solution (C1) is generated from multiple initial considerations about the urban regulation constraints, the access from the public space and the climatic aspects.

The design conditions and some initial concept solutions are progressively represented through an evolving sketch. The urban plan is the starting point for the concept formalization. Given the floor surface of the building, urban plan influences significantly Building Orientation (North/South) and Building Shape (4 floors compact block), and the design team has no much possibility to decide about these issues. Therefore, the form of the building rough volume in the project site can be defined very early in the process. Moreover, the building program requires two independents staircases to give access to the apartments. With the typical solution of this region with the stairs internal to the building fabric, some apartments would have only one external façade. This suggested the designer to adopt an external access system to assure two external façades for each apartment.

The solution (narrow layout / double exposition for each apartment) addresses multiple aspects of the design problem.

The uniform linear configuration entails uniformity of solution for all apartments under different aspects, including environmental conditions (in relation with solar radiation and ventilation) as well as internal distribution, views, etc. It also fosters uniform constructive solutions, having implications on construction cost. Meanwhile, the external balcony proposed to solve the access to apartments raises privacy and security issues.

In this moment, the design team decides to deepen some of these aspects: energy demand and indoor environmental conditions. The generated concept solution requires special attention to the façades (with large south and north surfaces), to explore and evaluate appropriate relations between transparent and opaque surfaces of envelope. Energy calculation can inform the designer on the trend of variation of energy demand with the variation of opening ratio.

This is the moment when concept solution is modelled in order to calculate its performances with Archisun. First, the boundary-conditions specified in Table I are modelled. Later, the description of the design solution is completed with the other modelling data initially unconstrained (in Figure 3).

Building envelope is characterized by 15% windows opening ratio in the South façade (C1) and then the ratio is increased to 45% (C2) to explore the effect on the energy demand. The calculation informs the design team on the appreciable reduction of heating demand produced by the variation (Table II). The design team deduces that heat lost through the South façade is compensated by increasing the solar gains (see Section C).

	Opening ratio	**Demand kWh/m^3·y**	
		Heating	***Cooling***
C1	15%	9.43	3.42
C2	45%	8.63	3.65

TABLE II. OPENING RATIO EFFECT ON HEATING/COOLING DEMAND

Meanwhile, calculation informs the design team that C1 and C2 show no significant difference in terms of cooling demand, due to the permanent shading elements on the transparent envelope. Finally, the simple and rapid modelling process and performance visualization facilitate the evaluation of energy performances results together with other aspects of the design problem affected by the opening ratio, such as lighting, privacy and external views.

The simplicity of compared alternative in this scenario permits to focus on the integration of energy calculation through the design process, accordingly with the aim of the research.

IV. DISCUSSION - FACTORS OF CHOICE OF ENERGY CALCULATION METHODS

ASHRAE defined few key factors for the choice of the energy calculation methods and tools [4]. Although, specificity of the different design stages are not stressed. In this paper, different factors are discussed, carefully considering the complexity of the building design process: attention is paid to the transition through different process stages, the convergence of different competences and the interaction of multiple design problems, not merely quantifiable. The factors influencing the choice assume different importance and priority at each design stage.

In the presented case study, each design stage is characterized by specific conditions and decisions at stake, thus specific hypotheses apply for energy calculation, as Section III illustrates for the Concept design stage. Therefore, at each stage a suitable software tool for energy calculations is chosen. The Factors influencing the choice of the energy calculation method are here identified and they are discussed to evaluate Archisun suitability to this design scenario at concept design stage.

- **Level of discretization (detail)** of Archisun building model is quite low. The building is modelled as one single entity with a limited set of attributes. It corresponds to a single zone. Dynamic data are defined on daily basis for few typical days [10]. The limited detail required for modelling input responds to the general decisions considered in concept design stage. It helps to keep under control the design problem/solution and to understand the problem/solution which at this stage is not completely clear. It also limits the time and resources consuming demands of modelling. Modelling detail is limited to the essential for the fulfilment of an acceptable accuracy, cf., McElroy [1]. In the example (Section III-D), only a single input of transparent surface ratio is necessary for each façade to model solution C1-C2.
- **Level of complexity of calculation algorithm** of Archisun is relatively low for automated calculation, thus limiting Accuracy but enhancing Feedback immediacy – a priority at this stage. The limited Level of complexity of the calculation algorithm is expected to enhance the understanding of input variables effect on energy performances.
- **Responsiveness to design decisions** of Archisun is appropriate for the project. The tool inputs correspond to the few main overall variables of Building and Systems, that the designer needs to explore in order to face the typical decisions of this stage, e.g., transparent surface ratio of C1 and C2. In fact, Archisun shell not constraint any project variable explored at this stage (cf., Section III-C). Inputs respond to the specific decisions considered for this stage, made at global level of the building (e.g., global U of the envelope), and do not force to anticipate further decisions (e.g., on detailed components characteristics). Moreover, Archisun outputs provide the performance indicators for the whole building conditioned space, which are useful at this stage to take decisions.
- **Feedback immediacy** of Archisun is high, as its calculation method is relatively simple (cf., Level of discretization and Level of complexity of calculation algorithm). High Feedback immediacy is a priority at this design stage in order to explore rapidly a large number of alternative solutions that are highly uncertain and open (note that C1 and C1 are just a small sample of a more complex scenario). The small design team works in close cooperation and real-time calculations offered by the tool make much easier for the architect to obtain the specialist feedback before moving forward, having immediate communication with the energy specialist, cf., INTEND [9].
- **Flexibility to design modification** of Archisun is adequate to the project concept stage. Modeller can quickly explore modifications of the solution under analysis. A radical concept reformulation, e.g., of the building shape, can be represented with a moderate effort, manipulating a few parameters without a deep re-modelling. This characteristic is strictly related to the Level of discretization and with the Responsiveness to design decisions of the tool (as Flexibility is important specifically for the key parameters for design decisions). Archisun also allows, throughout the concept design, to adjust with a moderate effort the hypotheses for energy calculation boundary conditions initially set down (e.g., Conditioned volume), manipulating few parameters without a deep re-modelling. Flexibility is lost in case design modifications impose to skip from a higher to a lower level of input complexity: in this case several data must be re-introduced.
- **Flexibility to solution representation** indicates that Archisun model is able to suitably represent the conceived alternative solutions at this design stage. The model input, envelope opening ratio, represents directly the solution (C1/C2) conceived by the designer, without any stretching. The low Flexibility of this tool in the representation of building Use related factors does not limit the design solution exploration, as the design team decided that Use related factors are not object of design decisions at this stage.
- **Accuracy** of Archisun is expected to be acceptable for a residential building project (with simple HVAC system) at this design stage (cf., Level of discretization). At this stage, high Accuracy in energy performance prediction is not the first priority. But a minimum Accuracy is necessary to correctly point out and compare the different performance of design alternatives C1 and C2. The approach of Archisun versus the uncertainty of some parameters is of fixing to default values some quantities (in particular, building Use related factors), instead of entrusting them to the concept designer discretionality (see constrained variables in Section III-C) in order to avoid arbitrary modelling assumptions. Archisun sensitivity analyses are provided by Palme [14].
- **Integrability in multiple design problem** of Archisun is facilitated by its low Discretization, which limits the design team resources dedicated to energy analysis and subtracted to other design aspects. Archisun is characterized by a high Feedback immediacy, which allows a rapid shift

from one problem domain to another, and by a wide range of multiple performances that can be calculated (thermal, lighting, acoustic comfort and energy). Integrability facilitates a better understanding and control of the overall problem: this is crucial when energy and environmental performances are considered together with all the other aspects of the design problem.

- **Data coherence preservation capability** is facilitated by the double level of complexity of Archisun input model which intends initially to handle overall solutions and later to refine them. Thus, in the later refinement the initial data are preserved. Nevertheless, in this scenario different tools are used at each design stages, in order to fulfil with their different requirements. Therefore, Data coherence preservation does not depend only on Archisun. Data coherence passing from one stage/tool to the next is actually not easy to solve. In particular, the transition of energy calculation input from one tool to the other is demanding, but it cannot be simplified as a matter of tools limitation. In fact, driving the solution from some generic model to more specific one (e.g., from one single U for façade to specific components properties) is right the role of designers. A limitation of Archisun is its lack of transparency about the building use related data, that consequently can hardly be reproduced coherently with Energy Plus in the next design stage.

According to most of the considered factors, Archisun seems appropriate for concept design stage. Nevertheless, many of these factors are strictly interrelated, controversial and in some cases one is conflicting with another.

V. Conclusion and Future Work

The case study highlights the role of several factors identified in this paper as predominant for the choice of suitable energy calculation method to support the design process. The proposed key factors pretend to make a step forward in the line of precedent literature indications [4][5][6], as their discussion in the case study intends to explain. In fact, existing energy calculation tools and the underlying methods are many and very different, and a good choice, based on their specific characteristics, is fundamental to foster the exploitation of energy calculation potentials through the entire design process. In particular, the factors for the choice of suitable energy calculation methods are specific for each design stage, as shown in the case study. Finally, the priority and the evaluation (favourable or not to the choice) of these factors vary according to the specificity of the individual project and of the design process evolution – not totally predictable a priori.

Therefore, it is suggested that the factors of choice should provide solid applicable principles, which are flexible to the specificity of the project. In fact, their purpose is not to impose rigid and deterministic rules, universally applicable in advance to any project. With this regard, a remark is addressed to the fact that most of these factors are strictly interrelated, someone is controversial and in some cases one is conflicting with another, therefore this procedure does not provide an absolute judgment.

Within the future work, authors intend to

- extend the analysis to the whole design process, including the design development phase
- improve and provide more specific definitions and discussions about the key factors identified in this paper.
- corroborate the hypothetical scenario with the feedback from direct experiences of different practitioners in Europe.

VI. Acknowledgment

This work was developed within the research line of the group ARC, related with the projects IntUBE (7th Framework Programme, 2008-2011) and REPENER (Spanish National RDI plan, 2009-2012), on ICT and energy assessment through the entire building life-cycle. M. Massetti thanks M. Palme for contributing to the initial ideas of investigating key characteristics of energy calculation tools.

References

[1] L. B. McElroy, Embedding integrated building performance assessment in design practice, PhD Thesis, University of Strathclyde, 2009.

[2] J. A. Clarke, Energy simulation in building design, Oxford: Butterworth-Heinemann, 2001.

[3] C. J. Hopfe, C. Struck, G. Ulukavak Harputlugil, J. Hensen and P. Wilde, "Exploration of using building performance simulation tools for conceptual building design", in Proc. IBPSA-NVL conference, 2005, p8

[4] ASHRAE, ASHRAE handbook: fundamentals. Atlanta, 2009

[5] ISO 13790, Energy performance of buildings - Calculation of energy use for space heating and cooling, 2008

[6] J. P. Waltz, Computerized building energy simulation handbook, Monticello, N.Y.: Marcel Dekker, 2000.

[7] B. Lawson, How designers think: the design process demystified, 4th ed., Oxford: Architectural Press, 2006.

[8] L. Madrazo, M. Massetti, G. Font and I. Alomar, “Integrating energy simulation in the early stage of building design,” Proc. IBPSA-Germany Conference on Building Performance Simulation in a Changing Environment (BauSIM 2010), BPSA-Germany, 2010, pp. 175-182

[9] INTEND, Integrated Energy Design. IED, 2009, 03.06.2011: www.intendesign.com

[10] R. Serra, J. Roset, “Energy Conscious Design,” Proc. World Renewable energy Congress VI (WREC VI), July 2000, pp. 494-499.

[11] Código Técnico de la Edificación. Madrid: Ministerio de la Vivienda, 2006.

[12] Real Decreto 1027/2007, “Reglamento de Instalaciones Térmicas en los Edificios” BOE, nº 207, pp. 35931-35984, Agoust 2007.

[13] Modificació del pla parcial centre direccional de Cerdanyola del Vallès, Unitat de planejament de l’àrea de sòl, October 2005.

[14] M. Palme, La Sensibilidad energética de los edificios, PhD Thesis, Universitat Politècnica de Catalunya, 2010.

Simulating Counterinsurgency and Coalition Strategies

David C. Arney
Department of Mathematical Sciences
United States Military Academy
West Point, USA
Email: david.arney@usma.edu

Kristin M. Arney
Department of Mathematical Sciences
United States Military Academy
West Point, USA
Email: kristin.arney@usma.edu

***Abstract*—We model insurgency (IN) and counterinsurgency (COIN) operations with a large-scale system of differential equations that is connected to a coalition network model. Our simulations analyze components of leadership, promotion, recruitment, financial resources, operational techniques, network communications, coalition cooperation, logistics, security, intelligence, infrastructure development, humanitarian aid, and psychological warfare, with the goal of informing today's decision makers of the options available in counterinsurgency tactics, operations, and strategy. In order to be more effective, the US military must improve its counterinsurgency capabilities and flexibility to match the adaptability of insurgent networks and terror cells. Our simulation model combines elements of traditional differential equation force-on-force modeling with modern social science modeling of networks, psyop, and coalition cooperation to inform the tactics and strategies of counterinsurgency decision makers. We calibrated our model with baseline data intended to keep the balanced strength equilibrium. We show the model development and results of a four-stage counter-insurgency scenario.**

Keywords- counterinsurgency; force-on-force model; differential eqiation; network model

I. The COIN Model

In modeling an insurgent or terrorist organization, we modify the differential equation model in [1] that tracks several groups within the terrorist organization: senior leaders (*l*), junior leaders (*j*), outside supporters (*o*), bomb-makers (*b*), and foot soldiers (*f*). The model also includes equations for the intensity of several terrorism factors: financial support for the organization (*m*), intellectual level of the organization (*i*), public (in-country, local) support for the organization or cause (*p*), and world-wide support for the cause (*w*). These elements all factor into the overall strength of the terror organization (*s*). Considering counterinsurgency factors (all in upper case), we model: public support for the counter-terrorism effort (*C*), the cooperative coalition (multi-national/multi-agency) effort (*CC*), aggressiveness of direct CT operations (*D*), aggressiveness of intelligence gathering (*G*), aggressiveness of PSYOP/information distribution (*P*), aggressiveness of aid to the local public/host country government (*A*), aggressiveness of US/coalition logistics (*L*), and aggressiveness of US/coalition security (*Y*). We then combine these COIN measures to determine the overall strength of the COIN operations (*S*). The model consists of 19 dependent factors with 19 equations and over 80 parameters. The roots of this differential equations model stem from ideas in [2, 3, 4, 5]. Many of the primary factors discussed in [6] for terrorism (T) and counter-terrorism (CT) operations are included in this model. These equations are dynamic and time dependent as we use time (*t*) as our one independent variable.

II. Coalition Effectiveness And Collaboration

One of the most important aspects of counterinsurgency operation is the effectiveness of the coalition of organizations and agencies involved in the operation [7]. For the purposes of this simulation, we use a coalition network model that consists of three subgroups: US agencies (governmental and nongovernmental), host country organizations, and world-wide organizations (other countries forces and agencies, world-level nongovernmental organizations, and UN organizations).

COIN, IN, T and CT operations involve not only power, force, control, and other military-based components, but also diplomatic and nation-building elements of influence, politics, legitimacy, and service [6, 8, 9, 10, 11]. The agencies that work with the populace along with the military forces form the COIN/CT coalition that wages the counterinsurgency. FM3-24, page 2.1, explains the roles these coalition partners play to succeed in COIN: "Although military efforts are necessary and important, they are only effective if integrated into a comprehensive strategy employing all instruments of national power…. The integration of civilian and military efforts is crucial in COIN and must be focused on supporting the local population and the HN government. Political, social and economic programs are usually more valuable than conventional military operations as a means to address root causes of conflict and undermine an insurgency. In COIN, military personnel, diplomats, police, politicians, humanitarian aid workers, contractors, and local leaders are faced with

ISBN: 978-1-61208-169-4

making decisions and solving problems in a complex and acutely challenging environment" [6].

The coordination of effort and cooperation the coalition network is essential. The JP 3-24 explains: "Unified action refers to the synchronization, coordination, and/or integration of military operations with the activities of governmental and nongovernmental entities to achieve unity of effort. The military contribution to COIN must be coordinated with the activities of the United States Government, interagency partners, IGOs (Intergovernmental Organizations), NGOs (Nongovernmental Organizations), regional organizations, the operations of multinational forces, and activities of various HN (Host Nation) agencies to be successful…Successful interagency, IGO, and NGO coordination helps enable the USG to build international support, conserve resources, and conduct coherent operations that efficiently achieve shared goals." [9].

In summary from page 2-1 of FM 3-24, "The preference in COIN is always to have civilians carry out civilian tasks. Civilian agencies of individuals with the greatest expertise for a given task should perform it – with special preference for legitimate local civil authorities… the preferred or ideal division of labor is frequently unattainable. The more violent the insurgency, the more unrealistic is this preferred division of labor" [6].

III. The Coalition Network Model

In order to compute viable measurements for the effectiveness of the coalition, we represent the coalition with a network structure. We model the various organizations as nodes and the strength of the collaboration between the organizations as weighted edges. More precisely, the weights on the edges are the percent of the perfect or desired collaboration between the two connecting organizations. As indicated in [12], some organizations should maintain an intense collaboration with another organization because of the nature of their missions, whereas others may have little need to collaborate in COIN except to maintain communication of basic information. Therefore, in our model, a coalition network with perfect collaboration for their suited purposes is a completely connected graph with all its weighted links all set to 1 (or 100% effective collaboration). A completely dysfunctional coalition with none of the effective collaboration needed is modeled by a completely disconnected network graph.

Our network metrics measure the strength of collaboration. A coalition's collaboration strength (*CCS*) is the weighted density measure of the graph. For a undirected graph, the sums of all the weighs of connecting edges (Σe_k, where k goes from 1 to Z, the total number of possible connections) are divided by the total possible connections of the graph $Z=(M)(M-1)/2$, where M is the number of nodes in the graph or total number of agencies in the network. Subgroups of the overall coalition produce two collaboration measures, its own internal collaboration strength (*ICS*) measured by only taking into account the network of the subgroup and the external collaboration strength (*ECS*) by taking into account the weights of links between the subgroup and its complement.

IV. Using The Network Metrics In The Coin Model

As indicated in the model description, one of the key elements in CT/COIN success and a major component in our model is the Cooperation/Coalition factor (*CC*). We use a coalition network model with three subgroups of 1) US --- US forces and organizations (governmental and nongovernmental), 2) Host -- host country forces and organizations, and 3) World -- world-wide forces and organizations (other countries forces and agencies, world-level nongovernmental organizations, and UN organizations) to calculate the metrics to use in our COIN model. The ten network metrics we use are the seven Coalition Network Metrics of CCS, $\mathrm{ICS}_{\mathrm{US}}$, $\mathrm{ECS}_{\mathrm{US}}$, $\mathrm{ICS}_{\mathrm{World}}$, $\mathrm{ECS}_{\mathrm{World}}$, $\mathrm{ICS}_{\mathrm{Host}}$, $\mathrm{ECS}_{\mathrm{Host}}$, the Link Density (LD). CC is computed as a weighted sum of these elements of the CT coalition network while also being proportional to the levels of aggressiveness of security (*Y*), aggressiveness of intelligence gathering (*G*), aggressiveness of PSYOP (*P*), aggressiveness of US aid (*A*), aggressiveness of CT logistics (*L*), number of nations in the coalition squared N^2, and number of total organizations in the coalition as shown in (1). The non-linear squared term for the number of nations is the key part of this measure showing the important nature of that aspect of Coalition strength. The *CC* factor is an influential component of our dynamic COIN model

$$CC = \eta_1 N^2 + \eta_2 M + \eta_3 LD + \eta_4 CCS + \eta_5 ICS_{US} + \eta_6 ECS_{US} + \eta_7 ICS_{Host} + \eta_8 ECS_{Host} + \eta_9 ICS_{World} + \eta_{10} ECS_{World} + \eta_{11} G + \eta_{12} P + \eta_{13} A + \eta_{14} L + \eta_{15} Y + \eta_{16} D \quad (1)$$

V. COIN Scenario Using Coalition Networks

To show the effects of the dynamics of the Coalition Network on the COIN model, we simulate a four-stage scenario of Coalition evolution. Since each stage affects the COIN results, we will show the graph of the coalition network model, the computed collaboration metrics, and the results of running the COIN model for the six-month duration at each stage. For this scenario, we keep all six of the resource levels equal and constant at 0.83 to run a balanced COIN strategy.

A. Stage 1: The Initial Coalition (9 nodes)

We start with the US Forces arriving in a Host country to form a small, weakly connected coalition with several Host country organizations. This coalition has no elements outside those of the US and the Host country. The Coalition is modeled by the 9-node network shown in Fig.1. We track the three subgroups US, Host, and World. Subgroup Host contains three nodes, Subgroup World has no nodes, and Subgroup US contains six nodes. From the

ISBN: 978-1-61208-169-4

collaboration weights, we compute the seven possible collaboration strength (CS) scores of CCS, ICS_{US}, ECS_{US}, ICS_{World}, ECS_{World}, ICS_{Host}, ECS_{Host}. The CCS is computed as 4.5/36= 0.125. For the Host Subgroup, ICS_{Host} = 1.3/3 = 0.43 and ECS_{Host}=0.6/18=0.03. For the World Subgroup, ICS_{Worls}= ECS_{World}=0, since there are no World organizations in the Coalition. For the US Subgroup, ICS_{US}=2.6/15=0.173 and ECS_{US}= ECS_{Host}=0.6/18=0.03, since there are only two subgroups present in the network, the External Collaboration scores must be the same. The LD is 12/36=0.33. We run the COIN model for 6 months to obtain the results shown in Table I along with the coalition metrics. The collaboration scores show that the US and Host country do not yet collaborate very effectively.

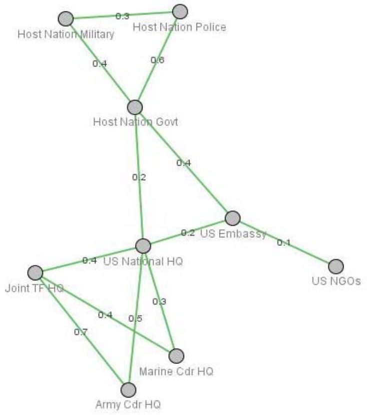


Figure 1. The Coalition Collaboration Network for Stage 1.

TABLE I. RESULTS OF STAGE 1

Collaboration Metrics for the Stage 1 Coalition (N=2 (US and Host), M=9)										
US nodes	Host Nodes	World Nodes	CCS	ICS_{Host}	ECS_{Host}	ICS_{World}	ECS_{World}	ICS_{US}	ECS_{US}	Link Density
6	3	0	0.125	0.43	0.03	0	0	0.173	0.03	0.33
COIN Model Metrics for 6 months with Stage 1 Coalition										
CC	s	S	S/s ratio	change of S/s	% change in o	% change in m	% change in i	% change in p	5 change in w	% change in C
0.4126	0.765	0.817	1.07	-0.003	0	0.014	0.005	-0.01	0.015	0.003

B. *Stage 2: The Coalition Grows: World Organizations and Allied Force Arrives (16 nodes)*

At this stage, the coalition has added more US forces, maintained the same basic Host nation involvement, and added one other allied country force along with some UN and world-wide organizations. The model for this rudimentary coalition of 16 nodes with the weights of the collaborations is shown in Fig. 2.

This modest growth in the coalition increases the collaboration strengths from Stage 1. The Subgroup Host contains 3 nodes, Subgroup World has 6 nodes, and Subgroup US contains 7 nodes. We compute the seven possible collaboration strength (CS) scores of CCS, ICS_{US}, ECS_{US}, ICS_{World}, ECS_{World}, ICS_{Host}, ECS_{Host}. The CCS is computed as 11.4/120= 0.095. For the Host Subgroup, ICS_{Host} = 1.4/3 = 0.47 and ECS_{Host}=1.6/39=0.04. For the World Subgroup, ICS_{Worls}=2/15=0.133 and ECS_{World}=2.6/60=0.043. For the US Subgroup, ICS_{US}=4.3/21=0.20 and ECS_{US}=2.4/63=0.04. The LD is 27/120= 0.23.

We run the COIN model for six months to obtain the results shown in Table II along with the coalition metrics. These results show that the collaboration has improved with a higher CC score.

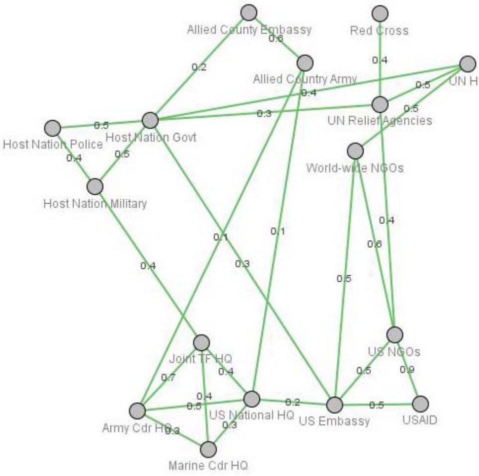


Figure 2. The Coalition Collaboration Network for Stage 2.

ISBN: 978-1-61208-169-4

TABLE II. RESULTS OF STAGE 2

Collaboration Metrics for the Stage 2 Coalition (N=3, M=16)										
US nodes	Host Nodes	World Nodes	CCS	ICS_{Host}	ECS_{Host}	ICS_{World}	ECS_{World}	ICS_{US}	ECS_{US}	Link Density
7	3	6	0.095	0.47	0.04	0.133	0.043	0.20	0.04	0.23
COIN Model Metrics for 6 months with Stage 2 Coalition										
CC	s	S	S/s ratio	% change of S/s	% change in o	% change in m	% change in i	% change in p	5 change in w	% change in C
0.7372	0.767	1.189	1.549	0.45	0	.001	-0.005	-0.01	0.01	0.05

C. *Stage 3: The Coaltion Expands (47 nodes)*

During this stage the Coalition grows substantially to 47 organizations and five countries, but they are still sparsely linked with little collaborations across the three subgroups. One of the countries is involved diplomatically, but not militarily and contributes one node to the network ("Involved Country Embassy"). This Coalition network is shown in Fig. 3.

Subgroup Host contains 7 nodes, Subgroup World has 17 nodes, and Subgroup US contains 23 nodes. We show the collaboration and coalition metrics in Table III. We run the COIN model for 6 months to obtain the results shown in Table III.

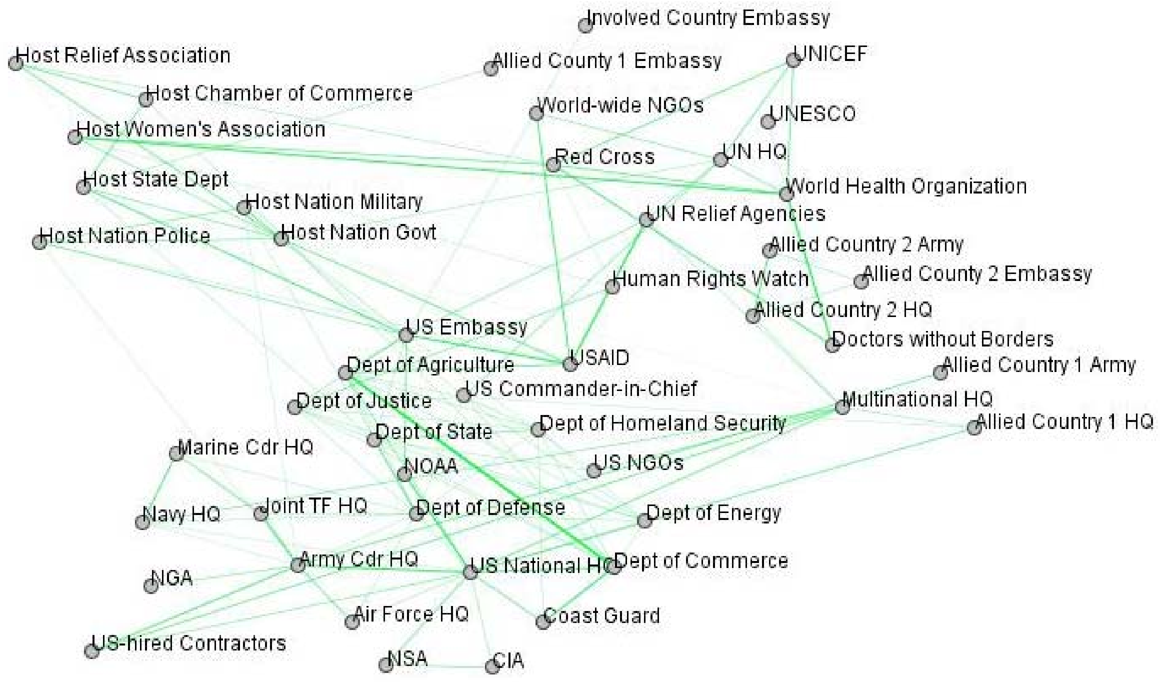


Figure 3. The Coalition Collaboration Network for Stage 3.

ISBN: 978-1-61208-169-4

TABLE III. RESULTS OF STAGE 3

Collaboration Metrics for the Stage 3 Coalition (N=5, M=47)										
US nodes	Host Nodes	World Nodes	CCS	ICS_{Host}	ECS_{Host}	ICS_{World}	ECS_{World}	ICS_{US}	ECS_{US}	Link Density
23	7	17	0.04	0.18	0.02	0.05	0.015	0.09	0.017	0.114
COIN Model Metrics for 6 months with Stage 3 Coalition										
CC	s	S	S/s ratio	% change	% change in o	% change in m	% change in i	% change in p	5 change in w	% change in C
1.7872	0.766	2.393	3.126	1.02	-0.01	-0.01	-0.04	-0.01	-0.005	0.02

The rapid growth in the coalition results in a CCS of 0.04, since the collaboration total is just 44.7 out of a possible of 1081. Also, the coalition has a LD of 123/1081=0.114. Just a little over 11% of the possible coordination links are even established by the coalition. The increased size of the coalition (five counties and 47 organizations) and the growing strengths of the three subgroups have resulted in the large increase in the CC value. This increase in CC leads to small decreases in the insurgency measures and a large increase in the strength of the counter insurgency. The effect is that the S/s ratio doubles during this Stage.

D. *Stage 4: The Military Forces Coalesce and Strengthen their Collaboarations (49 nodes)*

In this stage the military forces are able to coordinate their work within and between the US, the three Allied counties, and Host nation. Only two new organizations enter the coalition in this stage. Since the Involved country has now committed military forces, two new organizations (Allied Country 3 HQ and Allied Country 3 Army). Most of the effort during this stage has been to strengthen existing military collaborations. This new stronger Coalition network is shown in Fig. 4 with results provided in Table IV.

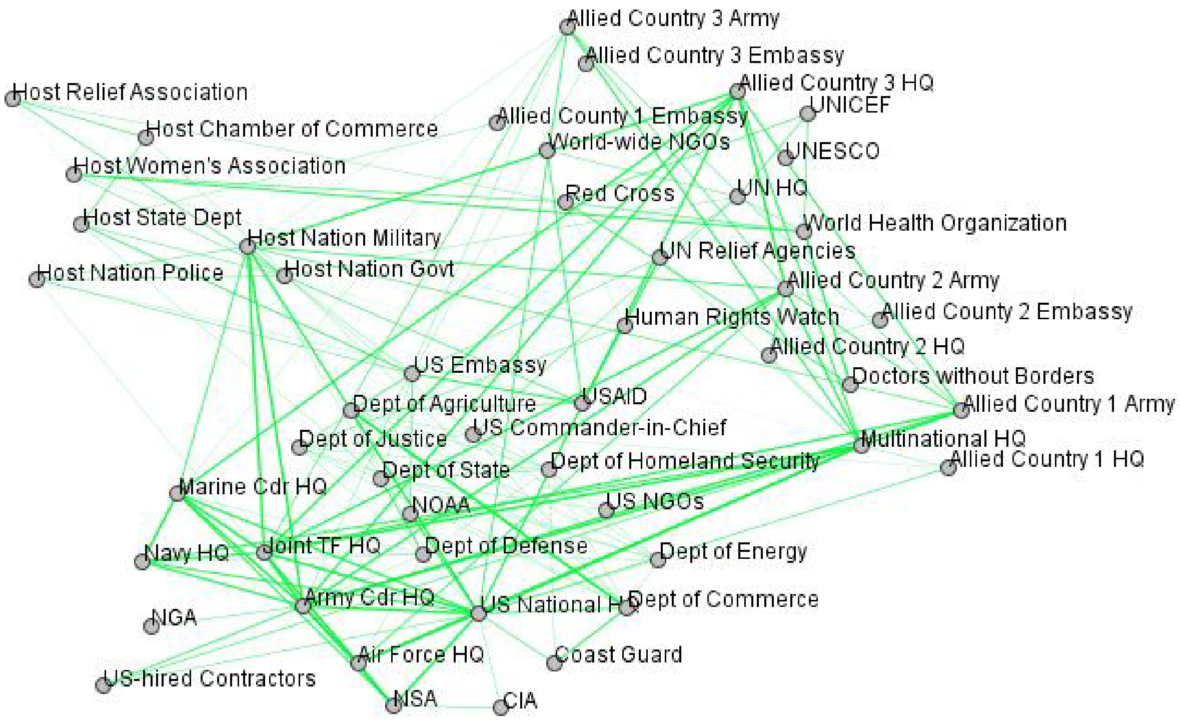


Figure 4. The Coalition Collaboration Network for Stage 4.

ISBN: 978-1-61208-169-4

TABLE IV. RESULTS OF STAGE 4

Collaboration Metrics for the Stage 4 Coalition (N=5, M=49)										
US nodes	Host Nodes	World Nodes	CCS	ICS_{Host}	ECS_{Host}	ICS_{World}	ECS_{World}	ICS_{US}	ECS_{US}	Link Density
23	7	19	0.066	0.18	0.04	0.07	0.04	0.13	0.035	0.14
COIN Model Metrics for 6 months with Stage 4 Coalition										
CC	s	S	S/s ratio	% change of S/s	% change in o	% change in m	% change in i	% change in p	5 change in w	% change in C
1.8208	0.7613	2.449	3.222	0.03	-0.01	-0.01	-0.06	-0.005	-0.01	0.03

The increased cooperation of the military forces in the coalition results in increases in all eight collaboration metrics. The COIN operation is starting to show is strength in the model and affect the insurgency elements --- all of which are decreasing.

VI. CONCLUSION

We used this rather small and simple scenario to simulate COIN operations in a dynamic environment to show the functionality of our coalition network model and its interface to the differential equations model. The mathematical issues of combining large networks and large systems of differential and algebraic equations are not known. However, we see this combination as giving us better insights into the complexity of warfare. Our hybrid model (force-on-force, COIN factors, and coalition network model) enables study of the most feared and possibly likely war of the future – a hybrid war. As described in [13 and 14], these full spectrum conflicts will involve many elements of COIN-CT-and full force-on-force operations along with the psychological aspects of conflict on the US populace, basic elements of which are found in our model.

REFERENCES

[1] C. Arney, Z. Silvis, M. Thielen, and J. Yao, "Modeling the complexity of the terrorism/counter-terrorism struggle: mathematics of the 'hearts and minds'," International J. of Operations Res. and Information Systems, in press.

[2] A. Gutfraind, "Understanding terrorist organizations with a dynamic model," Studies in Conflict and Terrorism, 32:1, 2009, pp. 45–59

[3] F. Udwadia, G. Leitmann, and L. Lambertini,. "A dynamical model of terrorism," Discrete Dynamics in Nature & Society, 2006, pp. 1-31.

[4] W. P. Fox, "Discrete combat models: investigating the solutions to discrete forms of Lanchester's combat models," International J. of Operations Res. and Information Systems**,** 1(1), 2009, pp. 16-34.

[5] Committee on Organizational Modeling from Individuals to Societies. Behavioral modeling and Simulation: from individual to societies. National Research Council, 2008.

[6] FM 3-24, Counterinsurgency, Department of the Army, 2006.

[7] B. Ganor, The Counter-terrorism Puzzle: guide for decision makers, Transaction Publishers, NJ, 2005.

[8] FM 3-24.2, Counterinsurgency Tactics, Department of the Army, 2009.

[9] JP 3-24, Counterinsurgency Operations, Department of Defense, 2009.

[10] JP 3-08, Interagency, Intergovernmental Organization, and Nongovernmental Organization Coordination during Joint Operations, 17 March 2006.

[11] JP 3-16, Multi-national Operations, 7 March 2007.

[12] Joint Special Operations University, Special Operations Forces Interagency counterterrorism reference manual (2nd editon), JSOU Press, 2011.

[13] J. J. McCuen, ""Hybrid wars, Military Review (March-April), 2008, pp. 2-8.

[14] M. Cancian. "Winning hearts and minds at home," Proceedings, US Naval Institute. 2010.

ISBN: 978-1-61208-169-4

Simulation of Bacterial Self-Organization in Circular Container Along Contact Line as Detected by Bioluminescence Imaging

Romas Baronas
Faculty of Mathematics and Informatics
Vilnius University
Naugarduko 24, LT-03225 Vilnius, Lithuania
Email: romas.baronas@mif.vu.lt

Abstract*—Simulation of quasi-one dimensional spatiotemporal pattern formation along the three phase contact line in the fluid cultures of lux-gene engineered *Escherichia coli* is investigated in this paper. The numerical simulation is based on a one-dimensional-in-space mathematical model of a bacterial self-organization as detected by quasi-one-dimensional bioluminescence imaging. The pattern formation in a luminous *E. coli* colony was mathematically modeled by the nonlinear reaction-diffusion-chemotaxis equations. The numerical simulation was carried out using the finite difference technique. Regular oscillations as well as chaotic fluctuations similar to experimental ones were computationally simulated. The effect of the signal-dependent as well as density-dependent chemotactic sensitivity on the pattern formation was investigated. The simulations showed that a constant chemotactic sensitivity can be applied for modeling the formation of the bioluminescence patterns in a colony of luminous *E. coli.

***Keywords*-chemotaxis; reaction-diffusion; pattern formation; whole-cell biosensor.**

I. Introduction

Microorganisms respond to different chemicals found in their environment by migrating either toward or away from them. The directed movement of microorganisms in response to chemical gradients is called chemotaxis [1]. Chemotaxis plays crucial role in a wide range of biological phenomena, e.g. within the embryo, chemotaxis affects avian gastrulation and patterning of the nervous system [2]. Although chemotaxis has been observed in many bacterial species, *Escherichia coli* is one of the mostly studied examples. *E. coli* respond to the chemical stimulus by alternating the rotational direction of their flagella [1], [2].

Various mathematical models on the basis of Patlak-Keller-Segel model have been successfully used as important tools to study the mechanisms of chemotaxis [3]. A comprehensive review on the mathematical modeling of chemotaxis has been presented by Hillen and Painter [4].

Bacterial species including *E. coli* have been observed to form various patterns under various environmental conditions [5], [6], [7]. Populations of bacteria are capable of self-organization into states exhibiting strong inhomogeneities in density [8]. Recently, the spatiotemporal patterns in the fluid cultures of *E. coli* have been observed by employing lux-gene engineered cells and a bioluminescence imaging technique [9], [10]. However, the mechanisms governing the formation of bioluminescence patterns still remain unclear.

Over the last two decades, lux-gene engineered bacteria have been successfully used to develop whole cell-based biosensors [11]. A whole-cell biosensor is an analyte probe consisting of a biological element, such as a genetically engineered bacteria, integrated with an electronic component to yield a measurable signal. Whole-cell biosensors have been successfully used for the detection of environmental pollutant bioavailability, various stressors, including dioxins, endocrine-disrupting chemicals, and ionizing radiation [12]. To solve the problems currently limiting the practical use of whole-cell biosensors, the bacterial self-organization within the biosensors have to be comprehensively investigated.

This paper investigates the bacterial self-organization in a small circular container near the three phase contact line as detected by quasi-one-dimensional bioluminescence imaging. The aim of this work was to develop a computational model for simulating the spatiotemporal pattern formation of bioluminescence in the fluid cultures of *E. coli* [9], [10], [13]. The pattern formation in a luminous *E. coli* colony was modeled by the nonlinear reaction-diffusion-chemotaxis equations assuming two kinds of the chemotactic sensitivity, the signal-dependent sensitivity and the density-dependent sensitivity. The model was formulated on a one-dimensional domain. The numerical simulation at transition conditions was carried out using the finite difference technique [14]. The computational model was validated by experimental data. By varying the input parameters the output results were analyzed with a special emphasis on the influence of the chemotactic sensitivity on the spatiotemporal pattern formation in the luminous *E. coli* colony. Regular oscillations as well as chaotic fluctuations similar to experimental ones were computationally simulated.

The rest of the paper is organized as follows. In Section II, the mathematical model is described. Section III discusses the computational modeling of a physical experiment. Section IV is devoted to present results of the numerical simulation. Finally, the main conclusions are summarized in Section V.

II. Mathematical Modeling

Various mathematical models based of advection-reaction-diffusion equations have been developed for modeling of pattern formation in bacterial colonies [5], [6], [15], [16], [17]. The system of coupled partial differential equations introduced by Keller and Segel are among the most widely used [3], [4].

A. Governing Equations

According to the Keller and Segel approach, the main biological processes can be described by a system of two conservation equations ($x \in \Omega,\ t > 0$),

$$\begin{aligned} \frac{\partial n}{\partial t} &= \nabla\left(D_n \nabla n - h(n,c) n \nabla c\right) + f(n,c), \\ \frac{\partial c}{\partial t} &= \nabla\left(D_c \nabla c\right) + g_p(n,c) n - g_d(n,c) c, \end{aligned} \tag{1}$$

where x and t stand for space and time, $n(x,t)$ is the cell density, $c(x,t)$ is the chemoattractant concentration, D_n and D_c are the diffusion coefficients usually assumed to be constant, $f(n,c)$ stands for cell growth and death, $h(n,c)$ stands the chemotactic sensitivity, g_p and g_d describe the production and degradation of the chemoattractant [3], [17].

The cell growth $f(n,c)$ is usually assumed to be logistic function, i.e., $f(n,c) = k_1 n(1 - n/n_0)$, where k_1 is the constant growth rate of the cell population, and n_0 is the ”carrying capacity” of the cell population [5].

A number of chemoattractant production functions have been employed in chemotactic models [4]. Usually, a saturating function of the cell density is used indicating that, as the cell density increases, the chemoattractant production decreases. The Michaelis-Menten function is widely used to express the production rate, $g_p(n,c) = k_2/(k_3 + n)$ [3], [13], [16], [18]. The degradation or consumption of the chemoattractant is typically constant, $g_d(n,c) = k_4$. Values of k_2, k_3 and k_4 are not exactly known [17].

The function $h(n,c)$ stands for the chemotactic sensitivity. The signal-dependent sensitivity and the density-dependent sensitivity are two main kinds of the chemotactic sensitivity [4]. Two commonly used forms of the signal-dependent sensitivity function $h(n,c)$ are the ”receptor” ($h(n,c) = k_5/(k_6 + c)^2$) and the ”logistic” ($h(n,c) = k_5/(k_6 + c)$) forms [4], [15], [17]. Assuming that cells carry a certain finite volume, a density-dependent chemotactic sensitivity function as well as volume-filling model were derived, $h(n,c) = k_5(1 - n/n_0)$, where n_0 denotes the maximal cell density [4]. Another form for the density-dependent chemotactic sensitivity ($h(n,c) = k_5/(k_6 + n)$) has been introduced by Velazquez [19].

In the simplest form, the chemotactic sensitivity is assumed to be independent of the chemoattractant concentration c as well as the cell density n, i.e., $h(n,c)$ is constant, $h(n,c) = k_5$. Since the proper form of the chemotactic sensitivity function $h(n,c)$ to be used for the simulation of the spatiotemporal pattern formation in the fluid cultures of lux-gene engineered *E. coli* is unknown, all these four forms of $h(n,c)$ were used to find out the most useful form.

When modeling the bacterial self-organization in a circular container along the contact line [9], [10], [13], the mathematical model can be defined on a one dimensional domain - the circumference of the vessel. Replacing f, g_p and g_d with the concrete expressions above, the governing equations (1) reduce to a cell kinetics model with nonlinear signal kinetics as well as the chemotactic sensitivity,

$$\begin{aligned} \frac{\partial n}{\partial t} &= D_n \Delta n - \nabla\left(h(n,c) n \nabla c\right) + k_1 n \left(1 - \frac{n}{n_0}\right), \\ \frac{\partial c}{\partial t} &= D_c \Delta c + \frac{k_2 n}{k_3 + n} - k_4 c, \quad x \in (0,l), \quad t > 0, \end{aligned} \tag{2}$$

where Δ is the Laplace operator formulated in the one-dimensional Cartesian coordinate system, and l is the length of the contact line, i.e., the circumference of the vessel. Assuming R as the vessel radius, $l = 2\pi R$, $x \in (0, 2\pi R)$.

B. Initial and Boundary Conditions

A non-uniform initial distribution of cells and zero concentration of the chemoattractant are assumed,

$$n(x,0) = n_{0x}(x), \quad c(x,0) = 0, \quad x \in [0,l], \tag{3}$$

where $n_{0x}(x)$ stands for the initial ($t = 0$) cell density.

For the bacterial simulation on a continuous circle of the length l of the circumference, the matching conditions are applied ($t > 0$):

$$\begin{aligned} & n(0,t) = n(l,t), \qquad c(0,t) = c(l,t), \\ & \frac{\partial n}{\partial x}\Big|_{x=0} = \frac{\partial n}{\partial x}\Big|_{x=l}, \qquad \frac{\partial c}{\partial x}\Big|_{x=0} = \frac{\partial c}{\partial x}\Big|_{x=l}. \end{aligned} \tag{4}$$

C. Dimensionless Model

In order to define the main governing parameters of the mathematical model (2)-(4) [4], [7], [18], a dimensionless mathematical model has been derived by setting

$$\begin{aligned} & u = \frac{n}{n_0}, \quad v = \frac{k_3 k_4 c}{k_2 n_0}, \quad t^* = \frac{k_4 t}{s}, \quad x^* = \sqrt{\frac{k_4}{D_c s}} x, \\ & D = \frac{D_n}{D_c}, \quad r = \frac{k_1}{k_4}, \quad \phi = \frac{n_0}{k_3}, \\ & \chi(u,v) = \frac{k_2 n_0}{k_3 k_4 D_c} h(n_0 u, k_2 n_0 c/(k_3 k_4)). \end{aligned} \tag{5}$$

Dropping the asterisks, the dimensionless governing equations then become ($t > 0$)

$$\begin{aligned} \frac{\partial u}{\partial t} &= D \frac{\partial^2 u}{\partial x^2} - \frac{\partial}{\partial x}\left(\chi(u,v) u \frac{\partial v}{\partial x}\right) + s r u (1 - u), \\ \frac{\partial v}{\partial t} &= \frac{\partial^2 v}{\partial x^2} + s\left(\frac{u}{1 + \phi u} - v\right), \quad x \in (0,1), \end{aligned} \tag{6}$$

where x and t stand for the dimensionless space and time, respectively, u is the dimensionless cell density, v is the dimensionless chemoattractant concentration, r is the dimensionless growth rate of the cell population, ϕ stands for saturating of the signal production, $\chi(u, v)$ is the dimensionless chemotactic sensitivity, and s stands for the spatial and temporal scale.

For the dimensionless simulation of the spatiotemporal pattern formation in a luminous *E. coli* colony, four forms of the chemotactic sensitivity function $\chi(u, v)$ were used to find out the best fitting pattern for the experimental data [9], [10], [13],

$$\chi(u,v) = \frac{\chi_0}{(1+\alpha v)^2}, \tag{7a}$$

$$\chi(u,v) = \chi_0 \frac{1+\beta}{v+\beta}, \tag{7b}$$

$$\chi(u,v) = \chi_0 \left(1 - \frac{u}{\gamma}\right), \tag{7c}$$

$$\chi(u,v) = \frac{\chi_0}{1+\epsilon u}. \tag{7d}$$

The first two forms (7a) and (7b) of the function $\chi(u, v)$ correspond to the signal-dependent sensitivity, while the other two (7c) and (7d) - for the density-dependent sensitivity [4]. Accepting $\alpha = 0$, $\beta \to \infty$, $\gamma \to \infty$ or $\epsilon = 0$ leads to a constant form of the chemotactic sensitivity, $\chi(u, v) = \chi_0$.

The initial conditions (3) take the following dimensionless form:

$$u(x, 0) = 1 + \varepsilon(x), \quad v(x, 0) = 0, \quad x \in [0, 1], \tag{8}$$

where $\varepsilon(x)$ was a 20% random uniform spatial perturbation.

The boundary conditions (4) transform to the following dimensionless equations ($t > 0$):

$$\begin{aligned} &u(0,t) = u(1,t), \qquad v(0,t) = c(1,t), \\ &\frac{\partial u}{\partial x}\bigg|_{x=0} = \frac{\partial u}{\partial x}\bigg|_{x=1}, \quad \frac{\partial v}{\partial x}\bigg|_{x=0} = \frac{\partial v}{\partial x}\bigg|_{x=1}. \end{aligned} \tag{9}$$

According to the classification of chemotaxis models, the dimensionless model of the pattern formation is a combination of the signal-dependent sensitivity (M2), the density-dependent sensitivity (M3), the saturating signal production (M6) and the cell kinetics (M8) models [4].

III. Numerical Simulation

The mathematical model (2)-(4), as well as the corresponding dimensionless model (6), (8), (9), has been defined as an initial boundary value problem based on a system of nonlinear partial differential equations. No analytical solution is possible because of the nonlinearity of the governing equations of the model [7]. Hence the bacterial self-organization was simulated numerically.

The numerical simulation was carried out using the finite difference technique [14]. To find a numerical solution of the problem a uniform discrete grid with 200 points and the dimensionless step size 0.005 (dimensionless units) in the space direction was introduced, $250 \times 0.004 = 1$. A constant dimensionless step size 10^{-6} was also used in the time direction. An explicit finite difference scheme has been built as a result of the difference approximation [14], [20]. The digital simulator has been programmed by the author in JAVA language [21].

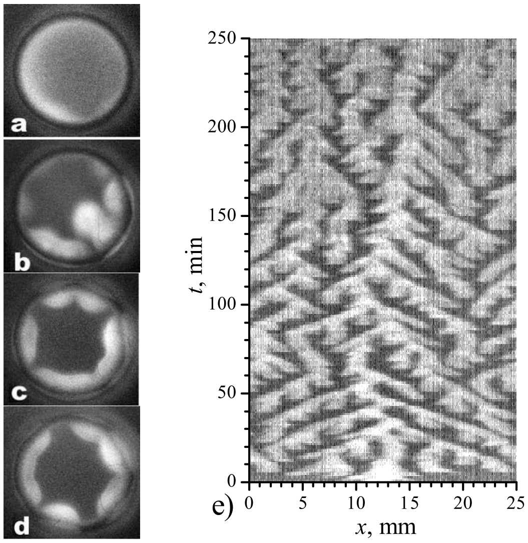


Figure 1. Top view bioluminescence images of the bacterial cultures in the cylindrical vessel at 5 (a), 20 (b), 40 (c), 60 (d) min and space-time plot along the contact line (e) [10].

The computational model was applied to the simulation of bioluminescence patterns observed in a small circular containers made of glass [10], [13]. Figures 1a-1d show typical top view bioluminescence images of bacterial cultures illustrating an accumulation of luminous bacteria near the contact line. In general, the dynamic processes in unstirred cultures are rather complicated and need to be modeled in three dimensional space [1], [9], [10]. Since luminous cells concentrate near the contact line, the three-dimensional processes were simulated in one dimension (quasi-one dimensional rings in Figures 1a-1d). Figure 1e shows the corresponding space-time plot of quasi-one-dimensional bioluminescence intensity.

By varying the model parameters the simulation results were analyzed with a special emphasis to achieving a spatiotemporal pattern similar to the experimentally obtained pattern shown in Figure 1e. Figure 2 shows the results of the informal pattern fitting, where Figures 2a and 2b present simulated space-time plots of the dimensionless cell density u and the chemoattractant concentration v, respectively. The corresponding values $\bar{u}$ and $\bar{v}$ averaged on circumference of the vessel are depicted in Figure 2c. Regular oscillations as well as chaotic fluctuations similar to experimental ones were computationally simulated. Accepting the constant form of the chemotactic sensitivity, $\chi(u, v) = \chi_0$, the dynamics of the bacterial population was simulated at the

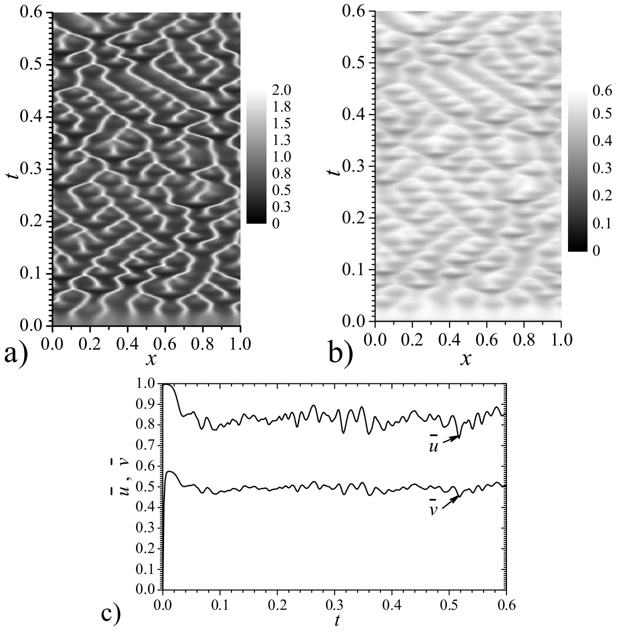


Figure 2. Simulated space-time plots of the dimensionless cell density u (a) as well as the chemoattractant concentration v (b) and the corresponding averaged values $\bar{u}$ and $\bar{v}$ (c). Values of the parameters are as defined in (10).

following values of the model parameters [13]:

$$D = 0.1,\ \chi_0 = 6.2,\ r = 1,\ \phi = 0.73,\ s = 625\,. \qquad (10)$$

Due to a relatively great number of model parameters, there is no guarantee that the values (10) mostly approach the pattern shown in Figure 1e. Similar patterns were achieved at different values of the model parameters. An increase in one parameter can be often compensated by decreasing or increasing another one [4], [17], [22] .

IV. Results and Discussion

By varying the input parameters the output results were analyzed with a special emphasis on the influence of the chemotactic sensitivity on the spatiotemporal pattern formation in the luminous *E. coli* colony. Figure 2a shows the spatiotemporal pattern for the constant form of the chemotactic sensitivity, $\chi(u, v) = \chi_0$.

A. The Effect of the Signal-Dependent Sensitivity

The signal-dependent sensitivity was modeled by two forms of the chemotactic sensitivity function χ: (7a) and (7b). The spatiotemporal patterns of the dimensionless cell density u were simulated at very different values of α and β. Figure 3 shows signal-dependency of the chemotactic sensitivity.

Accepting $\alpha = 0$ or $\beta \to \infty$ leads to a constant form of the chemotactic sensitivity, $\chi(u, v) = \chi_0$. Results of multiple simulations showed that the simulated patterns distinguish from the experimental ones (Figure 1e) when increasing α-parameter (Figure 3a) or decreasing β-parameter (Figure 3b). Because of this, there is no practical reason for application of a non-constant form of the signal-dependent sensitivity to modeling the formation of the bioluminescence patterns in a colony of luminous *E. coli.*

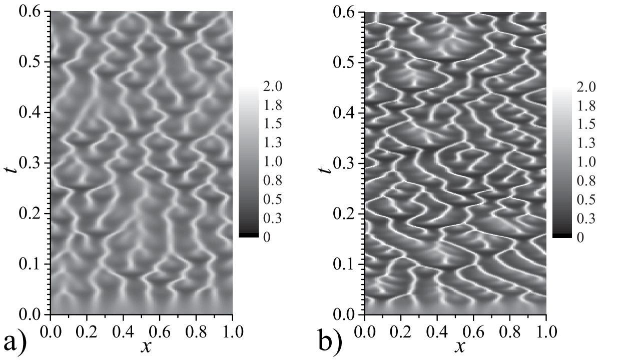


Figure 3. Spatiotemporal plots of the dimensionless cell density u for two forms of the signal-dependent chemotactic sensitivity $\chi(u, v)$: (7a) (α = 0.05) (a) and (7b) (β = 10) (b). Values of other parameters are as defined in (10).

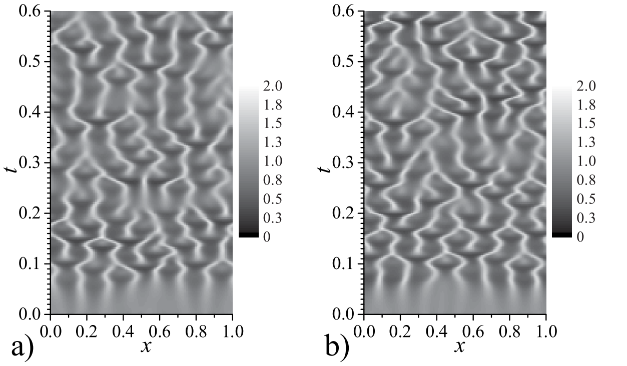


Figure 4. Spatiotemporal plots of the dimensionless cell density u for two forms of the density-dependent chemotactic sensitivity $\chi(u, v)$: (7c) (γ = 10) (a) and (7d) (ϵ = 0.1) (b). Values of other parameters are as defined in (10).

B. The Effect of the Density-Dependent Sensitivity

Two forms, (7c) and (7d), of the function χ were employed for modeling the density-dependent chemotactic sensitivity. The spatiotemporal patterns of the cell density u were simulated at various values of γ and ϵ. Figure 4 shows how the density-dependency affects the pattern formation.

Accepting $\gamma \to \infty$ or $\gamma = 0$ leads to a constant form of the chemotactic sensitivity, $\chi(u, v) = \chi_0$. Multiple simulation showed that the simulated patterns distinguish from the experimental ones (Figure 1e) when decreasing γ-parameter (Figure 4a) or increasing ϵ-parameter (Figure 4b). Because of this, similarly to the signal-dependent chemotactic sensitivity, there is no practical reason for application of a non-constant form also of the density-dependent sensitivity when modeling the pattern formation in a colony of luminous *E. coli.*

A simple constant form ($\chi(u, v) = \chi_0$) of the chemotactic sensitivity can be successfully applied to modeling the formation of the bioluminescence patterns in a colony of luminous *E. coli.* Oscillations and fluctuations similar to experimental ones can be computationally simulated ignoring the signal-dependence as well as the density-dependence of the chemotactic sensitivity.

V. Conclusions

The quasi-one dimensional spatiotemporal pattern formation along the three phase contact line in the fluid cultures of lux-gene engineered *Escherichia coli* can be simulated and studied on the basis of the Patlak-Keller-Segel model.

The mathematical model (2)-(4) and the corresponding dimensionless model (6), (8), (9) of the bacterial self-organization in a circular container as detected by bioluminescence imaging may be successfully used to investigate the pattern formation in a colony of luminous *E. coli.*

A constant function ($\chi(u, v)$ as well as $h(n, c)$) of the chemotactic sensitivity can be used for modeling the formation of the bioluminescence patterns in a colony of luminous *E. coli.* Oscillations and fluctuations similar to experimental ones can be computationally simulated ignoring the signal-dependence as well as the density-dependence of the chemotactic sensitivity.

The more precise and sophisticated two- and three-dimensional computational models implying the formation of structures observed on bioluminescence images are now under development.

Acknowledgment

This research was funded by the European Social Fund under Measure VP1-3.1-ŠMM-07-K ”Support to Research of Scientists and Other Researchers (Global Grant)”, Project ”Developing computational techniques, algorithms and tools for efficient simulation and optimization of biosensors of complex geometry”.

The author is grateful to Dr. R. Šimkus for the experimental data and valuable discussions.

References

[1] M. Eisenbach, *Chemotaxis*. London: Imperial College Press, 2004.

[2] T. C. Williams, *Chemotaxis: Types, Clinical Significance, and Mathematical Models*. New York: Nova Science, 2011.

[3] E. F. Keller and L. A. Segel, “Model for chemotaxis,” *J. Theor. Biol.*, vol. 30, no. 2, pp. 225–234, 1971.

[4] T. Hillen and K. J. Painter, “A users guide to pde models for chemotaxis,” *J. Math. Biol.*, vol. 58, no. 1-2, pp. 183–217, 2009.

[5] E. O. Budrene and H. C. Berg, “Dynamics of formation of symmetrical patterns by chemotactic bacteria,” *Nature*, vol. 376, no. 6535, pp. 49–53, 1995.

[6] M. P. Brenner, L. S. Levitov, and E. O. Budrene, “Physical mechanisms for chemotactic pattern formation by bacteria,” *Biophys. J.*, vol. 74, no. 4, pp. 1677–1693, 1998.

[7] J. D. Murray, *Mathematical Biology: II. Spatial Models and Biomedical Applications*, 3rd ed. Berlin: Springer, 2003.

[8] S. Sasaki *et al.*, “Spatio-temporal control of bacterial-suspension luminescence using a pdms cell,” *J. Chem. Engineer. Japan*, vol. 43, no. 11, pp. 960–965, 2010.

[9] R. Šimkus, “Bioluminescent monitoring of turbulent bioconvection,” *Luminescence*, vol. 21, no. 2, pp. 77–80, 2006.

[10] R. Šimkus, V. Kirejev, R. Meškienė, and R. Meškys, “Torus generated by *Escherichia coli*,” *Exp. Fluids*, vol. 46, no. 2, pp. 365–369, 2009.

[11] S. Daunert *et al.*, “Genetically engineered whole-cell sensing systems: coupling biological recognition with reporter genes,” *Chem. Rev.*, vol. 100, no. 7, pp. 2705–2738, 2000.

[12] R. J. M. M. B. Gu and B. C. Kim, “Whole-cell-based biosensors for environmental biomonitoring and application,” *Adv. Biochem. Eng. Biotechnol.*, vol. 87, pp. 269–305, 2004.

[13] R. Šimkus and R. Baronas, “Metabolic self-organization of bioluminescent *Escherichia coli*,” *Luminescence*, DOI 10.1002/bio.1303. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/bio.1303/full (Accessed Aug. 27, 2011).

[14] A. A. Samarskii, *The Theory of Difference Schemes*. New York-Basel: Marcel Dekker, 2001.

[15] I. R. Lapidus and R. Schiller, “Model for the chemotactic response of a bacterial population,” *Biophys. J.*, vol. 16, no. 7, pp. 779–789, 1976.

[16] P. K. Maini, M. R. Myerscough, K. H. Winters, and J. D. Murray, “Bifurcating spatially heterogeneous solutions in a chemotaxis model for biological pattern generation,” *Bull. Math. Biol.*, vol. 53, no. 5, pp. 701–719, 1991.

[17] R. Tyson, S. R. Lubkin, and J. D. Murray, “Model and analysis of chemotactic bacterial patterns in a liquid medium,” *J. Math. Biol.*, vol. 38, no. 4, pp. 359–375, 1999.

[18] M. R. Myerscough, P. K. Maini, and K. J. Painter, “Pattern formation in a generalized chemotactic model,” *Bull. Math. Biol.*, vol. 60, no. 1, pp. 1–26, 1998.

[19] J. J. L. Velazquez, “Point dynamics for a singular limit of the keller-segel model. i. motion of the concentration regions,” *SIAM J. Appl. Math.*, vol. 64, no. 4, pp. 1198–1223, 2004.

[20] R. Baronas, F. Ivanauskas, and J. Kulys, *Mathematical Modeling of Biosensors: An Introduction for Chemists and Mathematicians*, ser. Springer Series on Chemical Sensors and Biosensors, G. Urban, Ed. Dordrecht: Springer, 2010.

[21] J. E. Moreira *et al.*, “Java programming for high-performance numerical computing,” *IBM Syst. J.*, no. 6, pp. 21–56, 2000.

[22] K. J. Painter and T. Hillen, “Spatio-temporal chaos in a chemotactic model,” *Physica D*, vol. 240, no. 4-5, pp. 363–375, 2011.

Applying Simulation and Mathematical Programming on a Business Case Analysis for Setting up a Spare Part Logistics in the Construction Supply Industry

A Case Study

Wilhelm Dangelmaier
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany
whd@hni.upb.de

Christoph Laroque
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany
christoph.laroque@hni.upb.de

Robin Delius
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany
robin.delius@hni.upb.de

Jenny Streichhan
University of Paderborn
Paderborn, Germany
jenny.streichhan@hni.upb.de

Abstract— **This paper describes how methods and techniques from different fields of research can be combined to evaluate cost-intensive and business-critical decisions regarding future market development. In their concrete application a leading company from the construction supply industry has to make a decision on setting up of a spare part logistic. Three future alternatives (negative, constant, positive market growth) on market trends are simulated with a Monte Carlo simulation by considering a given demand history and possible locations for storage facilities were isolated by applying the Steiner-Weber method. Finally solving a mixed-integer formulation of the Uncapacitated-Facility-Location-Problem gives information on opening/closing new/existing storage facilities by minimizing all relevant costs. The results of this approach, containing information about a cost-based evaluation of all business related decision criteria, were examined with a sensitivity analysis.**

Keywords: spare parts logistics, Monte Carlo simulation, Steiner Weber iteration, mixed-integer programming, Uncapacitated Facility Location Problem, sensitivity analyses

I. Introduction

With the help of simulation, this case study analyzes the options for a German construction supplier to set up a spare part logistic. The company had an enormous growth during the last years - especially regarding the distribution of solar systems. Since these systems support the heating circuit of the ultimate customers, the company is in duty for delivering spare parts within a short time. Up to this date the company was able to serve these requests with the help of the general stocks but due to the enormous growth new options need to be developed. The first idea was to delegate this task to a service provider and some offers were invited. A comparison showed that the best offer was 460€ for delivering a single spare part to the customer. These high costs are the main reason for developing an own strategy with own storage facilities.

A possible scenario was developed to cover all requirements. The first step was done by analyzing the demand history of the last year. Due to the fact that these won't reflect the future development correctly, new market demands were created by using simulation. The second requirement within this scenario is the concrete location decision, based on geographic features. The last part of the basic consideration is the facility location problem, which solves the problem on making a cost-optimal decision regarding facility opening. The first assumptions were solved with the help of mathematical programming.

In order to include the unpredictable requirements of the market, a sensitivity analysis was applied. The scenario has been modified towards a positive and negative price trend of the element of costs for different market growth options.

II. Scenario Analysis

The present business case contains a lot of requirements which need to be considered. This section will define the restrictions which cause the analytic procedure.

A. The Market

The solar market experienced a strong growth during the last years. One reason is that the German government subsidized the setting up of solar systems until 2011. This discontinuation makes the market unpredictable for a broad consideration. Another aspect is that the demand is fluctuant during the year. There are peaks with 20% of the yearly demand and troughs with 2% of the yearly demand. To cover all of these deficits the demand has been forecasted for three different cases. The first case assumes a negative market growth with 70% of the former demand. This is named as the worst case. The next assumption covers up a faltering market growth by using 100% of the former demand. This approach comprises the average (avg) case. A positive market growth is coped by calculating with 130% of the former demand and is called the best case. These three scenarios should meet the requirements of the market.

B. Conditions of supply

Due to the fact that the spare parts need to be delivered within 12 hours, there are challenging requirements towards

ISBN: 978-1-61208-169-4

the transportation. This accomplishment could be ensured by using a courier service during the workday (69%) and a cap during the weekend and national German holidays (31%). Therefore the transportation costs are higher than using a shipping company. The used price for a kilometer is 1.00€. As a vendor of systems for using renewal energy the company focuses on reducing the CO2 emission for the transportation. This is one of the reasons why this work considers spread storage.

C. *Range of spare parts*

For using new storage capacities the space requirements need to be analyzed. The former sales are used for a forecasting. There are 24 articles within the assortment. The smallest one measures 17cm x 17cm x 1cm (length x width x height). The biggest one measures 40cm x 43cm x 7cm. All articles have the size of standardizes parcels. There are no special needs for the transportation. Due to the fact that even articles with a low sales figure need to be stored in each location, the spacial amount will be higher than with the use of one location. The needed storage area of the last year sales would be 358,2m². Regarding to this fact, 442,8m² would be needed for the spread storage.

In order to classify the articles, an ABC-Analysis has been done. One of those spare parts could be classified with A and two of them as B. The other 21 parts are within the range of C parts.

D. *Existing depots*

The company owns three storage facilities. One of them is located in the northwest of Germany at the postal-zip-code area beginning with the number three. The other two are located in the southwest very close to each other. Their postal-zip-code starts with the number six.

Those warehouses should be involved within this work as potential spare part storage locations. Due to the fact, that they have to be considered as privileged, the costs are termed as zero, because they exist even though they are not in use as a spare part storage facility.

III. APPROACH

This section will introduce the used approach to solve each of the tasks. At first the simulation of future demands has to b done. After that, the location decision is explained. In the end the uncapacitated facility location problem will be introduced which is formulated with a mixed-integer program (MIP).

A. *Simulation of the demand*

As described in section two, the forecast of demands is not easy. Simulation became a favored and important method for solving problems within the field of production and logistics [1]. That is the reason for using a simulative method for this fraught decision with risk. This approach sets up the different scenarios for the market growth and the possible consequences could be deduced.

Our decision went to the Monte-Carlo simulation as an approved method of choice for the task to generate randomized demands within a given planning horizon. This approach uses the aspects that all results have the same chance and are independent from each other [2].

Because of the named characteristics this method is used for risk analysis very often. It is in use for complex processes which could not be solved directly. The unsteady demand could be simulated through this method, which is presented by a normal curve of distribution with the range of 70% till 130% of the former demand. To cover up a negative market growth the simulated demands were multiplied by 70%. The same was done for the positive market growth of 130%

The five-figure postal-zip-code system is used to separate different areas of Germany. The first number of the code goes from 0 to 9. Therefore 10 markets could be pointed out. Fig. 1 shows the demand of the year 2010 separated for the postal-zip-code areas.

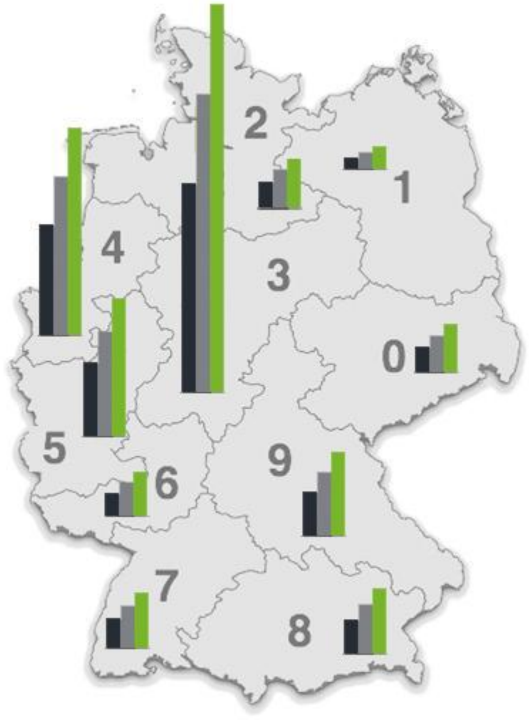


Figure 1. Separated demands (70%, 100% and 130%) by postal-zip-code

It is obvious that the demand of the market 3 is much higher than the demand at the area number 1. The different density of population is one of the reasons. The area one has 71-87 inhabitants per km² and the area 3 has 524 inhabitants per km². However the customers within Germany need to be delivered with taking into account the transportation restriction.

These demands are the basic approach for the normally distributed randomized demand within the given range. The numbers were computed with Minitab for each area.

TABLE I. Interval [Min; Max] of the Monte-Carlo Simulation results

	Worst-case	AVG-case	Best-case
Market 0	[130; 187]	[173; 255]	[247; 334]
Market 1	[59; 81]	[85; 115]	[102; 168]
Market 2	[115; 179]	[155; 270]	[206; 322]
Market 3	[987; 1394]	[1396; 2005]	[1639; 2601]
Market 4	[532; 798]	[720; 1110]	[850; 1333]
Market 5	[360; 545]	[443; 798]	[644; 899]
Market 6	[111; 169]	[162; 238]	[197; 323]
Market 7	[135; 218]	[200; 321]	[250; 382]
Market 8	[163; 250]	[231; 349]	[334; 447]
Market 9	[203; 333]	[296; 433]	[338; 595]

Table 1 gives a short overview about the results of the Monte-Carlo Simulation of the demand within a min./max.-interval out of the result series. These results are used for the third step where mathematical programming has to be applied but at first the decision on facility location has to be made. Its result will be used within the mixed-integer program as well.

B. Location decision

After determining the demand, it is necessary to consider the location decision problem. The separation of 10 areas leads to the idea to set up a single storage facility per market. A traditional method for solving this task is the Steiner-Weber location approach [3]. This technique is based on three assumptions:

- n customer with j= 1, …, n are supplied with a homogenous area. The position of the customer j is located at coordinates (u_j, v_j) with a demand of b_j.
- each position is a potential location and
- the transportation costs $c_{i,j}$ between two positions are proportional towards the transported amount and distance. The costs are consistent for each unit. For the distance measurement the Euclidean meter will be used.

The goal of this method is to discover the position which ensures an inexpensive supply to all customers. This problem could be formulated as minimizing problem:

$$K(x_s, y_s) = c \cdot \sum_{j=1}^{n} b_j \cdot d_j \tag{1}$$

$$K(x_s, y_s) = c \cdot \sum_{j=1}^{n} b_j \cdot \sqrt{\left(x_s - u_j\right)^2 + \left(y_s - v_j\right)^2} \tag{2}$$

One way to solve this equation is to split up the process into two steps. The first step is to calculate the weighted focus of all customer positions and their demands. This result will be improved with an approximate procedure until the result is almost optimal. This process is called the scheme of Miehle [6].

In order to use this method, the markets have been separated by the second number of their postal-zip-code. The focuses of these areas are used as the demand spots. Each location sums up all demands within this area

This method results in having 10 potential locations for setting up spare part storage locations as shown in Fig. 2. These locations are the basic approach for the next step during the whole analysis.

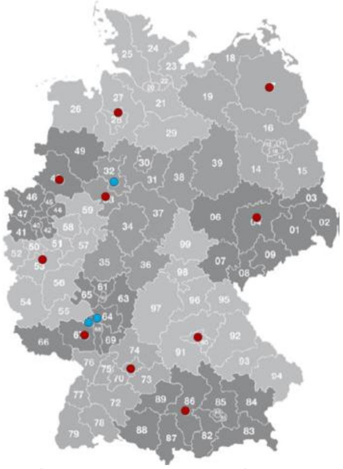


Figure 2. Results of the location decision after iterations of the Steiner-Weber method

C. Facility Location Problem

After nominating locations for new facilities, these have to be analyzed regarding the costs. Each of them causes costs for being operated and on the other hand reduces transportation costs to the customer due to shorter delivery routes. This class of problems is called facility location problem and covers all problems were a selection of different locations has to be made. The proceeding is based on graphs where the nodes are representing the locations and the edges are representing the distance between them. All nodes have a certain demand. These models are named as finitely discrete. Furthermore, a differentiation regarding the overall goal could be done. The minmax locations problem tries to minimize the largest distances between the locations. The other type is the minsum location problem and tries to minimize the sum of all distances [7]. A special model of this class is the Uncapacitated-Facility-Location-Problem (UFLP). This problem is also called Simple-Plant-Location- or Uncapacitated-Warehouse-Location-Problem. As given in the name there are no restrictions regarding the capacities since all location are new as in this present business case. The model contains the following parameters:

- n locations denoted as S_i with i=1, …, n,
- m customer denoted as K_j with j=1, …, m with a demand of D_j,
- fix costs F_i for opening up a location
- the transportation costs are between two locations with c_{ij}.

The objective function is:

$$\min Z = \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} \cdot z_{ij} + \sum_{i=1}^{n} F_i \cdot y_i \tag{3}$$

Under the constraints:

$$\sum_{i=1}^{n} z_{ij} = D_j \qquad j = 1, \dots, m \qquad (4)$$

$$z_{ij} \leq y_i D_j \qquad i = 1, \dots, n, \;\; j = 1, \dots, m \qquad (5)$$

$$z_{ij} \geq 0 \qquad i = 1, \dots, n, \;\; j = 1, \dots, \mathrm{m} \qquad (6)$$

$$y_i \in \{0, 1\} \qquad i = 1, \dots, n \qquad (7)$$

This model contains two decision variables within the constraints. One of them is y_i. It is a binary variable to decide either if a location is opened (y_i=1) or not (y_i=0). This is necessary because of the fixed cost for the facilities. The other variable is z_{ij} which stands for the available parts within this location for fulfilling the customer demands. The objective function minimizes the costs consisting of the sum of transportations cost in the first part and the cost for setting up the facility in the second part. These constrains make sure that all demands are covered (4) and that the demand is delivered by a location which exists (5).

The model was implemented with the tool IBM ILOG OPL Optimization studio in version 12. Due to the fact that this model does not consider variable costs the fixed costs include all fees like the rental fee for the storage space (different for each area), the picking fees (15€ per pick), the fix salary of the employee (3600€ per year) and costs for the material and equipment (1200€ per year). Those costs were calculated for the three scenarios with 70%, 100% and 130% of the former demand. The costs are shown in table 2.

TABLE II. Overview of the used fix costs

	Worst-case	AVG-case	Best-case
Market 0	9.280,58 €	10.275,12€	11.269,65€
Market 1	8.055,61 €	8.525,15€	8.994,70€
Market 2	9.540,94 €	10.647,06€	11.753,18€
Market 3	25.065,92 €	32.825,60€	40.585,28€
Market 4	16.667,09 €	20.827,27€	24.987,46€
Market 5	13.477,10 €	16.270,14€	19.063,18€
Market 6	9.084,49 €	9.994,98€	10.905,48€
Market 7	10.019,50 €	11.330,72€	12.641,93€
Market 8	10.204,99 €	11.595,70€	12.986,41€
Market 9	11.015,03 €	12.752,91€	14.490,78€

After analyzing the market, the occurred costs and computing the results, these outcomes need to be summed up and compared with the service provider.

IV. Results of the Basic Approach

After implementing the model and the data to the tool a clear result occurs. The results of the first assumption are explicit

A. *Average case results*

This case studied a continuous demand in referring to the former demand. For this case 30 different demands were proofed. The facilities of the markets 0, 1, 2, 3, 4, 5, 7, 8 and 9 were opened. In addition to this both existing facilities at the market 6 were used to deliver the spare parts. The facility of the market 3 was not in use. This result occurs for each demand. Summing these up costs 403.339,80€ per year could be expected.

B. *Worst case results*

An equal result as in the average case showed up in this case. The new facilities of all markets besides the market 6 are in use and in addition to this both existing facilities of the market 6. The location of the market 3 is not in use again. This effect was shown by all demands. The average cost of all iteration is 300.149,47€.

C. *Best case results*

For two demands out of this 30 series, another results showed up and new facility of the market 6 were opened up. The other new locations were opened too and the existing storage facilities were use too, besides of the market 3. The average cost are 502.542,73€ per year.

D. *Comparison to a service provider*

The most interesting investigation for the company within is the comparison of the costs referring to the fees of a service provider. This assumption has been done for the three market growth scenarios.

a) Average Case

The costs for delivering the spare parts to the facility differ about 69%. The company has to pay more because they have to deliver more facilities for storage. On the other hand the company does have less cost for delivering the parts to the customer. The company just would need pay 10,72% of the charged fees if they would do it by their self.

This could be shown summed up with the unit costs. One unit would cost 460,24€ with the service provider and 82,33€ with the introduced scenario.

TABLE III. Comparison of costs: service provider vs. the avg. case

	Service Provider	AVG-Case
Demand	4899 pcs.	4899 pcs.
Rental fee	1.201,46€	1.924,67€
Salary	152.284,65€	122.325,00€
Equipment	- €	10.800,00€
Transport to customer	2.097.969,81€	263.647,49€
Transport to facilities	3.257,50€	4.642,64€
Costs	2.254.713,41€	403.339,80€
Costs per piece	460,24€	82,33€

b) Worst Case

The first compared component is the delivery for the storage facilities. As shown before in this case the costs for the company are even higher too. They would pay 65% more. Even for this case the cost are higher if they would choose the service provider. The delivering could be realized for 15,31% of the offered price.

The unit cost are 460,74 € for the service provider and 87,52€ for the own concept. The reduction of costs per unit is 0,093% for the offered service and 6,45% for own facilities for considering the worst and the avg case.

TABLE IV. Comparison of costs: service provider and the worst case

	Service Provider	Worst-Case
Demand	3429,3 pcs.	3429,3 pcs.
Rental fee	841,02€	1.347,27€
Salary	107.679,26€	101.179,50€
Equipment	- €	10.800,00€
Transport to customer	1.468.578,86€	184.553,25€
Transport to facility	2.906,05€	2.269,45€
Costs	1.580.005,19€	300.149,47€
Costs per piece	460,74€	87,52€

c) Best Case

If the company would deliver the parts to the stocks by themselves they have to pay 64% more towards the offered service. But on the other hand they would save 89,28% by delivering the parts to the customers from the spread locations.

For the offered service a fee per unit would be 459,77€ and for the alternative 78,91€. Comparing this result to the average case it shows that the reduction of costs per unit would be 3,82% for the company option and 0,048% for the service provider.

TABLE V.Comparison of costs: service provider and the best-case

	Service Provider	Best-Case
Demand	6368,7 pcs.	6368,7pcs.
Rental fee	1.561,89€	2502,07€
Salary	196.890,05€	143.470,50€
Equipment	- €	10.800,00€
Transport to customer	2.727.360,75€	340.105,27€
Transport to facility	3.608,95€	5.664,89€
Costs	2.929.421,64€	502.542,73€
Costs per piece	459,97€	78,91€

Summarizing up the scenario is even more important if the company decides to enlarge the article range or if there would be a positive market growth.

V. SENSITIVITY ANALYSIS

In order to verify these results a sensitivity analysis was done for each scenario of market growth. In order to find out about the impact of the components within the fixed costs, they have been changed. As shown in table 6 the fees are studied with three different characteristics.

TABLE VI. Changed fees for the fixed costs

Fee			
Transportation costs	0,80 €	1,00 €	1,50 €
Fix salary	300,00 €	500,00 €	800,00 €
Pick fee	12,00 €	15,00 €	18,00 €

The new combinations of fixed costs have been implemented to the OPL model. For each case 10 different demand series were studied to show up the relationship between the elements. Another aspect which was observed is the rate of opening for each new facility.

The further proceeding is separated into two parts. The first one will consider the opening rate in correlation with costs. The results are represented in cubes for each scenario. The x-axis represents the costs per km, the y-axis shows the fix salary and the z-axis displays the picking fee. The average costs of the 10 series are presented by the number at each node.

A surprising aspect of the basic approach was that the new location of the market 3 was in use. The sensitivity analysis shows that this fact changes. The red line at the Fig. 3, 4 and 5 indicates that the normal stock location of 3 is in use, instead of the new one for more than 50% of the results. By comparing the three scenarios, it shows up that the surface at the dices is getting smaller with an increased market growth as shown up in Fig. 5. This led to the realization that if the market achieves a positive growth, the costs, especially the transportation costs, need to grow much more for using the basic stock location of the market 3 than it would be for the negative case.

Another aspect could be pointed out for the new facility at the market 5. The results for all market growth scenarios show a recommendation of 100% for most of the combinations. An exception occurs for all combinations with the transportation costs of 1,50€ per km. This new facility should be considered if the transportation costs are lower than this.

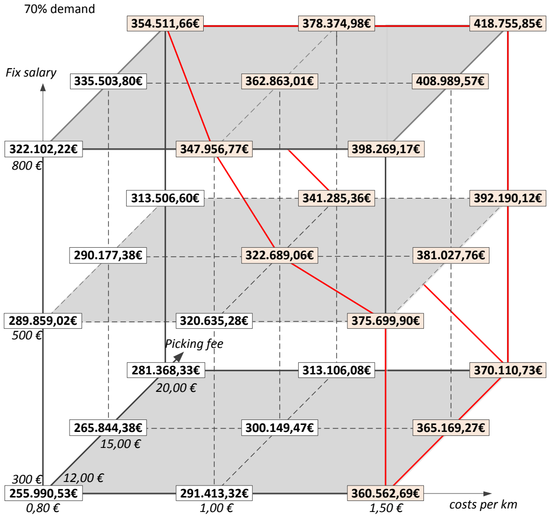


Figure 3. Cube illustrating the worst case results

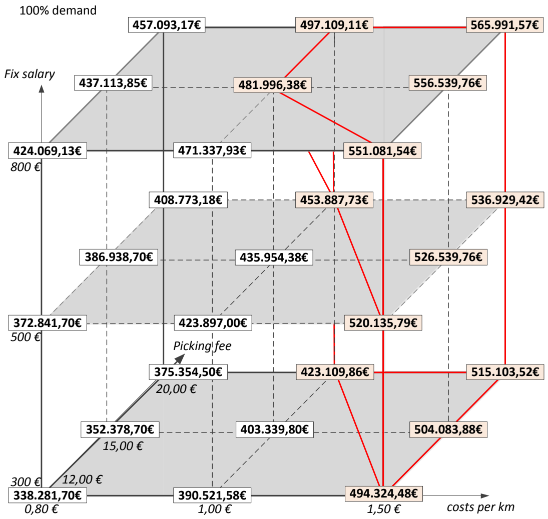


Figure 4. Cube illustrating the avg. case results

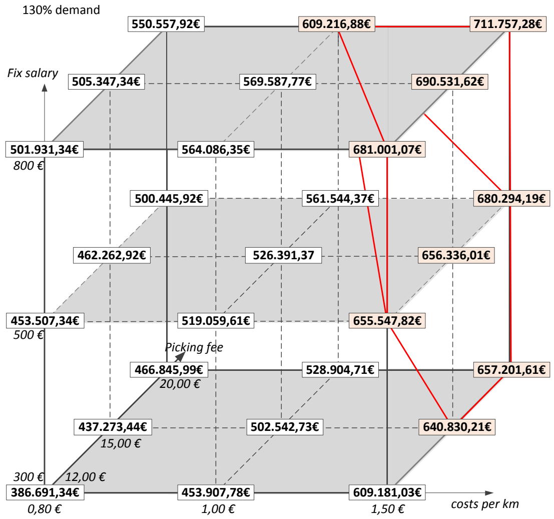


Figure 5. Cube illustrating the best case results

The new stock for the market 7 shows variations in recommendations too. Within the study of the lowest demand it shows up that this location was not in use for each combination of the transportation rate 1,50€ and for a fix salary 800€ and most of 500€. As for the market 3 with an increased demand this changes and more often this location is recommended. The average scenario shows that it would not be profitable to open up the stock for all combinations with a transportation rate of 1,50€, besides with a fix salary of 300€. The best case recommends this new facility not for all cases with a transportation rate of 1,50€. The costs per km are the most influential factor for this location. An equal observation could be made for the new facility of the market 8.

Moreover, the statistical analysis of the designed experiments allows the derivation of a mathematical meta-model for the resulting cost with a forecast quality of more than 98%. It's just based on the above mentioned input factors. Fig. 6 shows the relevance of each of the three input factors. It can be seen, that the fixed salary has the most significant influence on the resulting cost.

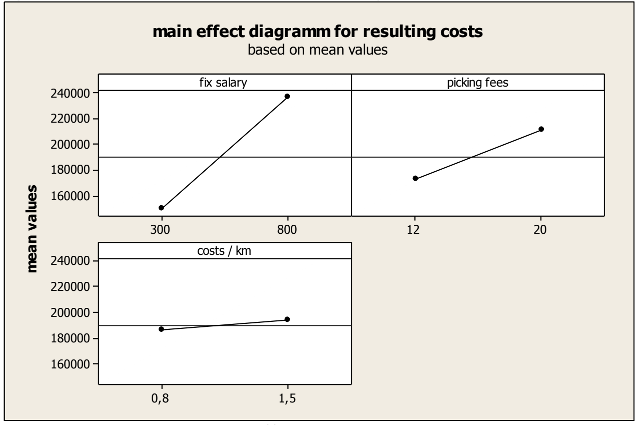


Figure 6. Main effect diagram on resulting costs

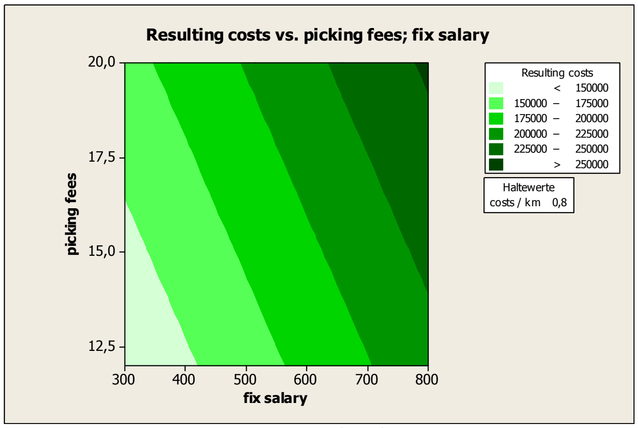


Figure 7. Response surface for resulting costs

Fig. 7 shows a derived response surface for the resulting costs of this model based on the derived meta-model. For each combination of fixed salary and picking fees, the resulting cost structure can be directly derived.

VI. Conclusion

This present business study shows how scientific methods can be applied to the real world and how they can support the decision making regarding real business problems. With this approach, using basic methods and implying real costs, we developed a framework which makes considering various alternatives possible. Furthermore the responsible management now has a basis for argumentation in cold print, since all parameters are given by the company and based on the company-wide costs.

For completeness of the results, the sensitivity analysis showed that the transportation cost and the fix salary have an enormous influence on the outcome calculated with the facility location problem. A possible recommendation for the company, based on the pure results could be to use new facilities within the markets 0, 1, 2, 4 and 9, as well as all basic stocks too, until the market growth could be estimated concretely. But the realistic orientation of these results needs to be considered too, since decisions within companies are made often in more difficult and elusive ways. Realistic options would probably be opening a single stock facility because of the unsafe market growth or maybe supply contracts with authorized premium dealer, which have available storage capacity and which are already familiar with the products.

Based on the more complex simulation and optimization models, a statistical meta-model could be derived via an experimental design study on the three input factors. It allows a forecast quality of the resulting cost structure of more than 98%, and thereby, can be used for further cost analysis, if some input factors change to a level, that has not been regarded during the here described study yet. Furthermore and referring to current discussions on sustainability and environmentally-conscious behavior, it could be of interest to investigate on the impact and influence of decision making towards reducing the emission of CO² in transport options.

References

[1] S. Terzi and S. Cavalieri "Simulation in the supply chain context: a survey." Computers in Industry, Volume 53, Issue 1, January 2004, pp. 3-16, ISSN 0166-3615, DOI: 10.1016/S0166-3615(03)00104-0

[2] M. H. Kalos and P. A. Whitlock "Monte Carlo methods" 2008 Wiley-VCH Verlag GmbH & Co. KGaA

[3] Z. Drezner, K. Klamroth, A. Schöbel, and G. O. Wesolowsky in Zvi Drezner "Facility location: applications and theory" Springer, pp.1-4, 2002

[4] W. Domschke und A. Scholl: „Grundlagen der Betriebswirtschaftslehre: Eine Einführung aus entscheidungsorientierter Sicht". ISBN 3-540-25047-6 3.Auflage, Springer, Berlin Heidelberg New York, 2005, pp. 173-174.

[5] W. Dangelmaier "Fertigungsplanung: Planung von Aufbau und Ablauf der Fertigung. Grundlagen, Algorithmen und Beispiele." 3rd Edition., Spinger, 2001., p. 153.

[6] C. Kriesel "Szenarioorientierte Unternehmensstruktur-optimierung. HNI-Verlagsschriftenreihe, Paderborn, 2006. p.32

[7] C. Royer „Simultane Optimierung von Produktions-standorten, Produktionsmengen und Distributions-gebieten. Herbert Utz Verlag, 2001. p.14

[8] J. Butler, J. Jia, and J. Dyer**,** Simulation techniques for the sensitivity analysis of multi-criteria decision models, European Journal of Operational Research Volume 103, Issue 3, 16 December 1997, pp. 531-546

Integrating Current State and Future State Value Stream Mapping with Discrete Event Simulation: A Lean Distribution Case Study

Amr Mahfouz, John Crowe and Amr Arisha
3S Group – College of Business
Dublin Institute of Technology (DIT)
Dublin 2, Ireland
E-mail: amr.mahfouz@dit.ie

Abstract: **In response to global recession and increased competition, organizations have tried to become more efficient by decreasing costs and streamlining operations. To achieve this, the philosophy of lean management has gained in popularity. The main obstacle organizations face when implementing lean is deciding which activities to implement lean principals on. A well known lean practice, value stream mapping, is a very effective tool in mapping the current and future state of an organizations lean activities. Limitations in calculating variability information that describe system variations and uncertainty means more powerful analytical tools are needed. Simulation offers a more thorough analysis of a system's data, including the examination of variability and has the ability to change certain parameters and measure key lean performance indicators. Using a tire distribution company as a case study, this paper has developed a framework that uses discrete event simulation as an integrative layer between current and future value stream mapping. The framework maps current state value and non-value activities in the company and through simulation has highlighted the activities that should be used when developing the future state map. This paper has highlighted simulation as a crucial middle layer in value stream mapping that will generate more accurate future state maps than the more common practices of using random estimates and experience alone.**

Keywords-Value stream mapping; distribution center; lean management; discrete event simulation

I. Introduction

The theory behind lean philosophy is to create more value with less. Over the last decade, competition between organizations has become a matter of not only productivity, but also of overall supply chain performance [1]. Delivering the right quantity of products to the right place and at the right time has become a necessity for supply chain survival in an ever-more-acutely competitive atmosphere [2]. The quest to offer high levels of service to customers, while keeping a worthwhile profit margin, has forced managers to think of new ways to eliminate waste from their internal operations. Lean thinking is one of the most effective techniques managers can use in this ambition.

The 'lean' strategy represents a holistic attack on all negative aspects of resource consumption, and seeks to achieve streamlined and waste-free operations [3]. While the focus of lean thinking literature has essentially been on production systems, the notion can also be stretched to cover every management activity. Recent research [4], [5] attention was directed into the use of simulation modeling in lean implementation and assessment processes due to many reasons including:

1. Identifying the factors and parameters involved in the manufacturing process.
2. Exploring the various opportunities of process improvement.
3. Predicting the impacts of the proposed changes before implementation.
4. Reducing the risks associated with lean implementation process.
5. Mapping the future state of organizations' – value stream mapping.
6. Assessing the interaction influence between system's components and parameters.

Based on the above reasons, primarily 1,3 and 5, and through case study application, this paper has developed a framework (Fig. 1) that uses simulation and modelling as an integrated layer between current and future state value stream maps. To achieve this, Section II will give a background overview of lean management, generally and from the case study perpective of distribution. This is followed by a detailed profile of the case study industry; tire distribution, and the case study tire distriubtion company (hereafter to be known as TDC) in relation to lean implementation in Section III. Section IV will develop a current value stream map of TDC using data collected through extensive field work in the industry. This map will then be used in Section V to build an accurate simuation model of the TDC system that can be analysed in Section VI to aid in a future state value stream map before conslusions and future work are discussed in Section VII.

II. Lean Management

Lean management as a philosophy, rather than a stand-alone practice, aims to create a streamlined, high quality system that can achieve a high level of customer service with minimum cost with little or no waste. Originating from Toyota Production System (TPS), lean thinking has become one of the most effective management concepts in the world [6]. Lean processes encompass a wide variety of

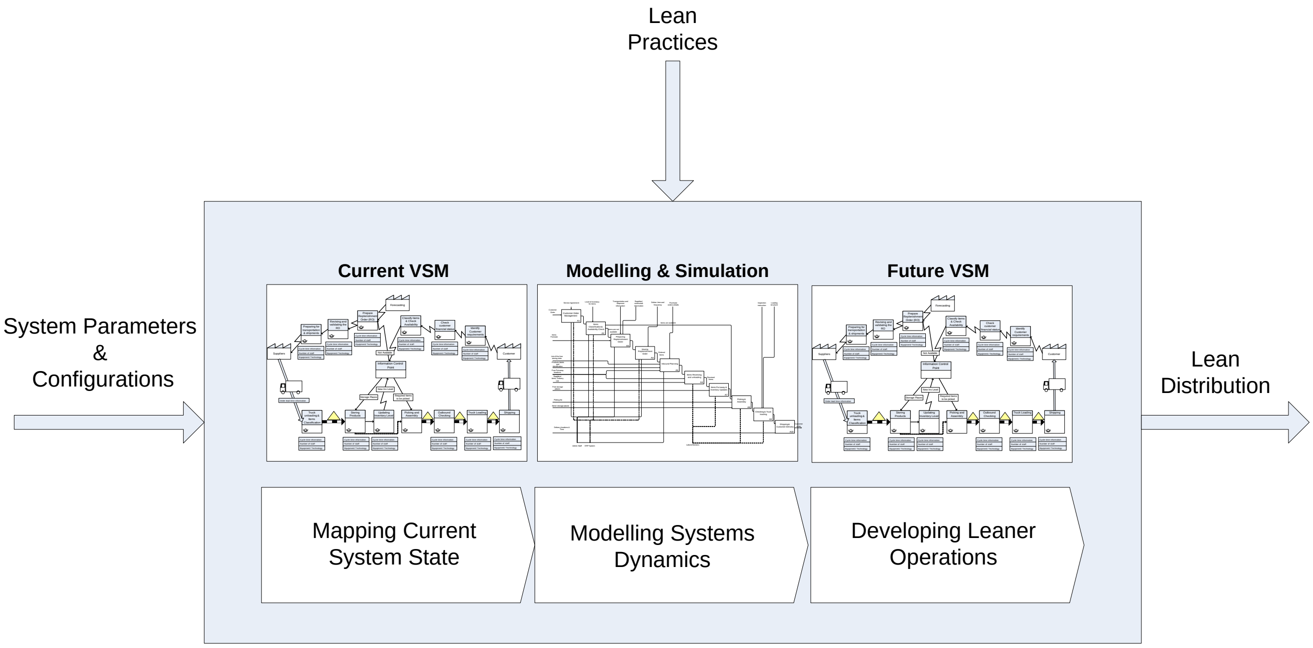


Figure 1. Integrated Value Stream Framework

management practices, and variations in system parameters can have significant impact on its implementation. In the last decade, many authors expanded their lean research beyond manufacturing to include lean services, lean supply chain and lean logistics. As a result, lean definitions became more generic and were identified as a series of activities and strategies that are applied to eliminate operations waste and non-value added processes. Because service applications are subject to a much greater degree of variability than industrial production, new lean practices are required to be applied in service and supply chain environments [7], [8], [9]. Applying lean thinking promotes many changes in system strategies, operational characteristics and human behaviors [10], and has been applied in many manufacturing environments and as many publications have flagged, TPS has given eminent proof of lean capabilities. Survey of the literature reveals the five main principles of lean management to be:

1. Identifying what creates value (from the customer perspective);
2. Identifying value streaming (by understanding all process steps and defining waste);
3. Establishing value flow (without interruption or waiting);
4. Production by pull concept (instead of producing in excess);
5. Achieving perfection (by eliminating all waste elements)

The supply chain environment presented in this paper is that of the distribution center, a strategically critical service provider within the supply chain which has not fully utilized the potential of lean management.

A. Lean Distribution

Despite the continuous growth in academic publications representing the implementation of lean management in service sectors and supply chains, applications of lean on distribution are still scarce [11]. Being the first innovator of lean manufacturing concept in 1960, Toyota is also considered the pioneer in expanding the same concept to other supply chain tiers such as distribution and suppliers [12]. Toyota applied several lean practices on distribution elements such as delivery, ordering, warehouse management, dealers and network structure aiming to reduce the stock level while keeping high service rates [8]. The development of lean practices on distribution by Toyota have been used as the foundations for the development and categorization of lean practices available to the case study company used in this paper.

III. Tyre Distribution Center Case Study

Intensive global competition, reductions in brand loyalty, increasing tire life spans and high costs of raw materials (e.g., natural rubber, bio-chemical materials) have impacted on tire distributors' financial performance negatively and increased the market pressure upon them. Despite market volumes growing by 2.3% over the seven years of 2003-2008, financial growth was just 2.1% [13], and this unsettled the industry's big players and led to a number of mergers and acquisitions, most notably the alliance of Goodyear Dunlop and Bridgestone/Firestone. In this acute atmosphere tire distributors have had to find efficient ways to cut costs and increase efficiency by reducing waste to survive.

Lean thinking is considered a robust concept to reduce different types of waste, so implementing lean practices acts to promote company competitiveness [14]. The important role that distribution activities play in achieving high customer satisfaction levels has prompted many tire distributors to adopt lean components in full or in part. TDC have agreed to implement lean management techniques to certain activities highlighted using the VSM framework developed in this paper. Because of financial confidentiality reasons, TDC would only allow data to be collected on non-financial specific details therefore total costs could not be measured; therefore this paper has concentrated on information and material flows and the parameters and performance indicators specific to them.

A. Tire Distribution Center

TDC is an Irish based distribution center for one of the biggest brand names in the global tire market. It supplies tires for a wide variety of customers ranged between individual customers to large scale companies which in turn impact on the variety of customer orders regarding to items quantities and types. In order to keep as many customers on board, the company's response to its customers has to be fast, accurate, on-time, with the least possible price. Hence, an advanced enterprise resource planning (ERP) system has been applied to link the customers directly to the distribution internal operations, replenishment process and item availability, providing improved transparency and efficiency for orders manipulation (i.e. information flow). The company also provides the proper capacity of equipments, labor and storage spaces to prevent operations bottlenecks and improve item flow. However, many supply chain and operational challenges has risen that prompted the company to think about applying different lean practices as noted in Table I. The lean practices represented in Table I have direct impacts on the distribution operations and can be quantitatively evaluated in terms of time and customer satisfaction.

B. Lean Initiatives on TDC Company

During this study, three of the illustrated lean initiatives in Table I are used in the proposed VSM framework. The selection process of these initiatives was based on a series of interviews, focus group and quality circle of the company's planning and operational teams and managers. The selected initiatives are:

1. Aggregate similar tire types in the replenishment process: Receiving large quantity of similar tire types facilitates the unloading and put-away operation. It also results in an easier planning process for put-away as similar types will be stored close together which in turn accelerate the picking and assembly process. The main drawback of this practice is that the replenishment order might take a longer time to be aggregated under this policy which consequently increases order cycle time.
2. Evaluating staff numbers and increasing labor hour productivity: Low utilization of labor hours can be seen as an operational waste. Non-value adding time (also known as vertical time) where staff is still getting paid increases operational cost and cycle time. Reducing duplication of work, unnecessary staff and increased training can increase staff utilization.
3. Increasing the maintenance services frequency for handling equipments: The breakdown of handling equipments negatively impacts on

TABLE I. LEAN INITIATIVES IN TIRE DISTRIBUTION CENTER

Challenges	Lean Initiative	Initiative Type
Details about item's inventory level, replenishment and delivery process are not clear to customers during processing their orders	Applying ERP system linking the ordering information with items' replenishment and delivery information	Customer Satisfaction
The major company's supplier impose restrictions on supplying particular items due to production restriction in his site	Identifying alternative suppliers even with highest price	Customer Satisfaction
High level of variation in customer's order details (e.g. types of items, items quantities) causes a high variability in picking processing time	Leveling customer demand to isolate the variation of customer orders.	Internal Operations
Low utilization and duplication of TDC staff resources.	Identify where staff utilization is poor and combine jobs to decrease staff numbers.	Internal Operations
Supplies send shipping trucks with high variety of tire types and quantities leading to increasing variability in storage plan and put-away processing time	Aggregating similar tire types in one replenishment order	Replenishment order
Long time is taken to create a full truck load before issuing the replenishment order	Decreasing the lot sizes and increasing the frequent of replenishment order	Replenishment orders
The frequency of the breakdowns for handling equipments is very high	Increasing the frequency of maintenance services for such equipments	Quality and Maintenance

items flow and equipments utilization. Applying regular maintenance services in fixed intervals contributes in decreasing the equipments breakdowns and underutilization.

After discussing the proposed lean practices in TDC, the implementation process of the VSM framework took place with the aid of the company's planning and operational staff. The framework (Fig. 1) was applied as follows:

1. Determine the scope of the study
   Various processes are involved to manage TDC's value stream, starting with receiving customer orders and ending with delivering the products to the end-customers. These processes include marketing, sales, finance, forecasting and planning, inbound and outbound operations and shipment. Lean principles can be employed to eliminate the waste and non-value added activities and isolate sources of variation from company entities – processes and parties –, however the scope of this paper will just focus on sales, planning and forecasting, internal operations and delivery processes in addition to the relationships between TDC and their customers and suppliers. This scope matches TDC points of interest and strategic goals.
2. Mapping the system's current state using VSM approach
   The value stream of TDC is similar to the generic distribution value stream of any distribution center. The company has two main ways to receive orders (1) sales team and (2) online purchasing. This area will be discussed in more detail in Section IV.
3. Collecting data concerning TDC processes and resources
   Three input variables are addressed in the VSM-framework; (1) operations processing times, (2) labors/staff hours and (3) equipment capacity (hours). Despite cost being an important dimension in the leanness measurement process; it was not included in the proposed model due to the confidentiality reasons. The data collection process has focused on three input variables by conducting series of interviews, focus groups, and quality circles of planning and operational teams in addition to observations of the operational activities in the distribution center. Historical data about arrival times of customer orders, the quantity of items in each order, the frequency and items quantities in the forecasting process, the breakdown rates of handling equipments and their repair time are also collected and statistically analyzed as a basic requirement for the simulation stage in the next step.
4. Simulation model for current TDC state
   Creating a conceptual model focusing on the relationships between system's components (i.e. entities and resources) and illustrating their interactions is the first step towards developing a simulation model for the TDC. IDEF language is selected for building the TDC conceptual model where IDEF0 is used to model the upper level of the system illustrating the inputs, outputs, controls and utilized resources for the main functions. This will be discussed further in Section V.

IV. Value Stream Mapping

A value stream is defined as the collection of activities (value added and non-value added) that are operated to produce a product or service or a combination of both to a customer [15]. These actions consider both information and materials flow within the overall supply chain [16]. The logic behind lean thinking is pursuing the optimisation of the value streams from the consumption point of view by eliminating the waste and non-value added activities. In order to identify the sources of waste, non-value added activities and opportunities of improvement, value stream activities have to be mapped using systematic tools and techniques – value stream mapping technique [15]. The VSM technique demonstrates the material and information flow, maps out value-added and non-value-added activities and provides information about time-based performance. This VSM technique is based on generating a current state map that shows the current performance and conditions of the studied systems and a future state map which serves as the target of improvement actions.

Given VSM features and capabilities, the tool is utilized in the first stage of the framework seeking to map the distribution activities and the types of waste and non-value added actions that are embedded in them (Fig 2). Identifying a generic process structure for the distribution function is the initial step towards creating a generic distribution value stream map. A senior manager in TDC, with 35 years operational experience in a variety of departments was interviewed to gather general information about distribution in TDC and the current shape of its supply chains and activities. The industry's current awareness of lean concepts and practices was also a key topic in the discussions and interviews. Meetings were also held with a number of supply chain and logistics professionals with the aim of determining the essential process structure in distribution sectors.

Initial findings from these discussions led to 14 standard operations in a distribution business, classified into three main categories, outlined in Table II. The operations have been modeled based on the standard operations in Table II yet modified to match TDC processes.

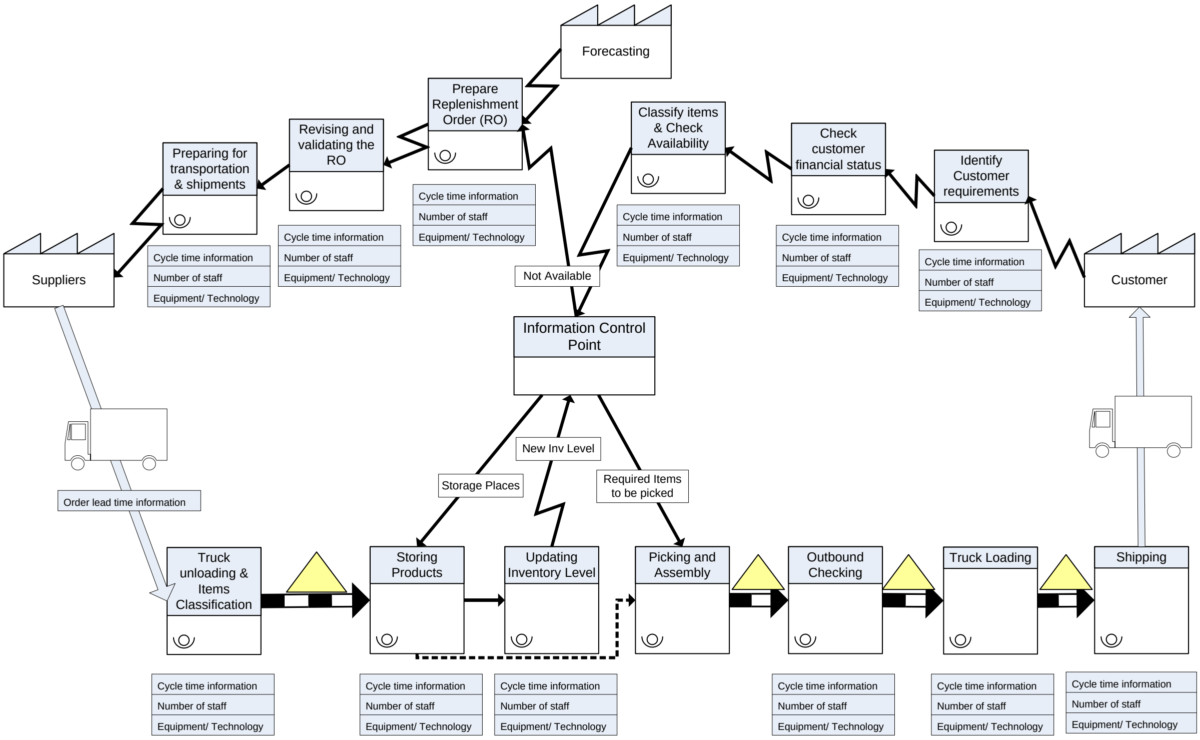

Forecasting
Prepare Replenishment Order (RO)
Revising and validating the RO
Preparing for transportation & shipments
Suppliers
Classify items & Check Availability
Check customer financial status
Identify Customer requirements
Customer
Cycle time information
Number of staff
Equipment/ Technology
Not Available
Information Control Point
New Inv Level
Storage Places
Required Items to be picked
Order lead time information
Truck unloading & Items Classification
Storing Products
Updating Inventory Level
Picking and Assembly
Outbound Checking
Truck Loading
Shipping

Figure 2. Current State Value Stream Map for TDC

TABLE II. DISTRIBUTION CENTER PROCESS CATEGORIES

Process Category	Processes
Orders Management	Orders identification
	Replenishment orders preparation
	Transportation arrangements
	Orders validation
	Orders financing
Inbound Management	Inbound planning
	Tipping
	Put-away
Outbound Management	Outbound planning
	Picking/Assembly
	Checking
	Loading
	Outbound Admin

A set of processes usually starts once replenished items are received. The physical flow (i.e. items flow) in this stage is combined with the information flow throughout the whole process starting from items unloading and classification from the received trucks and ended by loading customer trucks with the required orders. Information and physical flows are interacted in various locations in this path; in the storing process for instance the workers receive information of the storage locations that they should use to store the received products and items. Another interaction is observed when information of the items that need to be picked is passed to the picking and assembly staff to start the picking process. Various buffers are built up between some processes due to the variation in their completion times and labors capacity, for instance the buffers between picking, checking and truck loading processes. The associated data blocks for each process illustrated in the generic state map have shown three different input variables used to distinguish the value added and non-value added status of the modeled processes; total cycle time, number of process staff and resources availability rate (i.e. equipments and technology packages).

VSM has a high quality way of presenting system's parameters such as operations' cycle time and resources capacity and availability; however it does not have the ability to analyze the system settings impact on performance. Similarly, it is also difficult to know if the best future state regarding to the desired level of system performance is achieved. Moreover, value stream maps do not include information regarding variability (i.e. system variations and uncertainty) [17]. Hence, it is required to integrate VSM with another technique that can handle system's variation, show dynamics between system's components, and validate the future state before the real implementation of the improvement steps. Modeling and simulation capabilities can fulfill this requirement. The simulation capabilities will also be represented using the generic distribution structure and parameters mentioned above.

V. MODELING AND SIMULATION

Simulation can be used to master new business concepts such as agile and lean management [18]. The benefits of using simulation as part of lean and six sigma projects was emphasized by [19]. It has been published that simulation can be used to ‘master   Simulation offers more thorough analysis of a system’s data including the examination of variability, the determination as to whether the data is homogenous, and the estimation of the probability distribution that fits the data patterns. This kind of in-depth analysis of data enables simulation to be used to support continuous improvement [4] and to model systems’ future state map showing the ideal state that the system can pursued over time. The advantage of utilizing simulation approach in lean context is not limited to the phase of developing a future state map but is extended to selecting the best alternative to the current system status. This is not within the scope of this paper, but such selection is done by a carefully designed simulation experiments integrated with optimization tools such as Taguchi and response surface methods.

Based on the simulation capabilities and the potential important role it can demonstrate in the leanness assessment process, a generic simulation model mimics distribution operations and displays the interaction between its components, will be associated to the aforementioned distribution VSM. The model represents the general structure of the distribution processes, operations rules, items flow and resources and is developed through two main phases; (1) creating the conceptual distribution model using Integrated Definition Language (IDEF0) (Fig. 3) and (2) using the discrete event simulation to mimic the general features of the distribution systems.

A. Simulation Model

The stochastic technique for discrete-event simulation is selected due to its capability of dealing with the uncertainty resulted by customer demand patterns, the variability in operations times and resources availability in addition to high variance in handling systems [20]. A computer simulation model based on the IDEF0 conceptual model shown in Figure 3 was developed. The model assumptions are (i) no returnable items are modeled (ii) the resources availability rates are based on data collected from managers and (iii) the model focuses on the generic features (Table II) of the distribution activities. The model uses entities to describe the items movement through the distribution center, while resources represent the handling equipments, tools and labor that modify the entities. Resources are characterized by their capacity and availability, whilst the attributes of the entities are arrival time and processing time. Logical entities simulate the decisions for creating, joining, splitting, buffering and branching entities. Each product type has its own information (i.e. level of inventory, safety stock level, forecasting range and its supplier). As previously mentioned, the original purpose of the model is to accurately assess the system’s leanness by handling its variability and uncertainty as well as clearly estimating the system’s future state after lean practices implementation.

Both the current state VSM parameters and the future state VSM parameters will be simulated measuring the following performance indicators:

1. Cycle Time
2. Number of Late Jobs
3. Labor Utilization

The current state VSM has one scenario simulated; before lean implementation, which has no changes to current inputs. The future state VSM runs under

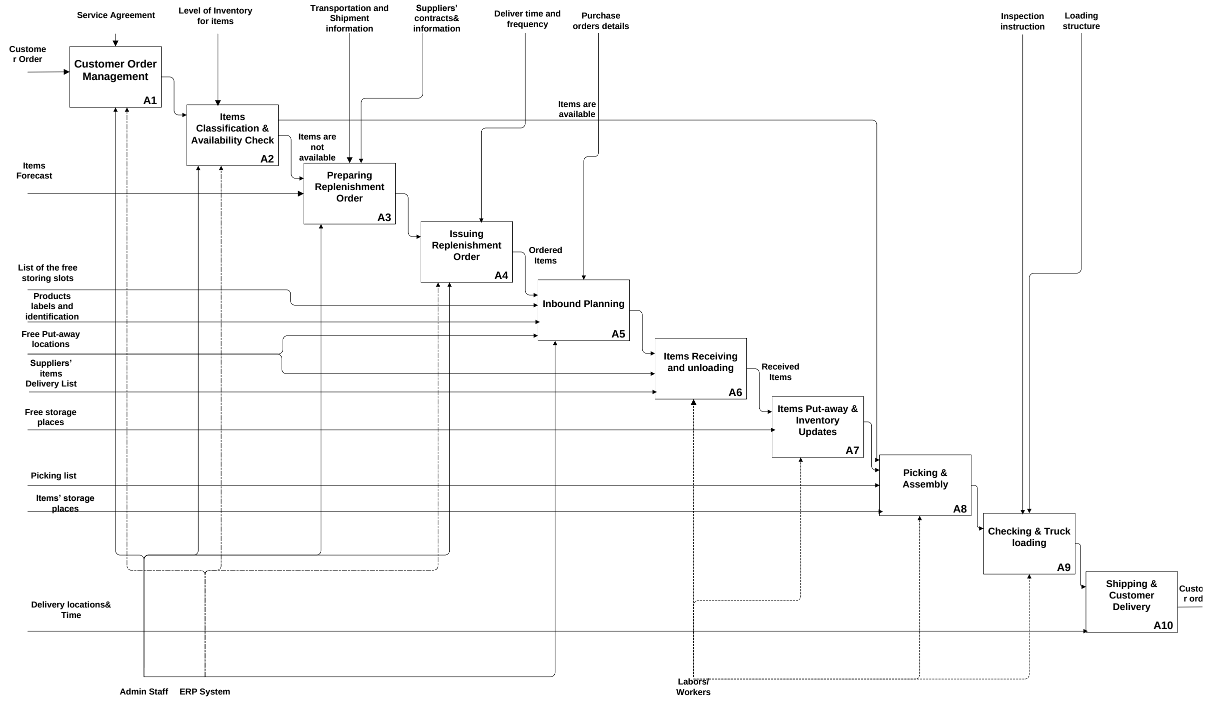

Service Agreement
Level of Inventory for items
Transportation and Shipment information
Suppliers’ contracts& information
Deliver time and frequency
Purchase orders details
Inspection instruction
Loading structure
Custome r Order
Customer Order Management
A1
Items Classification & Availability Check
A2
Items are available
Items are not available
Items Forecast
Preparing Replenishment Order
A3
Issuing Replenishment Order
A4
Ordered Items
List of the free storing slots
Products labels and identification
Inbound Planning
A5
Free Put-away locations
Suppliers’ items Delivery List
Items Receiving and unloading
A6
Received Items
Free storage places
Items Put-away & Inventory Updates
A7
Picking list
Items’ storage places
Picking & Assembly
A8
Checking & Truck loading
A9
Delivery locations& Time
Shipping & Customer Delivery
A10
Custo r ord
Admin Staff
ERP System
Labors/ Workers

Figure 3. IDEF0 Model of TDC Operations

three different lean management scenarios as discussed in Section III. They are; aggregated orders, maintenance and labor resources. The process parameters included in the new scenarios are; truck load quantities, equipment downtime and number of staff respectively.

For the model to reach its steady state condition, the warm-up period was found to be 48 hours. Every simulation run represents one month of actual timing. Each experiment result is an average of twelve independent replications

The simulation model used to model the distribution processes has used a generic package of simulation and customized it using Java and XML technologies. This selection provides flexible and efficient simulation model for three reasons; (1) it helps to provide object-oriented hierarchical and event-driven simulation capabilities for modeling such large-scale application, (2) It utilizes breakthrough activity-based modeling paradigm (i.e. real world activities such as assembly, batching and branching), and finally (3) it also used to customize objects in the package to mimic the real-life application characteristics.

In an effort to make the decisions taken based on simulation models more accurate, efficient methods of verification and validation are needed. For the verification process, in addition to decomposition model (i.e. to verify every group of blocks), a simulation software built-in debugger is used. A decomposition approach is effective in the detection of errors and insuring that every block functions as expected. The studied model has been validated using a ‘Face Validation’ approach that was performed by interviewing managers and operations teams in order to validate the structure of the generic simulation model.

VI. Results Analysis

The uncertain nature of customer demands and suppliers’ lead time makes it difficult to select the best system process parameters that can achieve high level of customer satisfaction (i.e. short cycle time and no late orders) while achieving the goals set out in the VSM. The core theme throughout this paper was to measure the impact certain individual lean process parameters would have on the system before developing a future state VSM. The average results of each simulation run can be seen in Table III.

Decreasing the aggregated orders by nearly 60% has had a significant impact on both cycle time and late jobs, decreasing by 13 days and 3.5 orders respectively, but not surprisingly has not improved labor utilization. Less time spent waiting to replenish orders decreases cycle time which in turn will decrease the chance of delivering orders late, although this may increase total costs as more orders will be shipped more frequently. On the other hand, management’s suggestion that decreasing the probability of breakdowns through applying regular maintenance services in fixed intervals did not materialize, suggesting that equipment breakdowns do not have a significant impact on order fulfillment at present. If management implemented these measures using random estimates and experience alone, it would be a costly mistake to make. Scenario 3 decreased the number of labor hours needed to operate TDC by merging many similar activities such as printing picking notes and picking orders, and sales and customer approval activities. This achieved major economies in labor utilization, increasing by just below 70% and decreasing staff numbers by 2. Although cycle time did not change a great amount due to the fact that the same work was being achieved at the same rate, job lateness decreased by 25%, suggesting the decreased staff numbers were more efficient within the same cycle time. Also, the decreased number of staff and increased labor hour productivity would have decreased operations costs and potentially decrease flow rates in the future.

VII. Conclusion

With ever increasing market pressure and competition, coupled with a global economic recession and high operating costs it has never been more prevalent for organizations to operate at an optimal level. In response, organizations have tried to become more efficient by decreasing costs and

TABLE III. Main effect of lean process parameters on performance Indicators

			Results		
	VSM	**Process Parameter**	**Cycle time (days)**	**No of Late Jobs**	**Labor Utilization**
Scenario 0 – Before Lean	Current State	No Changes	28.755266	3.666666667	0.264369636
Scenario 1 - Aggregated Orders	Future State	Decrease Aggregated Orders to 500 Tires	15.8524	0.25	0.270327248
Scenario 2 - Maintenance	Future State	Decrease Equipment Breakdowns by 50%	25.263963	4.416666667	0.28268486
Scenario 3 - Labor Resources	Future State	Decrease Staff Numbers and Merge Jobs	27.376786	2.833333333	0.440828095

streamlining operations. To achieve this, the philosophy of lean management has gained in popularity.

Through in-depth industry research, it was found that the main obstacle organizations face when implementing lean is deciding which activities to implement lean principals on and to calculate how optimized their decisions are. A well known lean practice, value stream mapping, is a very effective tool in mapping the current and future state of an organizations lean activities. Limitations in calculating variability information that describe the system variations and uncertainty means more powerful analytical tools are needed.

This paper highlights the potential of using simulation technologies in implementing lean practices. Simulation offers a more thorough analysis of a system's data including the examination of variability and has the ability to change certain parameters and measure key lean performance indicators. Using TDC as an applied case study, this paper presents a framework that uses discrete event simulation as an integrative layer between current and future value stream mapping for lean management. The framework accounts for the current value and non-value activities in the company and through simulation have highlighted the activities that should be used when developing the future state map.

This paper demonstrates how simulation can act as a catalyst layer in value stream mapping in order to provide a more accurate future state when lean implementation process is taken place.

Potential future work with TDC on the framework will include the evaluation of the interaction between various future state VSM's using design of experiments integrated with optimization tools such as Taguchi and response surface methods. The application of the framework will also consider system dynamics modeling.

ACKNOWLEDGMENTS

The authors wish to thank the Irish Research Council for Science, Engineering, and Technology (IRCSET) for contributing to the funding of this research project.

REFERENCES

[[1] S. Li, S. S. Rao, T. S. Ragu-Nathan *et al.*, "Development and validation of a measurement instrument for studying supply chain management practices," *Journal of Operations Management,* vol. 23, no. 6, pp. 618-641, 2005.

[2] A. Agarwal, R. Shankar, and M. Tiwari, "Modeling the metrics of lean, agile and leagile supply chain: An ANP-based approach," *European Journal of Operational Research,* vol. 173, no. 1, pp. 211-225, 2006.

[3] M. Christopher, and H. Peck, *Marketing logistics*: Butterworth-Heinemann, 2003.

[4] M. Adams, P. Componation, H. Czarnecki *et al.*, "Simulation as a tool for continuous process improvement." pp. 766-773 vol. 1, 1999.

[5] R. Diamond, C. R. Harrell, J. Henriksen *et al.*, "The current and future status of simulation software (panel)." pp. 1633-1640, 2002.

[6] A. V. Iyer, S. Seshadri, and R. Vasher, *Toyota Supply Chain Management*, New York: McGraw-Hill, 2009.

[7] C. Wright, and J. Lund, "Variations on a lean theme: work restructuring in retail distribution," *New Technology, Work and Employment,* vol. 21, no. 1, pp. 59-74, 2006.

[8] D. T. Jones, P. Hines, and N. Rich, "Lean logistics," *International Journal of Physical Distribution & Logistics Management,* vol. 27, no. 3/4, pp. 153-173, 1997.

[9] P. Hines, N. Rich, and A. Esain, "Value stream mapping: a distribution industry application," *Benchmarking: An International Journal,* vol. 6, no. 1, pp. 60-77, 1999.

[10] R. Shah, and P. T. Ward, "Lean manufacturing: context, practice bundles, and performance," *Journal of Operations Management,* vol. 21, no. 2, pp. 129-149, 2003.

[11] A. Reichhart, and M. Holweg, "L ean distribution: concepts, contributions, conflicts," *International journal of production research,* vol. 45, no. 16, pp. 3699-3722, 2007.

[12] P. Hines, "Internationalization and localization of the Kyoryoku Kai: the spread of best practice supplier development," *International Journal of Logistics Management, The,* vol. 5, no. 1, pp. 67-72, 1994.

[13] T. T. Association, "Tyre Trade News," *Tyre Trade News,* vol. January, 2009.

[14] H. Wan, and F. Frank Chen, "A leanness measure of manufacturing systems for quantifying impacts of lean initiatives," *International Journal of Production Research,* vol. 46, no. 23, pp. 6567-6584, 2008.

[15] M. Rother, and J. Shook, "Learning to See: Value Stream Mapping to Create Value and Eliminate Muda. v. 1.1," *Oct., The Lean Enterprise Inst., Brookline, Mass,* 1998.

[16] F. A. Abdulmalek, and J. Rajgopal, "Analyzing the benefits of lean manufacturing and value stream mapping via simulation: A process sector case study," *International Journal of Production Economics,* vol. 107, no. 1, pp. 223-236, 2007.

[17] C. R. Standridge, and J. H. Marvel, "Why lean needs simulation." pp. 1907-1913, 2006.

[18] D. J. van der Zee, and J. Slomp, "Simulation and gaming as a support tool for lean manufacturing systems: a case example from industry." pp. 2304-2313, 2005.

[19] D. M. Ferrin, M. J. Miller, and D. Muthler, "Lean sigma and simulation, so what's the correlation?: V2." pp. 2011-2015, 2005.

[20] J. Crowe, A. Mahfouz, A. Arisha *et al.*, "Customer Management Analysis of Irish Plumbing & Heating Distribution System: A Simulation Study," in 2nd International SIMUL Conference, Nice, 2010.

Review of Spatial Simulation Tools for Geographic Information Systems

Luís Moreira de Sousa
Instituto Superior Técnico
Lisbon, Portugal
luis.moreira.de.sousa@ist.utl.pt

Alberto Rodrigues da Silva
INESC-ID
Instituto Superior Técnico
Lisbon, Portugal
alberto.silva@acm.org

Abstract—**Spatial simulation has been largely absent from traditional Geographic Information Systems (GIS) software packages. Both the advanced skills needed to use this technique and the relative specificity of its application has resulted in a myriad of independent tools, each with different features. The choice of a proper tool for disclosing the dynamics of change in a GIS context is anything but obvious. This work presents a comparative review of different types of tools available for the development of Spatial Dynamics models. These tools are compared along three different vectors: application domain, ease of use by non-programmers (the typical GIS technician) and interoperability with geo-referenced data. Unlike for other disciplines (e.g. systems engineering) a simulation tool for GIS with a wide variety of application domains but accessible to non-programmers seems largely lacking.**

Keywords: Spatial Simulation; Cellular Automata; Agent Based Modelling;

I. Introduction

The data stored in an information system usually portraits the world as it is now, or was at a specific point or interval in time. This is especially true for spatial data but in this case with the added certainty that it will also evolve. The patterns of land cover and land use, of social, economic, and demographic variables in general, constantly change with time. Objectively, any piece of spatial data is valid only within a specific time frame, just as if any cartographic composition was a still picture taken to the elements represented.

In order to deal with this reality, entire organizations exist with the sole purpose of collecting and updating spatial data, by field campaigns with on site visits, by air borne or space borne data acquisition [1]. Nonetheless, regular data collection provides at best a periodic picture of the changing reality, which for some applications might not be enough [2]. Stakeholders of an information system may need to know not only how the data changed in the past, but in order to plan ahead or otherwise reason upon the data, they may also need to understand why it changed the way it did and how it might continue to evolve in the future.

This need is met recurring to two processes that are part of the same scientific domain: Spatial Modelling and Spatial Simulation. *Modelling* is the process by which the fundamental drivers of change - the *Spatial Dynamics* - are captured into mathematical, logic or functional constructs. *Simulation* is a process through which a model is applied to a set of data during a certain period of time. Modelling and Simulation can be seen as a single technology, for the process of Modelling is chiefly a trial version of Simulation. Spatial Dynamics is captured by applying heuristic or hypothetical models to periods of time for which the data evolution is known, thus allowing for validation and/or calibration. When the model reaches a satisfactory level of success against known data it can then be applied to periods of time for which knowledge is scarce (usually the future) producing new sets of data, pictures of time epochs missing from the base data [2].

The oldest of the techniques used in Spatial Simulation are Cellular Automata (CA) [3] in which the world is discretized in a grid of equal sized cells that evolve in accordance to a fixed set of rules. More recently, Agent-based Simulations have become a popular paradigm that has also been applied to spatial simulation. An agent can be defined as an autonomous object that perceives and reacts to its environment [4,5], a concept that largely benefited from the emergence of Object Oriented (OO) programming. Agent-based Simulations and CA are two concepts that superimpose to some extent in the GIS context, though the former brought new planes of processing where geographic entities not only react to stimuli but also store knowledge and reason before acting. Beyond that, agents can be used to model phenomena that do not have precise geographic meaning, such as social or economic interactions.

On the GIS related sciences, Spatial Simulations have been used extensively, of which the following fields can de highlighted

- *Urban Planning* - understanding and forecasting changes in the urban landscape to allocate new infrastructure [2];
- *Land Use* - studying the dynamics of land use, e.g. changes between agricultural, urban and forest areas [6];
- *Forestry/Wild Fire* - understanding forest growth, studying and anticipating fire spread [7];
- *Biology* - modelling habitat evolution and studying population dynamics [8].

Of the several spatial analysis techniques, Spatial Simulation is the most complex; a simple statistical or mathematical trend analysis, predictive enough for most data recorded in regular information systems, is insufficient in GIS due to the multi-dimensional and heterogeneous character of spatial data. Furthermore, the augmented degrees of freedom of spatial data result in highly specific models, only usable within the particular application in focus. Thus, most spatial dynamics models are developed *ad hoc* by the end user organization, developing its own software libraries. Modern GIS packages continue largely lacking tools dedicated to this technology.

Using a general purpose programming language to build a Spatial Simulation the majority of the instructions coded are extraneous to the concepts in the underlying model. Besides implementing the model, the program has to control the flow of execution, manage system resources, and manipulate data structures. Burdening the model with these tasks can lead to several problems [9]: (i) difficulties verifying the correct model implementation by the program; (ii) limited model generality due to difficult modification and/or adaptation; (iii) difficulty comparing computer models, usually restricted to their inputs and outputs [10]; (iv) problematic integrating with other models or tools (e.g. GIS or visualization packages), often limited to the exchange of output files.

Beyond general purpose programming languages, presently a spectrum of Spatial Simulation tools can be devised, ranging from those that present support at Program-level, closer to the programming language, to those that operate at Model-level, closer to the conceptual model that represents reality [9]. Somewhere in the middle of this spectrum lay Domain Specific Languages (DSL). For each of these categories there is a set of advantages and drawbacks that must be carefully weighted before choosing a particular tool.

This article reviews a series of spatial modelling/simulation tools in the GIS context, in which of the categories presented: Program-level tools (Section II), Model-level tools (Section III) and Domain Specific Languages (Section IV). Section V compares the set of tools reviewed along three vectors: application domain, ease of use and interoperability with geo-referenced data. Section VI sums up the article and its conclusions.

II. Program-level Tools

Program-level support tools extend the facilities available in general-purpose programming languages, usually providing useful software libraries for building specific classes of models. This approach substantially reduces coding time and can increase program reliability. Higher-level code, usually in a general-purpose OO programming language, specifies how objects are used to produce the desired model behaviour. These tools can be called code packages, code libraries or toolkits.

The main advantage of this type of tools is the encapsulation of the model from functionality not directly related to spatial dynamics. These include, graphical display, data input and output, statistical data collection, etc, for which a plethora of functions is provided in the form of a code library. The improvements are two fold: (i) it relieves the modeller from banal programming tasks, allowing a higher focus on dynamics; (ii) it produces leaner and easier to read code, for much complexity is isolated and standardized by the code library.

On the downside these tools require an extra learning effort for their proper use. Beyond having relevant knowledge on the base programming language, a modeller wishing to use on of these tools must learn to some detail the behaviour of at least part of the functions/objects/methods provided by the tool-kit. The more the functionality it has to offer, the longer will it take to fully learn its usage. Besides that, Bennenson and Torrens [11] suggest that with denser libraries, programmers can eventually run into some discomfort with conflicting or incompatible functionality that is only found at later development stages. These disadvantages have been mitigated to some extent with the emergence of user communities that share experience and assistance and by opening and sharing the tool-kit's source code.

De Smith et. al. [12] reported that by 2007 more than 100 of these toolkits were available worldwide. A selection of the most popular is described below.

Swarm was the first of these tools, developed during the 1990s at the Santa Fé Institute, delivering a set of objects and methods for the development of spatial simulations and results presentation [13]. It yearned great popularity in its early days, but integration with GIS is weak, limited to raster data.

The Recursive Porous Agent Simulation Tool-kit (**RePAST**) is a newer Java library that evolved from an eclectic package at the Chicago University, supporting different techniques that go well beyond spatial simulation, which have made it very popular [14]. Perhaps the most useful of these tools today, it also provides good integration with GIS.

The Multi-Agent Simulator Of Neighbourhoods (**MASON**) is also a Java library but conceived with the aim of being light, fast and portable. Conceived at the George Mason University, it is a modern tool, highly compact, that although providing less functionality than RePAST [15], already supports interaction with both vector and raster data sets.

III. Model-level Tools

Model-level support tools allow the usage of spatial simulation models without requiring programming. These are pre-programmed models, designed for specific application fields that can be parametrized by the user. The larger the number of parameters the user can set and update, the larger the tool's flexibility. They allow faster model development and provide fairly straightforward mechanisms for implementation, though invariably constraint the modeller to a specific application and dynamics framework.

The Object-Based Environment for Urban Simulation (**OBEUS**) is a tool dedicated to Urban Planning and Management, developed at the Tel Aviv University, as an implementation of the theory of Geographic Automata Systems [11]. The tool allows the development of models through a graphical interface that then generates a C# coded simulation which maybe further refined by coding in a commercial C# workbench.

AnyLogic is an eclectic commercial tool supporting various areas of simulation, with pre built models on specific areas. It provides several graphical languages to develop model behaviour through state charts and flow diagrams, plus a code library to be used with Eclipse for model refining. It also ships with a GIS API that allows the input of spatial data.

The Tool for Exploratory Landscape Scenario Analyses (**TELSA**) is a program specialized in ecosystems, the typical commercial tool for spatial simulation, allowing the study of different management scenarios. It completely dispenses programming and is parametrizable through a diagrammatic language (VDDT) developed by the vendor, ESSA Technologies [16]. It is dependent on several third party commercial software, included those that provide GIS interoperability.

LANDIS is the result of a joint project of the US Forest Service with several universities of that country, is a simulation for the forest land cover at large scale. The user provides a set of input spatial variables in the form of raster layers for which a number of pre-defined behaviours is available [17].

SLEUTH is the oldest of the tools of this genre, created back in the mid 1990s at the University of California and dedicated to urban development. It uses only six spacial raster layers as input, for which a set of behaviours can be adjusted. It became a popular tools in its domain, being successfully applied to different parts of the world [18].

IV. Domain Specific Languages

Midway between Program-level and Model-level support tools are domain specific tools, usually providing Model-level support for a range of application domains. They make fewer assumptions about the underlying model structure than do pre-programmed models, often providing ways of developing new behaviours. Programming is often required but in a restricted environment where behaviour is described using simple constructs, encapsulating most of the traditional coding activities. A pure DSL provides a programming language, either textual or graphical, as the sole developing infrastructure.

StarLogo/NetLogo is the last of a generation of languages that evolved at the MIT from the functional language *Logo*, specialized on agent-based simulation. Closed source, it has been used as teaching tool due to the simplicity of the code it produces. Nevertheless, it may also be a useful option for prototyping in real life problems [19]. Interaction with GIS is supported, but only for input data sets.

AgentSheets is a simulation tool funded by the National Science Foundation in the United States , developed for teaching purposes whereby models are built in a drag-and-drop interface using graphical stereotypes [20]. It is being used as the basis of several educational courses mostly aimed at high school. Though simulations with a spatial meaning can be developed, no integration with GIS data is available.

The Spatially Explicit Landscape Event Simulator (**SELES**) is a declarative language for landscape dynamics modelling, resulting froma research project at the Simon Fraser University. It tries to balance the flexibility of programming with the ease of use of pre-programmed models [9]. A dedicated development environment is provided, that though closed source, is freely distributable. GIS interoperability is guaranteed by the input and output of raster datasets.

Financed by the Institut National de la Recherche Agronomique (INRA) in France, ,the Modelling Based on Individuals for the Dynamics of Communities (**MOBYDIC**) project produced a programming language dedicated to population dynamics. It allows the development of complex models from simple primitives, close to natural language [21], in reality being a code library for the OO language Smalltalk. It provides no direct interoperability with spatial information.

V. Comparison and Present Difficulties

In this section a comparative classification is performed of each tool according to three vectors of analysis: applications domain, ease of use and GIS interoperability. A three grade system is used: good, medium and weak, denoted respectively by three, two and one stars. In cases where a particular tool doesn't provide support no grade is attributed (represented with the “-” character).

Table I compares the application domain of each tool. In this comparison not only are taken into account the native application areas, but also the tools' underlying platform and distribution flavour. While a certain tool may present itself as a one-size-fits-all solution for spatial simulation, it is important to assess other constraints to its application, such as platform dependency, extensibility or portability. What can be observed from this comparison is that Program-level tools are much more broad reaching in this regard than other tools. Model-level tools or DSL not only narrow their scope in their native application field but also invariably introduce dependencies on third parties, either be it on other software or operating systems. Only two of these tools stand out in this regard: AnyLogic and NetLogo, which attempt at wider portability by adopting Java as platform; nonetheless, being closed source tools, are always at a disadvantage against Program-level tools, especially in scientific applications where verifiability is paramount.

In Table II is compared the ease of use of each tool. Without surprise Model-level tools appear as those easier to learn and use without programming training. These are also almost exclusively those tools that provide graphical interfaces for model development. RePAST provides a diagrammatic interface for behaviour description, but it is somewhat restricted to a single aspect of development. Of the DSL, only AgentSheets provides a graphical development interface, an aspect that casts these tools at visible disadvantage against Model-level solutions.

The last comparison vector, GIS integration, is presented in Table III. Each tool was assessed in terms of its capability to interact with spatial datasets in common formats (Shapefile, TIFF, etc) both as inputs to models and as outputs to visualize results in GIS software. This assessment also distinguished between vector and raster formats, for some data may only be available in one of them (e.g. satellite imagery). The first point to make is that specialization seems also to impose a loss of GIS interoperability. Of the twelve tools surveyed, only five can both read and write some form of geo-referenced data, and of these, only two - MASON and RePAST - operate with both raster and vector datasets. Two of the DSL don't even allow any sort of direct interaction with GIS data. Spatial result output, particularly, seems to be an area where many spatial simulation tools are yet to reach maturity.

Looking at Program-level tools in general, they can alleviate some of the burdening of directly using a general purpose programming language, but still require good programming skills from the modeller [22]. The full knowledge of one of these code libraries is something achievable only with several months of practice [23]. Today these tools are tendentiously open source, by one way or another operating on several computer platforms and providing good GIS integration. Coupling this characteristic to their wider application scope, Program-level tools usually gather around them large communities of users, that provide informal, but extensive, support.

Model-level support tools tend to be quite specific, and much of the model behaviour and assumptions are hidden in the program and may not be explicit or modified; their use in other application fields is largely impossible. The modeller can in fact dispense programming skills using this kind of tools but gets constrained to a specific field and overall simulation behaviour. They also tend to narrow the interaction with geo-referenced data, by imposing certain formats or in some cases by lacking output functionality. Evolution or generalization of these tools can sometimes become too expensive and fate them to extinction. Traditionally they take advantage of market niches providing for the needs of a specific and restricted group of users, thus the commercial nature of many of them. Community support is usually weak or non-existent; more often, support is a paid service.

The use of DSL facilitates modelling and reduces the build-up time of Spatial Simulations, but existing languages do not avoid the need of programming skills. As with any other programming language, the user has to understand keyword meaning and how to compose a set of instructions into a program. Also, in general, these languages produce final models with lower computational performances than those produced with Program-level support tools. DSL for spatial simulation are found mainly for educational purposes, in some cases more resembling toys than analysis tools. This is also patent in the lack of GIS integration most of them show, some even totally lacking such sort of functionality. Users communities tend to be larger than those of Model-level tools, but on the other hand platform dependency is often an issue.

The survey presented can be used as a guide to choose a spatial simulation tool for GIS applications, but the weight of each comparison vector should always be adapted to each particular case. For applications where GIS integration is a relevant need, with both input and output of geo-referenced data being a requisite then MASON and RePAST are nearly the only options. On the other hand, if ease of use is a more important necessity, then models like LANDIS or SLEUTH can be options if matching the application domain. Somewhere in between can be found SELES, that too imposes a relevant application narrowing, and NetLogo, which essentially trades ease of use for GIS integration and extensibility.

VI. Summary

Techniques for Spatial Simulation have existed in one way or another for many decades, actually preceding the emergence of GIS software. It was only with the maturing of the latter that Simulation was envisioned on large scale spatial datasets. In the wake of the OO maturing process, a host of software tools appeared throughout the 1990s providing support for spatial simulation in most (if not all) GIS fields of application.

The main objective of these techniques is to study Spatial Dynamics, the set of local rules or constructs that when repeatedly applied to the variables and space considered produce unanticipated macroscopic results. Spatial Dynamics analysis is a process composed by two main steps: Modelling and Simulation. The Modelling phase discloses the rules by which the variables in analysis changed the way they did; this is usually a prototyping process against a set of historical data. With the model fully developed, it is then applied to the last known state of the space domain for predictive purposes. The results of this process are the drivers of change (the Dynamics) and future evolution of the spatial domain being studied.

Existing software tools for Spatial Simulation can be classified in three types: Program-level, Model-level and DSL. Program-level tools are code libraries providing specific methods for the rapid coding of models with popular OO languages; usually multi-purpose and cross-platform, they gather large user communities. Model-level tools are parametrizable pre-programmed models aimed at strict fields of application; largely dispensing programming skills, they tend to be used by small groups of users and are usually dependent on commercial software or are commercial themselves. DSL try to bridge between the two other types, providing easier model set-up environments without compromising application scope as much as model-level tools; whilst gathering relevant communities in some cases, DSL tend to be mostly educational tools, with fewer examples of real-life application.

Of a set of twelve different simulation tools surveyed only two showed to be fully matured when it comes to the integration with GIS data, both Program-level libraries: MASON and RePAST. A trend is apparent whereby ease of use implies a loss of functionality regarding geo-referenced data input and output; some Model-level tools show some degree of GIS integration but impose a significant scope limitations. NetLogo is the tool closest to bridge this gap, though impaired by a closed source philosophy and lack of geo-referenced data output.

All the tools considered, with no exception, present important compromises in their choice for spatial simulation in the GIS domain. Space for improvement in the field seems to exist.

References

[1] Kraak, M. and Ormeling, F., "Cartography: Visualization of Spatial Data". Prentice Hall; 3rd edition. 2009.

[2] Batty, M., "Cities and Complexity", MIT Press, 2007.

[3] Wuensche, A. and Lesser, M., "The Global Dynamics of Cellular Automata", Addison-Wesley, 1992.

[4] Ferber, J., "Multi-Agents Systems. In: An Introduction to Distributed Artificial Intelligence", Addison-Wesley, 1999.

[5] Weiss, G., ed., "Multiagents Systems: a Modern Approach to Distributed Artificial Intelligence", MIT Press, 1999.

[6] Messina, J.P. and Walsh, S.J., "The application of a cellular automaton model for predicting deforestation", In: Proceedings of the 4th International Conference on Integrating GIS and Environmental Modelling: Problems, Prospects and Research Needs, Banff, Canada, 2000.

[7] Li, X. and Magill, W., "Modeling fire spread under environmental influence using a cellular automaton approach", Complexity International, 8, 1–14, 2001.

[8] Ermentrout, G.B. and Edelstein-Keshet, L., "Cellular Automata Approaches to Biological Modeling", Journal of Theoretical Biology, 160, 97–133, 1993.

[9] Fall, A. and Fall, J., "A domain-specific language for models of landscape dynamics", Ecological Modelling, 141, 1–18, 2001.

[10] Olde, V. and Wassen, M., "A comparison of six models predicting vegetation response to hydrological habitat change", Ecological Modelling, 101, 347–361, 1997.

[11] Benenson, I. and Torrens, P., "Geosimulation: Automata-based modeling of urban phenomena", London: John Wiley & Sons, 2004.

[12] de Smith, M.J., Goodchild, M.F., and Longley, P.A., "Geospatial Analysis: A Comprehensive Guide to Principles, Techniques and Software Tools", Troubador Publishing Ltd, 2007.

[13] Minar, N., Burkhart, R., Langton, C. and Askenazi, M., "The Swarm simulation system: a toolkit for building multi-agent simulations", 1996.

[14] Collier, N., Howe, T., and North, M., "Onward and upward: the transition to Repast 2.0", In: First Annual North American Association for Computational Social and Organizational Science Conference, Pittsburgh, Pa, 2003.

[15] Luke, S., et al., "MASON: A Multi-Agent Simulation Environment". Simulation: Transactions of the society for Modeling and Simulation International, 82(7), 517–527, 2005.

[16] Merzenich, J. and Frid, L., "Projecting Landscape Conditions in Southern Utah Using VDDT", In: M. Bevers and T.M. Barrett, eds, Systems Analysis in Forest Resources: Proceedings of the 2003 Symposium, Portland, OR, 157–163, 2005.

[17] Mladenoff, D., "LANDIS and forest landscape models". Ecological Modelling, 180 (1), 7 – 19, 2004.

[18] Yi, W. and He, B., "Applying SLEUTH for Simulating Urban Expansion of Beijing", In: 2009 International Joint Conference on Artificial Intelligenc, Vol. 2, 652–656, 2009.

[19] Railsback, S.F., Steven, L.L., and Jackson, J.K., "Agent-based Simulation Platforms: Review and Development Recommendations", Simulation, 82 Issue 9, 2006.

[20] Repenning, A. and Ioannidou, A., "Broadening Participation through Scalable Game Design", In: in Proceedings of the ACM Special Interest Group on Computer Science Education Conference, (SIGCSE 2008), 305–309, 2008.

[21] Ginot, V., Le Page, C., and Souissi, S., "A multi-agents architecture to enhance end-user individual based modelling", Ecological Modelling, 157, 23–41, 2002.

[22] Tobias, R. and Hofmann, C., "Evaluation of free Java-libraries for social-scientific agent based simulation", Journal of Artificial Societies and Social Simulation, 7 (1), 2004.

[23] Samuelson, D. and Macal, C., "Agent-Based Simulation Comes of Age", OR/MS Today, 33 (4), Lionheart Publishing, Marietta, GA, USA, 2006.

TABLE I. COMPARISON OF THE TOOLS SURVEYED REGARDING APPLICATION RANGE.

		Application	**Programming Platform**	**Distribution**	
Program-level tools	**Swarm**	Multi-purpose	Objective-C	Open Source	★★★
	MASON	Multi-purpose	Java	Open Source	★★★
	RePast	Multi-purpose	Java, .NET	Open Source	★★★
Model-level tools	**OBEUS**	Urban Planning	.NET	Shareware	★
	AnyLogic	Several Specific	Java	Commercial	★★
	TELSA	Landscape Management	unknown	Commercial	★
	LANDIS	Forest Succession	.NET	Shareware	★
	SLEUTH	Urban Development	C	Open Source	★
Domain Specific Languages	**NetLogo**	Multi-purpose	Java	Shareware	★★
	AgentSheets	Educational	unknown	Commercial	★
	SELES	General Landscape	unknown	Shareware	★
	MOBIDYC	Population Dynamics	Smallralk	Open Source[a]	★

a. Dependent on commercial software.

TABLE II. COMPARISON OF THE TOOLS SURVEYED REGARDING EASE OF USE.

		Modelling Language	**Development GUI**	**Programming Skills**	**Community**	
Program-level tools	**Swarm**	Objective-C, Java	none	high	Wiki, Mail-list	★
	MASON	Java	none	high	Mail-list	★
	RePAST	Java, C#, others	Eclipse	high	Mail-list	★
Model-level tools	**OBEUS**	C#	yes	low to high	none	★★
	AnyLogic	Diagrammatic, Java	yes	low to high	none	★★
	TELSA	VDDT	yes	none	none	★★★
	LANDIS	Parametric	none	none	Forum	★★★
	SLEUTH	Parametric	none	none	Forum	★★★
Domain Specific Languages	**NetLogo**	Logo specialization	none	low to medium	Mail-list	★★
	AgentSheets	Conversational Prog.	yes	none to high	none	★★
	SELES	Declarative DSL	none	none to medium	Wiki, Forum	★★
	MOBIDYC	Declarative DSL	none	none to medium	none	★★

TABLE III. COMPARISON OF THE TOOLS SURVEYED REGARDING GIS INTEROPERABILITY.

		Input	**Output**	**Vector**	**Raster**	
Program-level tools	**Swarm**	√	×	×	√	★
	MASON	√	√	√	√	★★★
	RePast	√	√	√	√	★★★
Model-level tools	**OBEUS**	√	×	√	×	★
	AnyLogic	√	×	√	×	★
	TELSA	√	×	√	×	★
	LANDIS	√	√	×	√	★★
	SLEUTH	√	√	×	√	★★
Domain Specific Languages	**NetLogo**	√	×	√	√	★★
	AgentSheets	×	×	×	×	-
	SELES	√	√	×	√	★★
	MOBIDYC	×	×	×	×	-

Towards Internet Scale Simulation

Anthony J McGregor
Computer Science Department, The University of Waikato
Hamilton, New Zealand
Email: tonym@cs.waikato.ac.nz

***Abstract*—Simulation of the Internet has long been understood to be very challenging mostly because of its scale, diversity and the lack of detailed knowledge of many of its components. However, two recent developments (macroscopic topology discovery and large memory servers) mean that some of these problems are now more tractable. Although problems like the lack of detailed link information remain, models are are useful for some problems that require an understanding of how an application interacts with the Internet as a whole. The paper presents is-0, an Internet Simulator. is-0 derives its model of Internet topology directly from the output of an Internet topology mapping project. Efficient design allows is-0 to simulate packet-by-packet, hop-by-hop behaviour at Internet scale. Validation of is-0, an example application and performance measurements are included.**

***Keywords*-Discrete Event Simulation, Internet Simulation.**

I. Introduction

Understanding the performance of the Internet and the way that new applications and protocols will perform and interact is a challenging problem that has been noted by many authors. For example, in 1997, Paxon and Floyd wrote:

> "As the research community begins to address questions of scale, however, the utility of small, simple simulation scenarios is reduced, and it becomes more critical for researchers to address questions of topology, trafc generation, and multiple layers of protocols." [1]

The main challenges they noted were: heterogeneity in nodes, links and protocols; rapid rate of change including size, applications, protocols and traffic characteristics; large size; and the difficulties of building traffic models [1], [2]. These problems are "moving targets". Not only is the Internet big and diverse, it is growing rapidly in both respects. The challenges of Internet simulation, especially the scale of the Internet, has been reiterated by many others including [3] [4] [5] [6] [7].

Two developments have made it possible to make progress towards simulation of the Internet as a whole. These are: Internet macroscopic topology discovery projects and a large increase in the memory capacity of commodity servers. In recent years, there have been several projects that are producing useful maps of the Internet. These include CAIDA's ARC infrastructure [8] running the Scamper mapping tool [9] and Dimes [10]. Further progress on mapping the macroscopic topology of the Internet can be expected in the coming years with projects like the RIPE NCC Atlas [11]. The entry of large memory servers to the commodity market is driven by the trend towards vitalisation. Computers with up to 512GB of RAM are now available at less than US$40,000.

Together these two developments mean, it is now possible to build a packet, node and link level simulation model for the Internet as a whole that will run on commodity server hardware. There are still many challenges that prevent high fidelity simulation of the Internet including lack of knowledge of the characteristics of each link and the cross traffic links carry. However, it is possible to make models that are useful for some problems including those where the topology of the Internet interacts with a system in complex ways but where fine grained temporal results are less important. Examples include content distribution, reducing peer-to-peer traffic loads and multicast optimisation.

In this paper, we describe the use of simulation to investigate the traffic load created by large scale use of the DoubleTree optimisation [12] for topology discovery. This was motivated by the desire to implement a Hubble [13] like application on Atlas [11]. We present a new, open source, simulation system, **is-0**. The system includes a discrete event simulator and surrounding infrastructure to support the process of converting the output of an Internet mapping project to a topology model suitable for simulation, running a set of simulations across a range of parameters values utilising the massively parallel nature of most simulation experiments and presenting outputs.

There are other open source, discrete event simulators. Some, like ns-2 [14] are well established and we might have based this project on one of them. However, the core of a discrete event simulator is simple and most of the contribution of this project lies in managing scale and in supporting the whole process of taking an Internet topology map through to the result of simulation experiment, possibly involving many individual simulations (see Figure 1). **is-0** must meet the needs of a large topology and, potentially, billions of packet events. On the other hand, it provides less fine grained temporal behaviour than many other simulation projects. The need to optimise performance in terms of both memory and CPU cycles leads to the requirement for an implementation tailored to meeting these needs in the

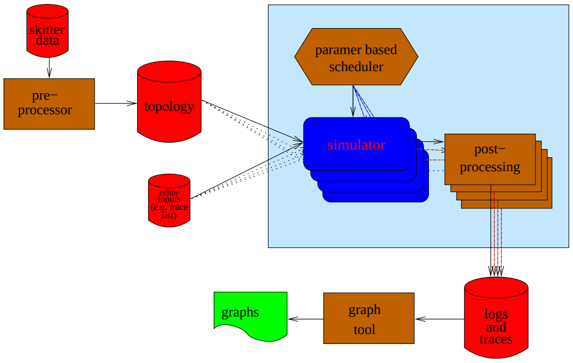


Figure 1. **is-0** Architecture

context of an Internet model.

There is a great deal more planned for **is-0**. We are prereleasing it now because it has already demonstrated its usefulness to us in an important research area. We believed that, other researchers will find **is-0** useful. The rest of this paper describes the system in more detail. An example application and performance statistics are presented.

II. **is-0** OVERVIEW

A. Topology

The current version of **is-0** builds its topology from Scamper [9] output files. Scamper measures the global macroscopic Internet topology from a set of measurement points to one destination in every routed IPv4 `/24`. The output from Scamper is a set of runs where each run contains a traceroute style probe to every destination address from one monitor (team probing).

From this data, **is-0** builds a topology model of the Internet based on nodes and links between them. The nodes and links alone are not enough to route packets; routing information is also required. In the simulator, routing information at each node represented as a table of destinations and the associated next-hop. This information is inherent in the Scamper data set and it is simply maintain in the simulator topology. For conciseness, we refer to the data structure as the "topology model" even though it contains elements of topology and routing. The following sections refine the topology model as it is built from Scamper data.

B. Interfaces vs Routers

Scamper, like all traceroute based tools, discovers interfaces not routers; the raw data does not show which interfaces are on the same router. The simulator topology is, therefore, also built in terms of interfaces not routers. This is not normally problematic because simulations are performed in terms of packets being passed from interface to interface. References in this paper to nodes in the topology model are to a particular interface (not a router). Similarly, links are between interfaces.

C. Discarded paths

Some of the paths in the Scamper data are not usable in the simulator, mostly because they are not well formed. For example, Scamper discovers loops in some paths. In others, it abandons tracing because too many nodes do not reply with a TTL expired message. In these cases, a complete path from the source to the destination is not discovered and the path can not be used in simulation. The Scamper data set used for the example in section example had 5.6 million paths discovered in part or full by Scamper. Of these, 241,763 (4%) were omitted because of the reasons described above.

D. Alternative Paths

In some cases, Scamper discovers alternative paths between nodes. Alternative paths may arise because of load balancing or because the topology has changed during the measurement. Scamper may discover different paths between the same nodes when it probes between different source/destination pairs. These alternatives are maintained in the topology model. The same path variant is used when packets are sent from a source to a destination as was discovered when Scamper measured the route between the two. In the topology data structure, this is done by including a source as well as the destination in the next-hop table.

match the behaviour of the Internet in all cases, however it is likely to be correct in most cases. If the source of the alternative paths is a path change during Scampers probing, either path is acceptable for the simulation. It is not required that we maintain both paths in this case but it is acceptable. In the case where there are alternative paths due to load balancing, using the same path as the one Scamper discovered for this source/destination pair will mostly match the behaviour of the Internet. This is because per-destination and per-flow load balances are more common than per-packet load balances in the Internet [15].

E. Unknown Paths

Scamper data does not provide a complete map of the Internet. While it contains paths from the monitors to most destinations it does not have the reverse paths or paths between destinations. The extent of this missing topology is not currently known.

The lack of return paths is resolved in the simulator by adding a symmetric path from the destination back to the source. It is known that Internet paths are not always symmetric [16] . For many simulations, it is the overall structure of the Internet (i.e., path lengths and branching) not the exact details of particular paths that is important. If this is the case, the symmetric nature of paths will not unduly influence simulation results. However, without a measured non-symmetric topology, there is no way of demonstrating that this is true for a particular experiment.

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-169-4

The omission of paths between destinations is not problematic for simulations where packets are only sent between sources and destinations (as is mostly the case in our example). If this is not the case, the simulator has the ability to add extra paths to the topology. These paths are, by necessity, not based on measured topology. Extra paths are discovered using a breadth first search from the source to the destination using links that were discovered by scamper. If paths from single source to multiple destinations are required, a single search can create all the paths.

Added paths do not necessarily follow the same route as in the Internet. The breadth first search finds the shortest path based on the hops that were discovered by Scamper. While Internet routing is designed minimise path length this is in terms of the ASs that a path passes through. This may use a link Scamper did not discover or, within an AS, the shortest path may not be followed. Internet routing may also includes policy which restricts the choices for a path.

An understanding of the extent to which this affects the results of a particular simulation may be gained from a comparison of the performance of Scamper measured paths and the equivalent paths formed by the methodology above. A future release of **is-0** will include automated sensitivity analysis including the effect of added paths (although cross traffic sensitivity is the highest priority for automated sensitivity analysis).

F. Missing hops

During traceroute style probing, it is common for some hops to not reply with a TTL expired message. Often the hop is known to exist, because later hops do respond, but the address of the hop is unknown. In the data set used for the example in section VII, approximately 22% of hops are not identified. Within the simulator, these non-responding nodes are given a unique address.

This procedure may not exactly replicate the structure of the Internet at the time Scamper was probing. It is possible that, a missing hop in two different paths might be the same interface, however, this approach inserts two different interfaces. Automated sensitivity analysis could also allow the impact of this effect to be determined.

G. Topology Data Structure

Within the simulator, the topology is represented in a data structure based on nodes and links. Links contain a reference to the node at each end of the link, the link latency, serialisation rate, and the current link state (queue length and when the current packet, if there is one, will have been completely added to the the link). Links also contain performance metrics including packets dropped, packets sent by packet type and the peak queue length.

Nodes include their address (IP address or missing node address) and a table of references to links that leave this node. The table is indexed by either the destination (of the path, not the next hop) or, where there is more than one path to a destination (see section II-D), the source (of the path) and the destination.

Type	**Hooks**
Packet Events	newPacketHook, packetQueuedHook, packetArrivedHook, packetDropHook, ttlExpiredHook, changePacketTypeHook
Simulation Start and Termination	startHook, usageHook, argsHook, logConstantsHook, cleanupHook, heapMapValidHook, buildHook, newNodeHook, saveBuildGlobalsHook, restoreBuildGlobalsHook
Hash Management	newHashEntryHook, freeHashElementHook
Reporting	progressHeaderHook, progressHook, printStatsHook, packetInfoHook, summaryStatsHook, specialAddrHook, nodeSummaryHook

Figure 2. API Event Hooks

H. Building The Data Structure

The raw Scamper data is pre-processed into a record for each node that contains the next hop links from that node. The resulting files are large. For the example application described in section VII, they total a little over 2GB. Before a particular simulation can be run, this information must be built into the internal simulator data structures including the hash tables, initialisation of performance metrics etc. Any additional paths must also be added to the topology. The resulting data structure is large (in the order of 8GB for the example application) and it takes several minutes to load and build. If many simulations are to be run (around 8,000 were needed for our example experiment) the total time taken to repeatedly load and build the data structure is significant and the size of the data structure may limit the number of simultaneous simulations that can be run on a machine.

is-0 can store the topology data structure in a memory mapped file. This has two advantages. Firstly, the data structure can be reused, avoiding most of the time otherwise required for building it. Secondly, memory mapped files can be shared and only a single copy kept for their read only components. This may allow more simulations to be run in parallel reducing the time required for large topologies on machines with many cores.

III. Parameter Exploration

Scripts that manage the process of concurrently running as many parallel simulations as the hardware supports with different parameters are included with the simulator. These scripts manage the process of running simulations, recording the results in appropriately named files, first cut checking that simulations complete successfully, re-starting batches after an interruption and logging and reporting overall progress.

Scripts that produce plots over different combinations of parameters are also included. Users can select the parameter for the x- and y-axes and also an additional parameter (and perhaps some specific values of this parameter) if more than one line is to be drawn per plot.[1] While much of this is mundane, in a typical simulation experiment considerable researcher time is spent on these mundane matters and **is-0** can significantly reduce this effort. A future release will include more sophisticated parameter space exploration inspired by Nimrod [17] and other similar tools. This will reduce the number of simulations that need to be run as part of an experiment by focusing attention on parts of the parameter space that cause significant variations.

IV. API

The simulator API has three components: data structure augmentation, event hooks and utility routines. These are described in the following sections.

A. Data Structure Augmentation

Application related data structures (like packets and nodes) can be extended. Our example application implements a traceroute like protocol so we have added a probe packet type that contains, amongst other things, the value that the TTL field had when the probe ended its outward journey and began to return to its source. The data structures that can be augmented in this way are: packets, which can have new packet types and/or additional generic fields in all packets; nodes; hash tables; and the set of global variables that is saved when a memory image is created and then restored when the image is reloaded for a particular simulation run (see section II-H above).

B. Event Hooks and Calls

The second component of the API is a set of event routines that can be called when simulation events occur that might be important to an application. For example, the example application uses `ttlExpiredHook`. There are currently 25 hooks as shown in Figure 2.

The API also includes 51 function calls (and related constants, macros and data structures) as shown in Figure 3.

Type	**Calls**
Sim. Infrastructure	`intArg boolArg usage queueEvent warpTime simMalloc simFree`
Hash	`makeHash freeHash makeSpaceInHash hashSize changeHashSize find dumpHashTable forEachHashEntry`
Address	`addr2str str2addr extractAddr addrEqual addrCpy setAddr2null`
Topology	`addLink queueLength findDistances makeDistancesFile processDistances addExtraPath`
Packet	`disposeOfPacket changePacketType makePacket packetInfo queuePacket4nextHop queuePacket addPacket2link swapAddrs`
Utility and progress reporting	`chopNl commas comment flag2bool processEachLine add2file lastModified fileSize skipUnknownTags getTaggedLine registerProgressReport unRegisterProgressReport recordTraceEvent`
For parameters	`newExploreNode freeExploreList packetArrivedEvent`

Figure 3. API Calls

V. Validation

The goal of the project is to simulate the Internet as a whole. As a consequence, it is not possible to compare the results of simulation with the real system. However, four other types of validation have been performed. These are: internal consistency, manual validation against a simple network scenario, external generic behaviour validation and external application specific validation.

A. Internal Validation

The simulator contains a substantial amount of internal consistency checking. There is liberal use of `asset` statements, particularly for pre-and post conditions (currently, there are > 400 asset statements in the C code base of 12,500 lines). The C code also contains a hierarchy of more extensive tests. These include, for example, checks on the consistency of data structures, sensibility checks on behaviours (e.g. that packets leave a FIFO queue in the order they entered) and event queue checks. There are none levels of the hierarchy and approximately 120 blocks of checking code. The simulator is also run under `valgrind` to ensure there are no uninitialised variables, accesses to freed memory or memory leaks.

[1] Currently, these scripts have not been fully generalised and are somewhat tailored to our example problem. However, changes required for other applications are not expected to be great.

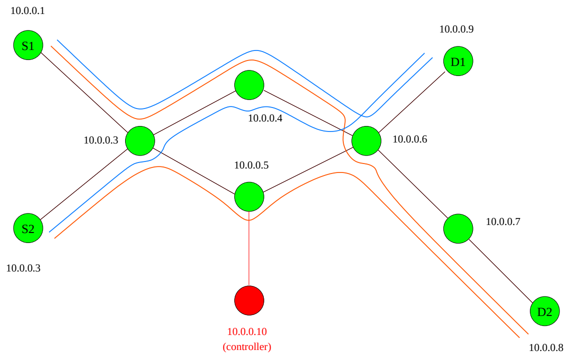


Figure 4. Network used for Hand Validation

B. Hand Validation

A simple network (see Figure 4) and test scenario was designed and the example application simulated. This involved nine nodes, nine links, including a diamond topology and alternative load balanced paths, and two traces. The simulator was configured to record a full trace of all events for each link and packet. These traces were then analysed by hand to check for consistency, errors and correct path discovery.

C. External Validation

Simulator trace files can also be checked for correct behaviour using a separate programme once a simulation is complete. Separate tests check that all packets are delivered exactly once, that they are delivered in sequence and that the transmission and queueing time is consistent with the links latency and serialisation rate.

The external validator was written by the same programmer who coded the main simulator but several months after the original coding and in a different language (perl). It is much slower and uses a lot more memory than the simulator so it is not suitable for use with every run of the simulator, rather it is normally used to check a representative set of results. It was used to check the simulations that were hand validated.

In addition to generic validation, which only relies on features of the base simulator, external validation was also applied to the example application. This involved running traceroutes over a network configuration and then running a program that checked that the paths discovered were correct. This program was coded in a similar way to the generic external validation.

VI. Performance

Achieving the required performance, including balancing memory use and CPU cycles, required careful implementation. This was informed by extensive use of the kcachegrind/valgrind tool set. The following sections describe two optimisations in the design (but there are may others). Section VII-A contains further performance statistics in the context of the example application.

The simulator uses five different types of hash table to improve performance. There are millions of hash tables in use in any particular simulation run. For example, any node can be found from its address via a hash table and the table of next hops within a node has an associated hash table.

A common hash mechanism is used across all hashes. It supports look up from one or two keys (e.g. the destination or source/destination pair), insertion and deletion, chaining through all valid entries, resizing of the hash table, and hash performance statistics. Collision resolution is managed via an alternate hash calculation and, if that also collides, by linear chaining. Use of indirection minimises the memory use of hash tables which normally have many unused entries.

The hash infrastructure includes (optional) performance metrics and automatic hash resizing to maintain sufficient head space for efficient operation. Resizing is relatively expensive (especially for large hashes) so it is avoided where possible. The topology pre-processing produces size hints that remove the need for most resizing.

A. Event Queue

The simulator event queue is optimised for Internet simulation. In particular, most events are added for times in the near future and there are often many events at the same time. A fixed size, event hint look-aside table allows the correct queue location for most events to be found with a single table look up.

VII. Example

The initial motivation for development of **is-0** was to investigate the potential for DoubleTree [12] to reduce the cost of measuring the path from many sources to a few destinations. This problem stems from the desire to design an implementation of an application like Hubble [13] on an infrastructure consisting of up to 100,000 vantage points (a design goal for the RIPE Atlas [11] project). In previous DoubleTree work, the few sources, many destinations scenario was investigated. [12]

The code for this application is included with the simulator as an example. It is divided into two parts, traceroute and extensions to traceroute for DoubleTree. Implementing traceroute took 950 lines of code including comments, checking and reporting code.. DoubleTree required an additional 1,300 lines of code.

Donnet built a simulator to explore DoubleTree's behaviour but was not totally happy with the level of detail that it provided. [18]. We believe that, had **is-0** been available at that time, he would have had access to detailed modelling with less effort. In turn, this may have allowed more development of DoubleTree for the same effort.

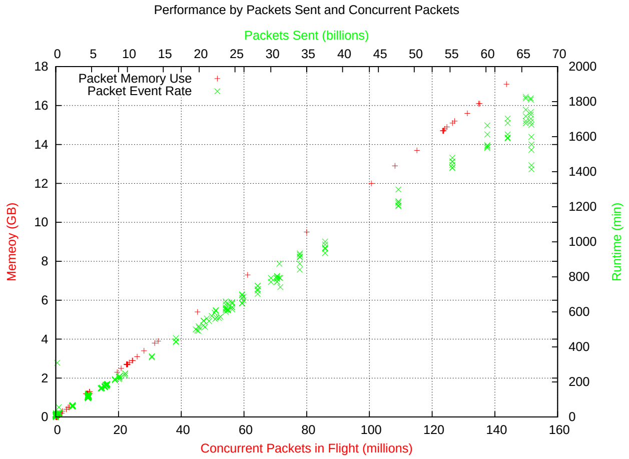


Figure 5. Resources by Number of Packets Sent and Concurrent Packets

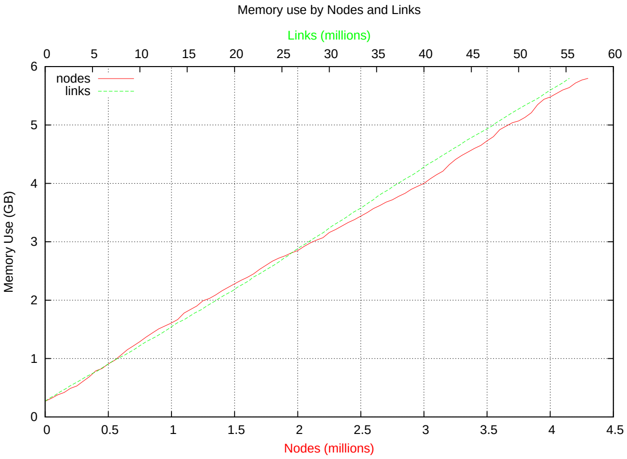


Figure 6. Memory Use Against Nodes and Links in the Topology

A. Performance

The Internet model in this example used a Scamper run from 3 Jan 2009. The typology had 4,300,000 nodes and 55,000,000 links (source/destination based next hops). 22,000 traceroutes were performed in each simulation. A total of 1835 simulations were run for each investigation with different parameters (e.g. starting TTL, trace schedule, stop set exchange).

The resources used per simulation varied as the simulation parameters were changed. A typical example `(DoubleTree=yes, sources=many, scheduling=1stage, probing=team, ttl=1 stopSetBin=250ms)` required simulation of sending 25,000,000 packets, had a peak and mean memory usage of 8GB and ran in 945s real time. A small number of (pathological) parameter combinations required far greater resources. For example, `(DoubleTree=yes, sources=many, scheduling=1stage, probing=team, ttl=1 stopSetBin=1ms)` required simulation of sending 2.3 billion packets, peak and mean memory usage of 15GB and 10GB (respectively) and ran for 4,103s. At the time of peak memory usage, there were 123 million packets in flight.

It is likely that, over time, growth of the Internet and more extensive topology discovery will result in larger topology models. **is-0** was designed to permit multiple CPU cores to be used in parallel on a single simulation. While this is not yet fully implemented, planning for it is well underway and it will be one of the first extensions implemented in future release.

Figure 5 shows the relationship between the number of events simulated and the memory use and real time taken for the example application. The graph demonstrated event rates of around 650,000 events/s on our hardware[2] Figure 6 shows the relationship between the number of nodes and links in the topology and the total simulator memory required for a null simulation. This data was collected by building the data structure for the first n nodes in the topology.

More details of this simulation and its results are available in [19].

VIII. Conclusion

is-0 only addresses a few of the challenges of simulation of the Internet as a whole. However, it has proven useful and we believe others will find it helpful in exploring problems that interact with the Internet as a whole. **is-0** supports simulations with millions of nodes and billions of packets on commodity hardware. It builds its topology model directly from the Scamper Internet macroscopic topology discovery project data. It also includes parameter exploration and graphing tools to reduce the time required to undertake Internet simulation experiments.

Some of the plans for extending **is-0** have already been mentioned. These include: using other sources of topology data (e.g. Dimes [10]); Nimrod [17] style parameter space exploration; and automated sensitivity analysis.

Currently, parallel use of **is-0** relies on the embarrassingly parallel nature of most simulation experiments by running multiple simulations concurrently. Support exists to reduce the memory overhead in this case. However, some simulations are long and this is expected to be more common as the Internet continues to grow and better models of the Internet become available. Support for employing multiple cores within a single simulation has been designed and will be included in a future release. The code for **is-0** is available from `http://research.wand.net.nz/software/`.

[2] Intel Xeon X5570, 2.93GHz, 8192KB cache, 800Mhz DDR3 triple channel memory, 12 concurrent simulations over 8 physical cores with two hyper-threads each.

ACKNOWLEDGEMENTS

This work was undertaken, in part, while the author was on sabbatical with the RIPE NCC. My thanks for their support both practical and academic during this time. I am grateful to the RIPE NCC and CAIDA for making the data used in this work available. Support for the CAIDA IPv4 Routed /24 Topology Data set is provided by the National Science Foundation, the US Department of Homeland Security, the WIDE Project, Cisco Systems, and CAIDA Members.

REFERENCES

[1] V. Paxson and S. Floyd, "Why we don't know how to simulate the internet," in *Proceedings of the 29th conference on Winter simulation*, ser. WSC '97. Washington, DC, USA: IEEE Computer Society, 1997, pp. 1037–1044.

[2] S. Floyd and V. Paxson, "Difficulties in simulating the internet," *IEEE/ACM Trans. Netw.*, vol. 9, pp. 392–403, August 2001.

[3] M. Liljenstam, Y. Yuan, B. J. Premore, and D. Nicol, "A mixed abstraction level simulation model of large-scale internet worm infestations," in *Proceedings of the 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*, ser. MASCOTS '02. Washington, DC, USA: IEEE Computer Society, 2002, pp. 109–. [Online]. Available: http://portal.acm.org/citation.cfm?id=882460.882592

[4] M. Crovella, C. Lindemann, and M. Reiser, "Internet performance modeling: the state of the art at the turn of the century," *Performance Evaluation*, vol. 42, no. 2-3, pp. 91 – 108, 2000.

[5] S. Wei, J. Mirkovic, and M. Swany, "Distributed worm simulation with a realistic internet model," in *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*, ser. PADS '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 71–79.

[6] J. H. Cowie, D. M. Nicol, and A. T. Ogielski, "Modeling the global internet," *Computing in Science and Engineering*, vol. 1, pp. 42–50, 1999.

[7] H. Ringberg, M. Roughan, and J. Rexford, "The need for simulation in evaluating anomaly detectors," *SIGCOMM Comput. Commun. Rev.*, vol. 38, pp. 55–59, January 2008.

[8] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, "Internet mapping: From art to science," in *Proceedings of the 2009 Cybersecurity Applications & Technology Conference for Homeland Security*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 205–211.

[9] M. Luckie, "Scamper: a scalable and extensible packet prober for active measurement of the internet," in *Proceedings of the 10th annual conference on Internet measurement*, ser. IMC '10. New York, NY, USA: ACM, 2010, pp. 239–245.

[10] Y. Shavitt and E. Shir, "Dimes: let the internet measure itself," *SIGCOMM Comput. Commun. Rev.*, vol. 35, pp. 71–74, October 2005.

[11] R. N. C. Centre. The ripe atlas website. *Last Accessed 5 June 2011*. [Online]. Available: http://atlas.ripe.net/

[12] B. Donnet, B. Huffaker, T. Friedman, and K. Claffy, *NETWORKING 2007. Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2006, vol. 4268, ch. Evaluation of a Large-Scale Topology Discovery Algorithm, pp. 193–204.

[13] E. Katz-Bassett, H. V. Madhyastha, J. P. John, A. Krishnamurthy, D. Wetherall, and T. Anderson, "Studying black holes in the internet with hubble," in *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, ser. NSDI'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 247–262. [Online]. Available: http://portal.acm.org/citation.cfm?id=1387589.1387607

[14] S. McCanne and S. Floyd. ns–network simulator. *Last Accessed 5 June 2011*. [Online]. Available: http://www.isi.edu/nsnam/ns/

[15] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira, "Avoiding traceroute anomalies with paris traceroute," in *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, ser. IMC '06. New York, NY, USA: ACM, 2006, pp. 153–158.

[16] V. Paxson, "End-to-end routing behavior in the internet," *SIGCOMM Comput. Commun. Rev.*, vol. 36, pp. 41–56, October 2006.

[17] D. Abramson, J. Giddy, and L. Kotler, "High performance parametric modeling with nimrod/g: Killer application for the global grid?" in *Proceedings of the 14th International Symposium on Parallel and Distributed Processing*. Washington, DC, USA: IEEE Computer Society, 2000, pp. 520–. [Online]. Available: http://portal.acm.org/citation.cfm?id=846234.849304

[18] B. Donnet, personal communication, 2011.

[19] T. McGregor, "DoubleTree with many sources," in *ICIMP11: The Sixth International Conference on Internet Monitoring and Protection*. Sint Maartin Island, The Netherland Antilies: IARIA, March 2011.

Simulation and Sustainability

Enhancing Event-Discrete-Simulation Software with Sustainability Criteria

Andi H. Widok
Department of Engineering II
Industrial Environmental Informatics Unit
University of Applied Sciences, HTW Berlin
Berlin, Germany
a.widok@htw-berlin.de

Volker Wohlgemuth
Department of Engineering II
Industrial Environmental Informatics Unit
University of Applied Sciences, HTW Berlin
Berlin, Germany
volker.wohlgemuth@htw-berlin.de

***Abstract* — Nowadays global initiatives face numerous problems: non-transparent financial developments on the global markets, only a few years after the biggest economic crisis of our times, unsolved ecological problems that, given the ascent of emerging economies, are seemingly getting worse and the almost surreal speed at, which new technologies are changing our societies. The impacts these changes are having on companies worldwide are as numerous as their effects on the population. Sustainability and Sustainable Development have become key words in the hope of addressing and managing the changes that lay ahead of human societal development. This paper attempts to highlight shortcomings in the concept of sustainability and ways to make the concept more workable by presenting the development of an Environmental Management Information System (EMIS) as a combination of discrete event simulation and ecological material flow analysis for production processes.**

Keywords – Sustainability, Simulation, Event-Discrete-Simulation, Sustainability Reporting.

I. INTRODUCTION

In recent decades there has been a significant increase in the attention paid to the concept of sustainability. Despite this positive development, there is still only a small number of simulation systems that pay tribute to the complex interdependencies of economic, ecological and social values. This chapter will address the problems of current developments and therefore describe the motivation for the development of the EMIS that will be presented in the following chapters.

A. Ecological Perspective

From an ecological point of view, the world is facing a wide variety of problems. Even though there is still and most likely will continue to be a debate about how much and to what extent the effects of climate change are anthropogenic, the results themselves have been empirically proven and will consequently change the socio-economic requirements on earth within coming decades [1][2][3]. Effects such as the extinction of species [4][5], deforestation [6], changes in ice distribution [7], droughts and increasing incidence of forest fires or other effects, such as the development of CO_2 Emissions [1] or as the overfishing of the seas [8] will have a huge impact on the quality of life in the coming century.

In this respect, the figures presented by the Intergovernmental Panel on Climate Change (IPCC) in the year 2007 [1] made it very clear how pressing the need for a community-wide approach is for sustainable development in environmental protection. Despite phases with little or no economic growth, there is no expectation that the ecosystem will experience periods of natural recovery in coming decades, quite the opposite in fact. If one looks at current metabolic rates in the world [9], one finds that they are still rising [9][10]. The World Resources Forum (WRF) estimates that global resource extraction will exceed 80 billion tons in 2020. This means that mankind will have doubled the annual rate of global resource extraction within only 40 years (1980–2020) [11]. It further states:

“Globalizing the traditional model of economic growth is leading to rapidly increasing consumption of limited natural resources, followed by ecological disruption. (…) Rising global consumption of raw materials (…) is beginning to affect the life-sustaining services of the earth, which are not replaceable by technical means. (...) Today, the fundamental flaw in human activities is the enormous consumption of natural resources per unit output of value or service. (...) The environmental safety threshold has already been surpassed, as is evident” various “developments (…). And yet, only some 20 per cent of humankind enjoy the full benefits of the mainstream economic model, while all people – in particular the poor – have begun to suffer the consequences of its flaws” [11].

This statement can be translated into a system-thinking realization, that the behavior and interaction of system-elements are currently endangering the stability of the system itself. In this respect, one can consider various escalating curves, the result of catching-up processes involving emerging industries and countries such as China and India and the physical impossibility [10] of extending the present consumption patterns of the industrialized countries to all parts of the world, which will ultimately lead to social problems.

B. Social Perspective

Morally there is no argument as to why the developing countries and the poorest countries should not be “allowed” to achieve the same state of production and wealth that the industrialized countries have achieved. The only argument at this point, with our current technology, is that it is physically impossible and, even if it wouldn’t be, the level of production, given a world with 7 billion citizens, would result in catastrophic ecological consequences, if processes would be rushed.

When bearing in mind the ecological part of the problem, one can see that the system is out of balance due to excessive pressure on several fronts; the keyword in this sentence is balance since this is basically the common denominator for the social problem as well. What can be considered as unjust pressure on the ecological side would be translated as unequal distribution on the social side with results/effects such as hunger, lack of education or even terrorism.

Given the growing metabolic rates of the emerging countries one must be realistic and see that of course people will try to reach a similar state of wealth and prosperity and therefore, due to the impossibility mentioned, the only way to preserve the current or a current-comparable standard of living throughout industrialized countries, emerging countries and others, would be to find ways to reduce resource usage to an extent that would allow the same high level of production using only a fraction of the raw resources. In order to sustain our economy without completely revising our standard of living, the only way to achieve justice in distribution is the dematerialization of our economies and greater resource efficiency.

C. Economical Perspective

This transformation of economies would make investment imperative. The financial sector, however, is currently experiencing problems of its own. The financial crisis of 2007/08, the very recent developments surrounding the Euro (considering Greece and other European countries) and also surrounding the Dollar and the government/budget deficit in the United States ought to demonstrate how much mankind tends to worry about wealth and status and also how interdependent the global market already is today. The consequences of the crisis can be observed on the large scale already referred to, but also at the level of small and medium sized companies (SME), which are failing to obtain necessary loans from banks.

Basically it comes down to a similar problem of distribution as that, which applies to raw resources. If we consider a company as a minimal representation of an economy, we understand that with a purely economic orientation it will not lead to sustainable growth. A strong social commitment or intensive environmental management, however, will not have any positive effects if the company structure cannot bear the load they place on it either. Thus it is imperative that these three measures of sustainability are combined by means of balanced efforts leading to a synergistic increase in value [12][13][14].

This balance in efforts is what sustainability has been trying to define from the very outset. Throughout sustainability theory, from Meadows (1972) [15], Lynam, Herdt (1989) [16] and Pezzey (1992) (who already listed 27 different definitions for sustainability) [17], Pretty (1995) [18] to (Bell and Morse, 2008) [19], there has been a broad understanding that shrinking processes can also be considered sustainable. They all addressed the question of the objective that had to be protected / balanced.

For companies, the main priority and the most important commodity must naturally be the financial side; otherwise the company could neither exist nor produce. The community on the other hand has other interests when it considers this company. On the one hand, it is imperative that the company creates goods for public consumption, which bring in money, so that some of the public will obtain their income from it, but it is also necessary that the production methods do not harm the people or their environment. Allowing for the interests of the people, laws were drafted to make sure that the interest of companies does not take precedence over the interest of the people (legal compliance). In this context, there has always been a huge debate between Europe and the United States about regulation and deregulation. When it comes to ecological impacts however, this discussion seems misplaced as the market does not guarantee to reflect people’s interests when their perception is limited by manufactured prices or other artificial local regulations.

II. Sustainability and Simulation

In Section I.A we stated that sustainability addresses the problems of distribution, it therefore follows the ideals of intra- and intergenerational justice and is a conclusion of the realization that human actions have consequences, if not for themselves then for other people with a shift in space or time. Consequently, we understand sustainability as acceptance of this responsibility and therefore as a need to act without a shift in time.

A. Normative understanding of Sustainability

In view of the fact already explained, that it is not possible to have an equal distribution of goods, wealth, resources and products in the world within a short period of time (where short can be 50 years or more) and to preserve the ecosystem, it follows that the ideals of intra- and intergenerational justice cannot be satisfied at this point in time. Therefore the concept of sustainability must be regarded as the means to achieve intra- and intergenerational justice and is consequently normative.

B. A Definition of Sustainability

To ensure that there is a clear understanding of the following, we will define sustainability under a capital-based approach (similar to the one used by McElroy, Jorna, van Engelen [20]).

We define sustainability mainly as the agglomeration of actions/campaigns/processes that have a positive effect on the regeneration of social, environmental and/or economical capital on the one hand, and/or reduce the degradation of this capital on the other, bearing in mind that the protection of that capital is the normative goal.

A third option being the allowance of the use of a different source, or not having to use the capital in question at all any longer. An example of this would be the usage of new processes allowing the substitution of different materials insofar as the old material would no longer be needed for the process and the new material would be less of a drain on overall capital.

The main problem with this definition lies in the specification of what social/economic/environmental capital is. While no one will argue that it exists, one can argue about the concrete indicators and the attributed values behind them or, more specifically, about the value-correlations between them. This is also what makes it so difficult to define a sustainable process. While a process may be very ecologically sustainable when measuring the amount of material used, it may also be very expensive and therefore drain economic capital or vice versa.

The question is how the three aspects are correlated or rather, which indicators have been attributed to each of the aspects in the first place. For that matter we argue that environmental and indicators related to Corporate Social Responsibility (CSR-related) processes have been greatly undervalued in recent decades. While we do not believe that every single process in a company can or even should be broken down into a value, we believe it to be possible, to do this with many more processes than is the case at present and especially with more environmental and social processes. More than that, it is important to do so because most of the negative influences on environment and or people do in fact happen due to the lack of knowledge about correlations and impact scenarios.

Last but not least, one must take into account that consumption (a reducing effect on capital) may also be sustainable if natural or otherwise regenerating capital, which is of value in a certain quantity becomes a danger at a higher concentration. An escalating feedback-loop can therefore come from the capital at some tipping-point of its existence, which has to be managed. This would make an inversion of the signs imperative in order to achieve equilibrium between existence and effect of the capital.

While we realize that the definition of sustainability indicators is one of the most critical parts of sustainability assessment, this definition in conjunction with intended usage in simulation experiments allows many different approaches to be tested when assessing the sustainability-enhancing potential of intended measures. Thus simulation is a way to assess the sustainability of new processes.

C. *Simulation as a way to get closer to the immeasurable (Sustainability)*

Simulation can be used to show the possible effects of alternative conditions and courses of action. It is also used when the real system cannot be engaged, because it may not be accessible, or it may be dangerous or unacceptable to engage, or it is being designed but not yet built, or it may simply not exist [21]. In that regard simulations are perfect tools when it comes to experiment with uncertain outcomes, which may be harming or contra-productive.

As stated in Section I, we see one of the main challenges of our time in the dematerialization of the economy and consequently much higher resource efficiency. Under those premises the simulation focus had been laid on usage in production. The rational use of goods, such as the production, consumption and distribution is widely known as economic activity. Its improvement is directly connected to the in- and output relations and consists of the attempt to get more returns while investing lesser resources [22]. This process is also called optimization and it is target-oriented (e.g., optimizing the costs, quality, efficiency or effectiveness). Optimizations can also be achieved using an operations research approach [23]. The operations research approach and most analytic methods however become problematic once one has to deal with many variables. That is precisely when simulations are more worthwhile. The simulation of production addresses a variety of different indicators, the most common measures of system performance being the following [24]:

- Throughput under average and peak loads;
- System cycle time (how long it take to produce one part);
- Utilization of resource, labor, and machines;
- Bottlenecks and choke points;
- Queuing at work locations;
- Queuing and delays caused by material-handling devices and systems;
- Work in progress (WIP) storage needs;
- Staffing requirements;
- Effectiveness of scheduling systems;

These indicators can be considered as the standard value set of today’s production optimization, they however do not incorporate environmental or social indicators and hence an optimization of the production using those key-indicators would go only in one direction, leading to a higher output. Even though a higher production can of course have other positive effects, they are far from guaranteed. In the coming chapter we’ll illustrate the integration of the environmental perspective in the same model, which is used for simulation runs. In Section IV we’ll then propose our current vision on how to even integrate the social perspective in the simulation model and thus acknowledging all three pillars of sustainability.

III. THE ROAD SO FAR – DEVELOPMENT

A. *The earlier years*

The techniques of modeling and simulation have been established as an important instrument for the analysis and planning of complex systems in many domains [25].

The deduction from investigations around year 2000 was the proposal to use simulation techniques for supporting the application of the Material Flow Network method [26] [27].

Following that proposal, simulation can be used to calculate unknown environmental quantities. For example, it allows determining the necessary load of connected input flows considering complex systems [28].

In a sense, the material flow perspective is more general than the discrete event perspective [29]. Information is rarely linked to objects like products or process steps. Material Flow Networks, which were also developed at the University of Hamburg [30], are based on the Petri-Net theory.

During one of the latest research projects, the prototype modeling- and simulation software named MILAN was developed. On one hand, its discrete event simulation components allow an accurate analysis of typically economic aspects and industry related aspects, presented under point II.C, and on the other hand, its material flow analysis components did add for the first time an environmental perspective to the discrete event simulation model, i.e., a consideration of relevant material flows and transformations such as:

- consumption of commodities, resources and additives;
- energy demand;
- waste accumulation;
- Emission generation.

Discrete event simulations are a powerful method to represent production processes close to reality and to follow time intervals of different sizes from few hours up to several business years for investigating aspects depicted in the introduction. With the generation of pseudo-random numbers following given stochastic distributions natural variations such as varying inter-mediate arrival times of production jobs can be represented.

In 2006, we presented the first application of the Material Flow Simulator Milan [29], since then we intensified our work on different levels of the architecture and extensions of the simulation engine as elucidated in the next chapter.

B. Recent developments

The first implementation of MILAN was realized using the Delphi version of DESMO-J, called DESMO-D, the framework and components in high level language Delphi. The component-based architecture was realized using COM-Technology [28]. This realization however seemed outdated and was renewed since 2009 and MILAN was re-implemented.

The new development of the material flow simulator MILAN is based on the open-source plugin framework EMPINIA (http://www.empinia.org) (comparable to the Java framework Eclipse (http://www.eclipse.org)). EMPINIA, which was developed in the course of the EMPORER project, is designed for the development of complex domain-specific applications especially in the field of environmental management information systems (EMIS) [31]. It is a component-orientated extensible application framework based on Microsofts.NET (http://msdn.microsoft.com/de-de/netframework/default.aspx) technology with the purpose to support and simplify the development of complex software systems.

For MILAN it was necessary to provide libraries of simulation components (e.g., for production systems: machines, transporters, system boundaries), which enable the modeler to represent and simulate his system adequately. These components can be added to an application i.e., as building blocks via a plugin mechanism and thus can be used to build a user-specific model.

This implementation may lead to an easy development of user-specific components with low dependencies and an attachment to a modeling tool box for a certain application field, which is not possible with other simulation tools [32] [25]. These components can either be generally applicable or might be used for very specialized purpose. Specialized entities are developed for a whole production sector (e.g., semi-conductor sector with coater, stepper and dispatcher) [33][27] or they represent a production component of a certain company with its specific parameters. In contrast general components are highly abstracted and are applicable for many production systems [34]. The goal of this project was the development and implementation of such general entities for MILAN.

Another important gain resulting from the EMPORER research project was the implementation of very abstract simulation entities for the analysis of production systems. These entities enable users to model and simulate a broad set of production systems. Because of their modularity and the plugin mechanisms of EMPINIA it is very easy to add more specialized entities to the production system’s domain and to use them for a material flow simulation.

After that the production components were verified by performing a simulation study in a company that produces solar panels. The problems, results and experiences of this validation were used to improve and enhance the components, the simulation infrastructure and MILAN as a simulation tool, itself.

Besides the components, which come with EMPINIA there are many plugins taken from a designed EMIS toolbox and were then combined with MILAN. The simulation capabilities of the MILAN software consist of the simulation core, a bundle for discrete event simulation and simulation components.

The simulation core consists of the central simulation service, interfaces and abstract base classes for models, experiments and model entities. These are used in each kind of simulation. The simulation service provides models and experiments in a way that other software parts can use them. The simulation core gives models and their entities access to the functionality of a domain model service. A domain model defines the domain of an EMPINIA-based application, its elements and their relations as well as rules that apply to this domain. MILAN consists of the domain 'simulation' with elements like 'model' and 'entity'. Among other important functionalities the domain service provides possibilities to persist its elements. That is the reason why this service is used in MILAN to save and load formerly created models.

A bundle for discrete event simulation extends the simulation core with classes specific to the discrete event simulation approach. These classes are using an EMPINIA extension that enables the development of logical graphs in order to combine entities of a model to a network diagram. The basic generic experiment component is extended with an event list and a scheduler, which are used to simulate time in discrete steps.

The simulation components have access to many stochastic distributions (e.g., Normal, Bernoulli). They are used to generate streams of random numbers, for example to schedule an event, which follows a certain arrival probability. Additional to these existing distributions user-defined distributions can also be added via plugins.

In the following the common features of the MILAN software will be summarized.

The graphical manipulation of building blocks leads to a faster development of a model. The graph editor can be used to manipulate and create models. The editor itself can work in different domains. Domain specific functionality and the graphical representation have to be defined by plugin developers enabling the editor to handle new domains and their components, which are also using plugin definitions.

Manipulating model parameter for the simulation and material flow perspective is done by means of property editors enabling a simple and consistent way of setting values for all types of properties. For the production system domain there are standard editors implemented. These allow the change of component specific parameters like setting distributions, accounting rules, queue lengths or capacities etc.

No analysis can be done without results. These are shown in reports, which can be designed with the help of the reporting system. The data for the reports is aggregated during simulation runs by a system of observers that listen to changes in the material accounting and simulation entities.

The development of new features and the testing of the full capacity of MILAN's functionality are ongoing. The Combination of economical and ecological indicators in one model has already been achieved. In the following chapter we'll outline visions on how MILAN might get even closer to a sustainability enhancing simulation system.

IV. Conclusion and Future Work

In an often cited interview the Nobel Prize Winner Milton Friedman said: "So the question is, do corporate executives, provided they stay within the law, have responsibilities in their business activities other than to make as much money for their stockholders as possible? And my answer to that is no, they do not" (February 1974) [35].

Even if one would tend to agree with Friedman, there are already examples of when and how this statement would be economically disadvantageous, considering Nike and their incident with child labor in their supply chain [35][36] or the case of Brent Spar and their sinking of an oil platform [35][37], which made obvious that the long term goal of profit maximization can only be achieved when parts of the social responsibility are also acknowledged [35][37][38][39]. In case of Nike, the sales figures dropped after the incident, resulting to a stock loss of 20 per cent [35][39][40][41]. The connection is already there.

Also the range of management approaches that look at social sustainability is relatively vast, so that one faces an unmanageable diversity of what are referred to as 'solutions'. There are however not many software solutions that pick up on social aspects and where they do their usage is rather infrequent. This fact alone narrows down the search for universal applications, but also opens another perspective on the much more discussed "opposition" between the achievement of economical and ecological objectives [42].

To make companies realize that they must aim not merely for financial stability, it is mandatory that corporate social responsibility (CSR) and environmental efforts become a financial attribute and thus have an economic value too. The lack of these values, or rather their unspecific nature in the past, has led to many of today's undesirable developments, as profit is often solely attributed with financial growth while social, human, environmental profit is only of relevance when it comes to legal compliance [42].

In that regard current research at the HTW Berlin also tries to incorporate social indicators for the assessment of sustainable growth in production. Through the EMPINIA extension mechanism it is possible to define new resources, in this regard, human resources. These resources are then getting attributes, such as, for example, workload/contract information and references to the workstations, these references are basically the skills of the current employees. In order to pay tribute to the different abilities of the employees the workstations/building blocks themselves are more or less in dependence of human resources to function properly and the human resources have a variety of criteria that, for example inhibits them to work 30 hour shifts. There is a whole framework of social criteria possible to be attributed to these new "resources"; however research is still on its very beginning. The first focus of the introduction of social criteria will be health. Employees should not work longer then a certain amount of time; they should have the possibility to take all their vacation and should not get in contact with any harming emissions, noise, particular matter or other harming material. Even though that does not sound revolutionizing it is the first step in addressing more complex interactions, such as financial equilibrium, daycare for children or other criteria.

We hope that in the future, after testing the introduction thoroughly, we can implement more criteria and define new functions of correlation and interdependencies. In this paper we tried to give further input to the ongoing discussion on how to assess sustainability and more precisely the sustainability of producing companies. We tried to show in the introduction that no matter, which pillar of sustainability is considered the negative influence, the loose ends, are likely to be a result of a system-imbalance. They are the underlying conditions for most of the problems we face today. We also tried to show that the change of human economies will become imperative and must be managed in a way that intends to address the issue of participation, which we consider to be one of the main problems of the sustainability dilemma. People and companies, as system-elements will not intensify their positive influence unless the instability of the system is made obvious to them. The combination of different perspectives of sustainability in one model might contribute to this thesis and will therefore be our ongoing focus in the future.

References

[1] IPCC (Intergovernmental Panel on Climate Change), 2007, IPCC Assessment Report

[2] Rottke, N.B., 2009, Ökonomie versus Ökologie – Nachhaltigkeit in der Immobilienwirtschaft?, Köln

[3] Benz, G., 2009, Naturkatastrophen sind Kulturkatasstrophen! Umwelthistorische Grundlagen von Riosikoanalysen für Naturgefahren, article published in: Beiträge zum Göttinger Umwelthistorischen Kolloquium, Göttingen

[4] Butchart, S.H.M. et al., 2004, Measuring Global Trends in the Status of Biodiversity: Red List Indices for Birds, PLoS Biol 2(12): e383. doi:10.1371/journal.pbio.0020383, 2004

[5] Butchart, S.H.M. et al., 2007, Improvements to the Red List Index, PLoS ONE 2(1): e140. doi:10.1371/journal.pone. 0000140

[6] Corbera, E. Estrada, M., and Brown, K., 2010, Reducing greenhouse gas emissions from deforestation and forest degradation in developing countries: revisiting the assumptions

[7] Mallory, M.L., Gaston, A.J. Gilchrist, H.G. Robertson, G.J. Braune, B.M., 2010, Effects of Climate Change, Altered Sea-Ice Distribution and Seasonal Phenology on Marine Birds, from A Little Less Arctic: Top Predators in the World's Largest Northern Inland Sea

[8] FAO (Food and Agriculture Organization of the United Nations), 2007, The State of world fisheries and aquaculture 2006, Rome

[9] OECD (Organization for Economic Cooperation and Development) 2008, Measuring material flows and resource productivity, Synthesis Report, Paris

[10] Hilty, L.M. and Ruddy, T.F., 2010, Sustainable Development and ICT interpreted in a natural science context, Information, Communication & Society, 13: p. 1, 7 — 22

[11] WRF (World Resources Forum), Draft Declaration of the World Resources Forum, 16 October 2008, p. 1

[12] Stahlmann, V., 2008, Lernziel: Ökonomie der Nachhaltigkeit, München

[13] Schmidt-Bleek, F., 2008, Nutzen wir die Erde richtig?: Von der Notwendigkeit einer neuen industriellen Revolution, Frankfurt

[14] von Pappenheim, J. R., 2009, Das Prinzip Verantwortung, Gabler, GWV Fachverlage GmbH, Wiesbaden

[15] Meadows et. al., 1972, The Limits to Growth, Report, Club of Rome

[16] Lynam, J. K. and Herdt, R. W., 1989, „Sense and sustainability: Sustainability as an objective in international agricultural research", Agricultural Economics, volume 3, n° 4, p. 381 – 398

[17] Pezzey, J, 1992, "Sustainability: An Interdisciplinary Guide." Environmental Values 1, 1992, p. 321 - 362

[18] Pretty, J., 1995, Participatory learning for sustainable agriculture World Development, Vol. 23 No.8, 1995, p.1247-1263, World Development, 8th edition

[19] Bell, S. and Morse, S., 2008, Sustainability Indicators: Measuring the Immeasurable? Sterling, Second Edition

[20] McElroy, M.W. Jorna, J.R., and van Engelen, J., 2007, Sustainability Quotients and the Social Footprint published in Corporate Social Responsibility and Environmental Management, John Wiley and Sons Ltd and The European Research Press Ltd

[21] Sokolowski, J.A. and Banks, C.M., 2009, Principles of Modeling and Simulation. Hoboken, NJ: Wiley. p. 6. ISBN 978-0-470-28943-3.

[22] Wöhe, G. and Döring, U. 2008, Einführung in die Betriebswirtschaftslehre. 23th edition. Vahlen, München.

[23] Domschke, W. and Drexl, A. 2007, Einführung in Operations Research. 7th edition. Springer, Berlin.

[24] Banks, J., Carson J., Nelson B.L., Nicol, D., 2005, Discrete-event system simulation (4th ed.). Upper Saddle River, NJ: Pearson Prentice Hall. ISBN 978-0-13-088702-3.

[25] Page, B. and Kreutzer, W. 2005, The Java Simulation Handbook: Simulating Discrete Event System with UML and Java. Shaker Verlag, Aachen.

[26] Wohlgemuth, V., Bruns, L., and Page, B., 2001, Simulation als Ansatz zur ökologischen und ökonomischen Planungsunterstützung im Kontext betrieblicher Umweltinformationssysteme (BUIS), in: Hilty, L.M., Gilgen, P.W. (Hrsg.): Sustainability in the Information Society. 15. Internationales Symposium "Informatik für den Umweltschutz" der Gesellschaft für Informatik (GI), Zürich 2001. Metropolis Verlag, Marburg, Band 2, p. 999-1008.

[27] Wohlgemuth, V. 2005, Komponentenbasierte Unterstützung von Methoden der Modellbildung und Simulation im Einsatzkontext des betrieblichen Umweltschutzes. University of Hamburg: Thesis (PhD). Aachen, Shaker Verlag.

[28] Joschko, P., Page, B., and Wohlgemuth, V., 2009, Combination of Job Oriented Simulation with Ecological Material Flow Analysis as integrated Analysis Tool for Business Production Processes, Proceedings of the 2009 Winter Simulation Conference

[29] Wohlgemuth, V., Page, B., and Kreutzer, W., 2006, Combining discrete event simulation and material flow analysis in a component-based approach to industrial environmental protection. Environmental Modelling & Software, p. 1607-1617.

[30] Möller, A. 2000, Grundlagen stoffstrombasierter betrieblicher Umweltinformationssysteme. Projekt-Verlag, Bochum.

[31] Wohlgemuth, V., Schnackenbeck, T., Panic, D., and Barling, R.-L. 2008, Development of an Open Source Software Framework as a Basis for Implementing Plugin-Based Environmental Management Information Systems (EMIS). In: Möller, A.; Page, B.; Schreiber, M. (Eds.): EnviroInfo 2008. Proceedings of the 22nd International Conference Environmental Informatics - Informatics for Environmental Protection, Sustainable Development and Risk Management, September 10-12, 2008, Leuphana University Lüneburg. Shaker Verlag, Aachen, p. 584-592

[32] Page, B., Lechler, T., and Claassen, S. 2000, Objektorientierte Simulation in Java mit dem Framework Desmo-J. Libri Books,

[33] Wohlgemuth, V., Page, B., Mäusbacher, M., and Staudt-Fischbach, P. 2004, Component-Based Integration of Discrete Event Simulation and Material Flow Analysis for Industrial Environmental Protection: A Case Study in Wafer Production, in: Proceedings of the 18th International Conference for Environmental Protection, October 21-23, CERN, Geneva, p. 303-312

[34] Jahr, P., Schiemann, L., and Wohlgemuth V. 2009, Development of simulation components for material flow simulation of production systems based on the plugin architecture framework EMPINIA. In: Wittmann, J.; Flechsig, M. (Eds.): Simulation in Umwelt- und Geowissenschaften. Shaker Verlag, Aachen, p. 57-69

[35] Dubielzig, F., 2009, Sozio Controlling in Unternehmen, Das Management erfolgsrelevanter sozial-gesellschaftlicher Themen in der Praxis, Gabler Edition Wissenschaft, Dissertation Leuphana Universität Lüneburg, 2008, First edition

[36] Insight Investment 2003, Labour Standards and Working Conditions in Supply Chains. Downloaded from www.insightinvest-ment.com/documents/responsibility/ir_labour.pdf (28th Mai 2008)

[37] Mantow, W. 1995, Die Ereignisse um Brent Spar in Deutschland: Darstellung und Dokumentation mit Daten und Fakten; Hintergründe und Einflussfaktoren; Kommentare und Medienresonanzen. Hamburg: Deutsche Shell AG.

[38] Bakan, J. 2005, Das Ende der Konzerne. Die selbstzerstörerische Kraft der Unternehmen. Hamburg, Europa Verlag.

[39] Mintzberg, H., 1983, The Case for Corporate Social Responsibility, The Journal of Business Strategy, 4 (2), p. 3-15.

[40] Murray, S., 2002, The Supply Chain. Working Lives under Scrutiny. Downloaded under www.theglobalalliance.org/documents/ FinancialTimes1202article_000.pdf (20th Mars 2005)

[41] Leitschuh-Fecht, H. and Bergius, S., 2007, Stakeholderdialoge können besser werden, UmweltWirtschaftsForum, 15 (1), p. 3-6.

[42] Colantonio, A., 2009, Social Sustainability, ed. Oxford Institute for Sustainable Development Downloaded under: http://Fec.europa.eu/ research/sd/conference/2009/presentations/7/andrea_colantonio_-_social_sustainability.ppt, (15th May 2011)

Towards a SDL-DEVS Simulator

Multiparadigm simulation

Pau Fonseca i Casas
Statistics and Operations Research Department
Universitat Politècnica de Catalunya
Barcelona, Catalunya, Spain
pau@fib.upc.edu

Josep Casanovas
Statistics and Operations Research Department
Universitat Politècnica de Catalunya
Barcelona, Catalunya, Spain
josepk@fib.upc.edu

Abstract— **In this paper, we present the first version of a simulator that allows executing models defined using Discrete Event System Specification and models defined using Specification and Description Language. Specification and Description Language (SDL) is a graphical language, standardized under the ITU Z.100 recommendation, widely used to represent telecommunication systems, process control and real-time applications in general. Discrete Event System Specification (DEVS) is a formalism widely used on the simulation field to represent Discrete Event Systems. The execution of the DEVS models is based on a transformation of the simulation model DEVS representation to an equivalent SDL representation. To do this, we propose a XML representation for the DEVS models, and a XML representation for SDL models. Also, we implement an algorithm capable to perform this transformation.**

Keywords-simulation; formal language; SDL; DEVS

I. INTRODUCTION

The purpose of the paper is to present a simulator capable to understand and use SDL, or DEVS language, in a single simulation model. Several simulators capable to understand DEVS language exist, like DEVS++ [1], CD++ [2] or Galatea [3] among others; also, several tools work with SDL, like Cinderella [4] or IBM's Tau Telelogic [5]. However, currently, there is no simulator capable to work with both languages. This capability improves the reusability of models and the combination of technologies in a single framework. The underline idea is to enable the use of several models in a bigger and detailed model composed by those models. Also, those models can be defined using different formal languages. In this paper, not only a simulator able to understand both languages is presented, but also a method that enables the translation from DEVS models to SDL, based on a proposed XML representation of DEVS models.

This paper is organized as follows: first, we review both languages. Next, we present how we can describe both languages using XML, proposing a new representation for atomic DEVS models. From this representation and thanks to an algorithm that allows the transformation from DEVS to SDL, we present a mechanism that allows performing this transformation automatically. Lastly, we present a system that is capable of perform a simulation using DEVS or SDL models.

II. SPECIFICATION AND DESCRIPTION LANGUAGE

Specification and Description Language (SDL) is an object-oriented, formal language defined by the International Telecommunication Union – Telecommunication Standardization Sector (ITU–T). The recommendation that summarizes its use is Z.100. The language is designed to specify complex, event-driven, real-time, interactive applications involving many concurrent activities using discrete signals to enable communication [6]. The definition of the model is based on different components:

- Structure: system, blocks, processes and processes hierarchy.
- Communication: signals, with the parameters and channels that the signals use to travel.
- Behavior: defined through the different processes and procedures.
- Data: based on Abstract Data Types (ADT).
- Inheritances: to describe the relationships between, and specialization of, the model elements.

The language has 4 levels (Figure 1):

1. System.
2. Blocks.
3. Processes
4. Procedures.

To know more about the Specification and Description Language, [6][7][8] can be consulted.

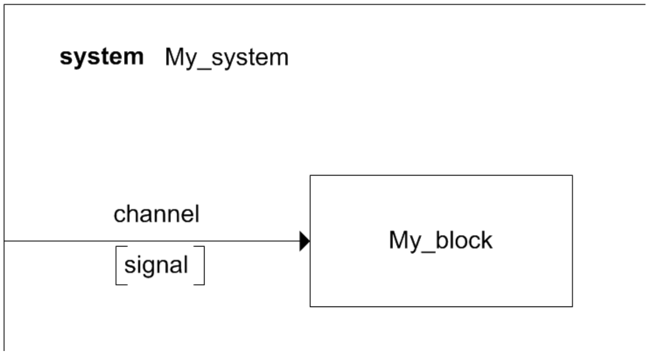
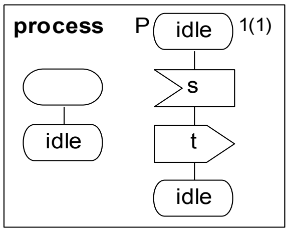


Figure 1. The figure shows the first level of an SDL model. Here, a system named “My_system” is shown. It is composed by a single block “My_block”, who receives a signal named “signal” from the environment through the channel named “channel”.

III. SDL REPRESENTATIONS

SDL have two ways to be represented, SDL PR and SDL GR. SDL-PR is conceived to be easily processed by computers, also allows a compact representation of a model. SDL-GR has some textual elements which are identical to SDL-PR (this is to allow specification of data and signals) but it is mainly graphical.

Figure 2 shows an example of a textual and graphical representation of an SDL process. We are not using the textual version of SDL only for one reason. Some different textual representations of DEVS based on XML format exist. Since we want to allow an automatic transformation from SDL to DEVS, the use of XML simplifies our programming code because now is easy to read and write structured text files that follow the XML syntax, and also, thanks to the XSD we can validate the correctness of its syntax. We are using the XML representation for SDL proposed in [9]. Since the more important aspects of an XML file can be represented, and validated, through an XSD file, in the next section some areas of the XSD file are shown.



```
process P;
 start;
 nextstate idle;
 state idle;
  input s;
   output t;
   nextstate idle;
 endstate idle;
endprocess P;
```

Figure 2. Textual and graphical SDL representation.

A. XML representation of an SDL simulation model

This representation was first presented on [10], no modifications have been done from this schema. We next describe the more important elements. For further details, please see [10], or download the complete schema from [11].

In Figure 3, we show the first level of the XSD schema we use to validate the structure of our XML. The first level of this schema represents the first level of the Specification and Description Language (system outmost block). Figure 4 shows the process *type* that allows represent an SDL process.

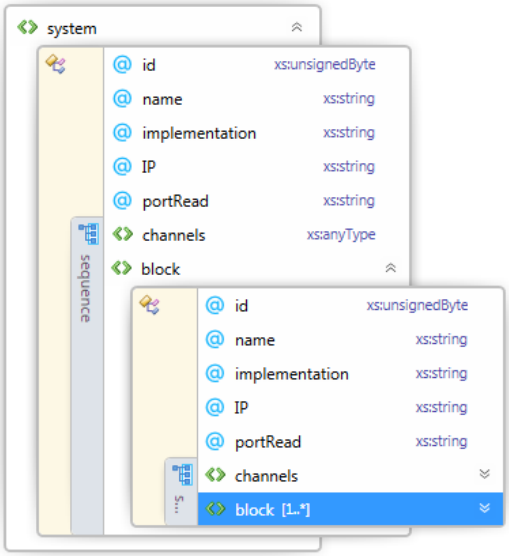


Figure 3. XSD schema, *system* view

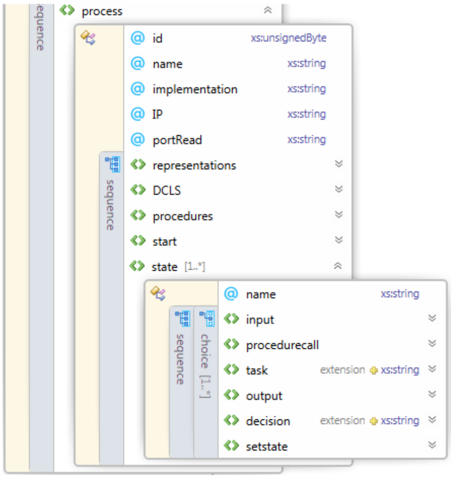


Figure 4. XSD schema, *process* view

IV. DEVS FORMALISM

Proposed by Bernard Zeigler in the 70’s [12], the main scope of Discrete Event System Specification (DEVS) is the

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-169-4

representation of simulation models. A DEVS model is a tuple composed by the elements defined as follows:

$$
\begin{aligned}
&M =< Z, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta > \boldsymbol{where},\\
&X: set\ of\ input\ values\\
&S: set\ of\ state\ values\\
&Y: set\ of\ output\ values\\
&\delta_{int}: internal\ transition\ function;\ \delta_{int}: S \to S\\
&\delta_{ext}: external\ transition\ function;\ \delta_{ext} \to Q\ \times\ X\ \to S\\
&Q = \{(s,e) | s \epsilon S, 0 \le e \le ta(s)\}: set\ of\ states\\
&e: time\ from\ the\ last\ transition\\
&\lambda: S \to Y: output\ function\\
&ta: time\ advance\ function\\
&ta: S \to R_0^+
\end{aligned}
$$

DEVS distinguishes between an internal and external transition. An internal transition is a kind of transition that doesn't need any external event to be launched. As an example, if in a "t" time, the system reach the state "s", the system remains in this state the during the time defined on a "time advance" function "ta(s)" (if no external event is received). When the time reach the value defined in the "ta(s)" function an output event is produced (this output is defined on the "λ(s)" function) and the state changes to "s' ". This process is defined in the internal transition s'= δ_{int}(s).

External transitions define the modifications in the model due to the reception of external events. For example, before the model reach the state "s' ", in a time "t", due to his internal transition, an external event, with value x, is processed. In this case the system reach state (s,e) where e<ta(s), the transition follows the external transition function, defined by s'= δ_{ext}(s,e,x), and no exit event is produced.

At this point, it is important to underline that "ta(s)" could be any real number, plus 0 and ∞, and:

- If ta(s) is 0, "s" is a *transitory* state.
- If ta(s)=∞, "s" is a *passive* state.

In the next lines, we review two examples from [12]. We use these two models to transform them automatically to a SDL specification and then execute the models using SDLPS [9].

A. Processor example

This example represents a single processor that receives different jobs. Each job has associated a processing time (represented by a real number). Once the time is over event "ready" is produced. When a new event reach the processor, if this is working with a job, this event is ignored. The DEVS formalization of this model is:

$$
\begin{aligned}
&M =< X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta > \boldsymbol{where},\\
&X = \{job_1\ , job_2, .., job_n\}\\
&S = \{job_1\ , job_2, .., job_n\} \cup [\emptyset] \times R^+\\
&Y = \{y(job_1), y(job_2), .., y(job_n)\}\\
&\delta_{int}(job, \sigma) = (\emptyset, \infty)\\
&\delta_{ext}(job, \sigma, e, x) = \begin{cases} (x, tp(x) if\ job = \emptyset \\ (job, \sigma - e) otherwise \end{cases}\\
&\lambda(job, \sigma) = y(job)\\
&ta(job, \sigma) = \sigma
\end{aligned}
$$

B. FIFO Queue example

The queue represented in this example has the following characteristics:

- The queue has infinite capacity.
- Different jobs reach the queue to be stored, while the "ready" signals symbolize the necessity of transmit the first job of the queue.
- The transmission of this job is done through an output event.
- The queue spends 0 time units in the exit delay.

The DEVS model is:

$$
\begin{aligned}
&M =< X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta > \boldsymbol{where},\\
&X = \{job_1\ , job_2, .., job_n\} \cup \{'ready'\}\\
&S = \{job_1\ , job_2, .., job_n\} \cup [\emptyset] \times R^+\\
&Y = \{y(job_1), y(job_2), .., y(job_n)\}\\
&\delta_{int}(q \cdot job, \sigma) = (q, \infty)\\
&\delta_{ext}(job, \sigma, e, x) = \begin{cases} (x \cdot q, \infty) if\ x \in J \\ (q, 0) otherwise \end{cases}\\
&\lambda(q \cdot job, \sigma) = job\\
&ta(job, \sigma) = \sigma
\end{aligned}
$$

V. DEVS COUPLED MODELS

DEVS also allows formalizing simulation models without describing the behavior for each element belonging the model. It is possible to describe the structural relations that exist among identical elements. These models are named "coupled models". In DEVS there are two main types of coupled models:

- Modular coupling.
- Non modular coupling.

In modular coupling integration among different model components happens only across entries and exits defined in the components, while in non-modular coupling, interaction is produced across states. The literature established that is possible to pass from one kind of coupling model to the other [5], therefore in present paper we will focus on show the existing relation among SDL formalism and the DEVS modular formalism.

For simplicity, the DEVS coupled model used in this paper is DEVS coupled model with ports. In this model a series of input and output ports are described. With this logic is possible to depict the following example, see Figure 5, representing the combination of the two models that have been defined previously (the queue and the processor).

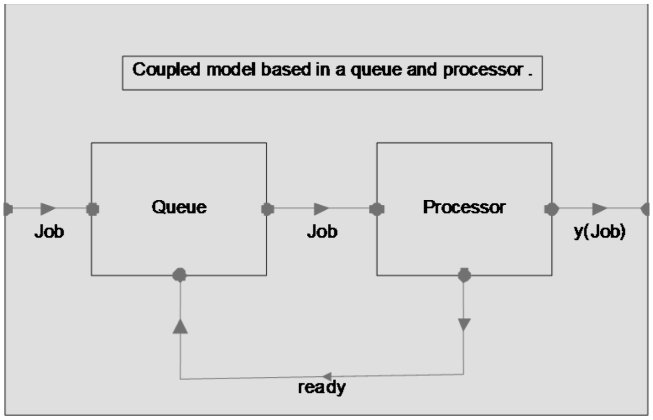


Figure 5. DEVS coupled model.

The coupling model specification for this model is:

N = (X, Y, D, {Md | d Î D}, EIC, EOC, IC, Select), on
X=Jx{inport1}
Y={y(Job) | Job є J } x { outport1}
D={P,Q}
EIC{(N, inport1), (Q, inport1)}
EOC{(P,outport1), (N, outport1)}
IC{(P, outport2), (Q, outport2)}

VI. XML REPRESENTATION OF DEVS MODELS

Some attempts have been made to represent DEVS models using XML. As an example, in [13], a schema is presented that cannot characterize the programming logic, loops and if-then-else constructs. Our approach is going further and allows the representation of those elements. We propose to use ANSI C (since it is an ISO standard) to represent the code contained in model. Also this simplifies the representation of the model on SDL, using a variant named SDL-RT who uses ANSI C too. In our point of view the DEVS-XML representation that we present here can be considered as a good starting point for a robust and complete representation of DEVS models using XML.

We follow some conventions to represent a DEVS model using XML syntax:

- All the code needed to fully define the simulation model is defined on the “values” xml section.
- The initial conditions of the model is defined in the XML as well, using a ”value” attribute related to all the variables that defines the state of an atomic DEVS model.
- Also, to represent the value ∞ used in the passive states we use ‘inf’ literal value.

Some parts of the XML schema used to represent coupled and an atomic models is shown in Figure 6.

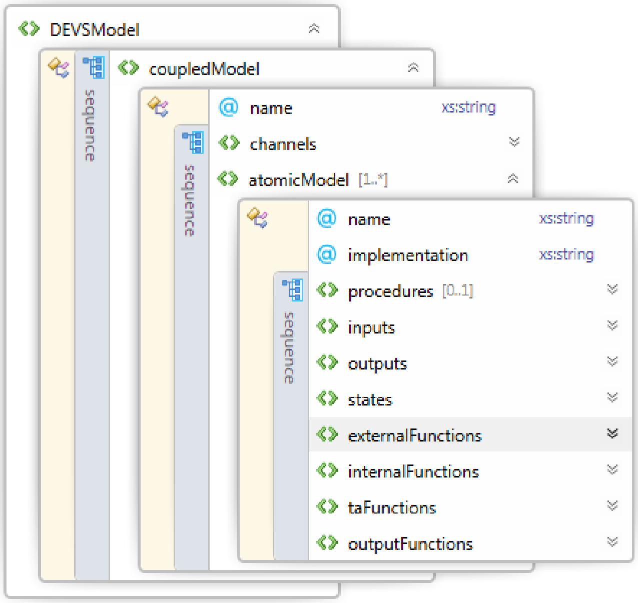
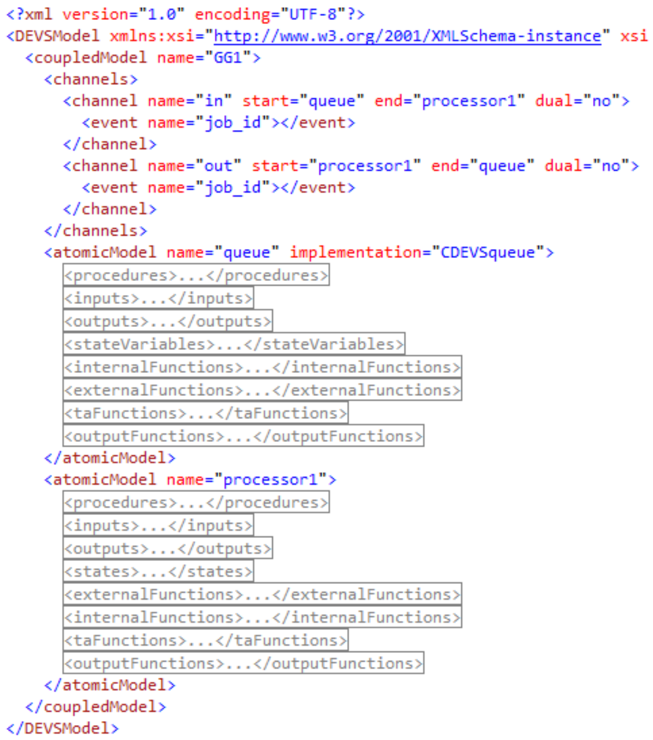


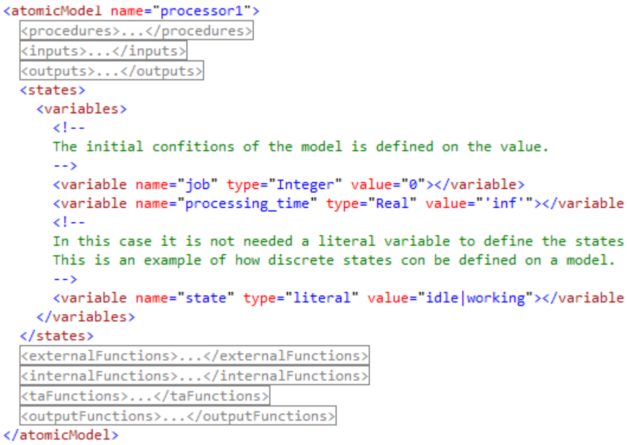
Figure 6. DEVS XML schema.

The complete definition of the *DEVSmodel* using XML is show next. In Figure 7 is represented the whole DEVS model using XML. In Figure 8 the definition of the *states* is shown. Figure 9 shows the definition of the *input* and the *output* elements. Figure 10 represents the *external functions* and in Figure 11 the *time advance* and *output functions*.



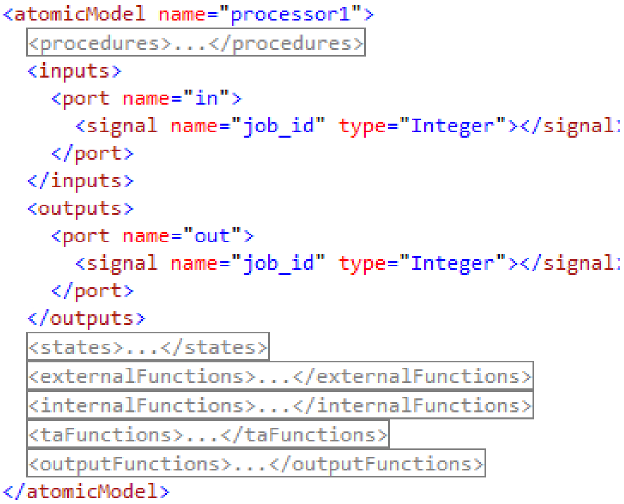
```
<?xml version="1.0" encoding="UTF-8"?>
<DEVSModel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi
  <coupledModel name="GG1">
    <channels>
      <channel name="in" start="queue" end="processor1" dual="no">
        <event name="job_id"></event>
      </channel>
      <channel name="out" start="processor1" end="queue" dual="no">
        <event name="job_id"></event>
      </channel>
    </channels>
    <atomicModel name="queue" implementation="CDEVSqueue">
      <procedures>...</procedures>
      <inputs>...</inputs>
      <outputs>...</outputs>
      <stateVariables>...</stateVariables>
      <internalFunctions>...</internalFunctions>
      <externalFunctions>...</externalFunctions>
      <taFunctions>...</taFunctions>
      <outputFunctions>...</outputFunctions>
    </atomicModel>
    <atomicModel name="processor1">
      <procedures>...</procedures>
      <inputs>...</inputs>
      <outputs>...</outputs>
      <states>...</states>
      <externalFunctions>...</externalFunctions>
      <internalFunctions>...</internalFunctions>
      <taFunctions>...</taFunctions>
      <outputFunctions>...</outputFunctions>
    </atomicModel>
  </coupledModel>
</DEVSModel>
```

Figure 7. GG1 DEVS model.



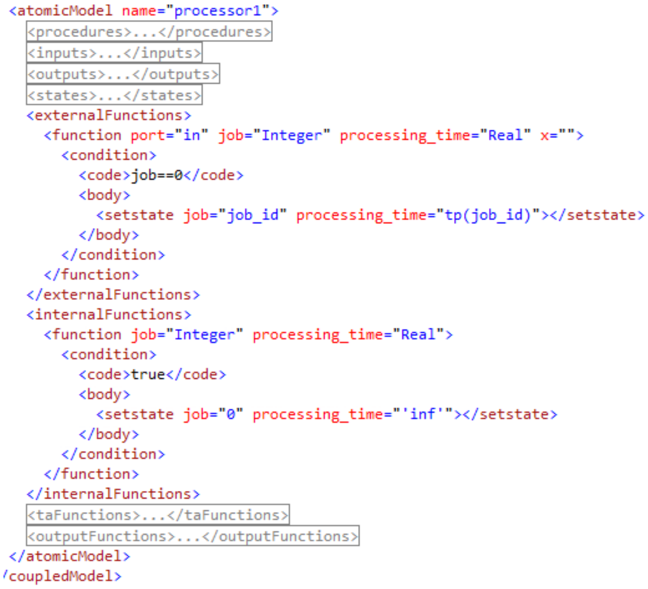
```
<atomicModel name="processor1">
  <procedures>...</procedures>
  <inputs>...</inputs>
  <outputs>...</outputs>
  <states>
    <variables>
      <!--
      The initial confitions of the model is defined on the value.
      -->
      <variable name="job" type="Integer" value="0"></variable>
      <variable name="processing_time" type="Real" value="'inf'"></variable
      <!--
      In this case it is not needed a literal variable to define the states
      This is an example of how discrete states con be defined on a model.
      -->
      <variable name="state" type="literal" value="idle|working"></variable
    </variables>
  </states>
  <externalFunctions>...</externalFunctions>
  <internalFunctions>...</internalFunctions>
  <taFunctions>...</taFunctions>
  <outputFunctions>...</outputFunctions>
</atomicModel>
```

Figure 8. States definition.



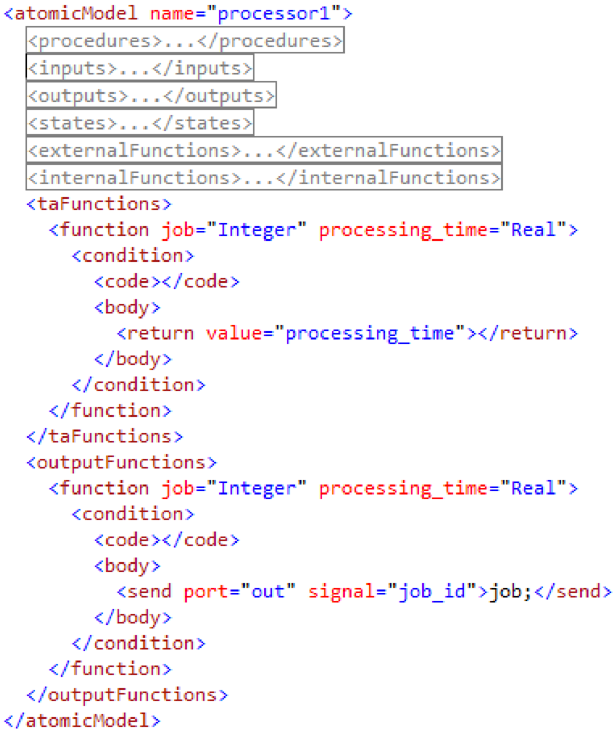
```
<atomicModel name="processor1">
  <procedures>...</procedures>
  <inputs>
    <port name="in">
      <signal name="job_id" type="Integer"></signal
    </port>
  </inputs>
  <outputs>
    <port name="out">
      <signal name="job_id" type="Integer"></signal
    </port>
  </outputs>
  <states>...</states>
  <externalFunctions>...</externalFunctions>
  <internalFunctions>...</internalFunctions>
  <taFunctions>...</taFunctions>
  <outputFunctions>...</outputFunctions>
</atomicModel>
```

Figure 9. Input and output elements.



```
<atomicModel name="processor1">
  <procedures>...</procedures>
  <inputs>...</inputs>
  <outputs>...</outputs>
  <states>...</states>
  <externalFunctions>
    <function port="in" job="Integer" processing_time="Real" x="">
      <condition>
        <code>job==0</code>
        <body>
          <setstate job="job_id" processing_time="tp(job_id)"></setstate>
        </body>
      </condition>
    </function>
  </externalFunctions>
  <internalFunctions>
    <function job="Integer" processing_time="Real">
      <condition>
        <code>true</code>
        <body>
          <setstate job="0" processing_time="'inf'"></setstate>
        </body>
      </condition>
    </function>
  </internalFunctions>
  <taFunctions>...</taFunctions>
  <outputFunctions>...</outputFunctions>
</atomicModel>
/coupledModel>
```

Figure 10. External an internal functions.



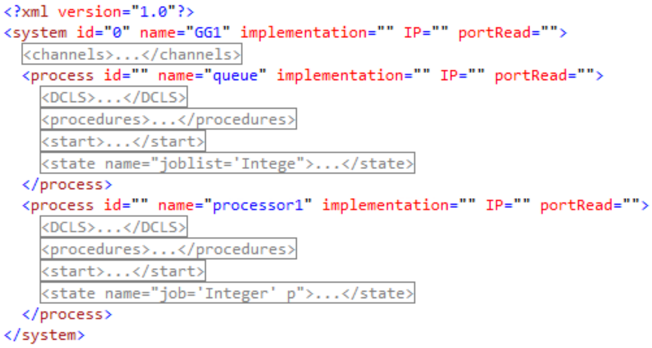
```
<atomicModel name="processor1">
  <procedures>...</procedures>
  <inputs>...</inputs>
  <outputs>...</outputs>
  <states>...</states>
  <externalFunctions>...</externalFunctions>
  <internalFunctions>...</internalFunctions>
  <taFunctions>
    <function job="Integer" processing_time="Real">
      <condition>
        <code></code>
        <body>
          <return value="processing_time"></return>
        </body>
      </condition>
    </function>
  </taFunctions>
  <outputFunctions>
    <function job="Integer" processing_time="Real">
      <condition>
        <code></code>
        <body>
          <send port="out" signal="job_id">job;</send>
        </body>
      </condition>
    </function>
  </outputFunctions>
</atomicModel>
```

Figure 11. Time advance and output functions.

From this DEVS-XML representation, we can obtain an equivalent model described using Specification and Description Language, using again XML (SDL-XML).

VII. TRANSFORMING FROM DEVS TO SDL

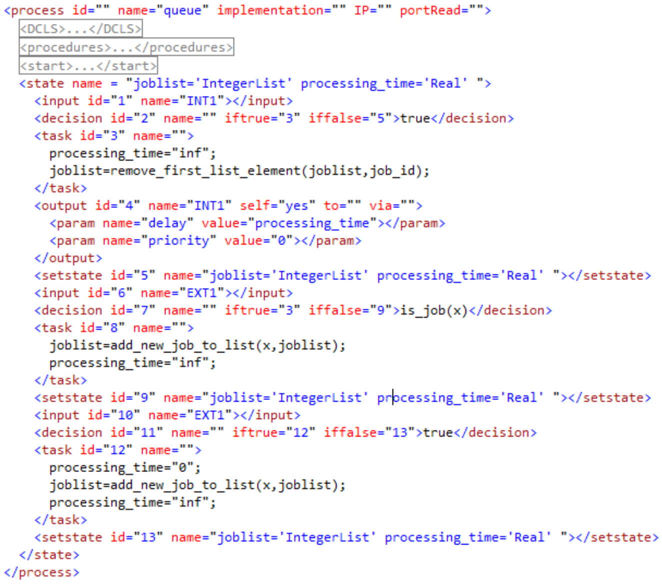
The transformation algorithm is based on the theoretical proposal presented in [14]. In this infrastructure, we implement this proposal using the XML representation for the SDL and DEVS model (DEVS-XML and SDL-XML). This allows us to obtain a new XML file that represents a DEVS model. The schema used here to represent the SDL model is based on those presented on [10] we only show here the more important aspects of the resulting XML file that represents the new proposal for the DEVS-XML representation.



```
<?xml version="1.0"?>
<system id="0" name="GG1" implementation="" IP="" portRead="">
  <channels>...</channels>
  <process id="" name="queue" implementation="" IP="" portRead="">
    <DCLS>...</DCLS>
    <procedures>...</procedures>
    <start>...</start>
    <state name="joblist='Intege">...</state>
  </process>
  <process id="" name="processor1" implementation="" IP="" portRead="">
    <DCLS>...</DCLS>
    <procedures>...</procedures>
    <start>...</start>
    <state name="job='Integer' p">...</state>
  </process>
</system>
```

Figure 12. XML representation of the SDL model.

In Figure 12, we can see the whole representation of the DEVS-XML model, now transformed to a SDL-XML representation. We can see, as we can expect, that the model contains two processes, the *queue* and the *procesor1*.



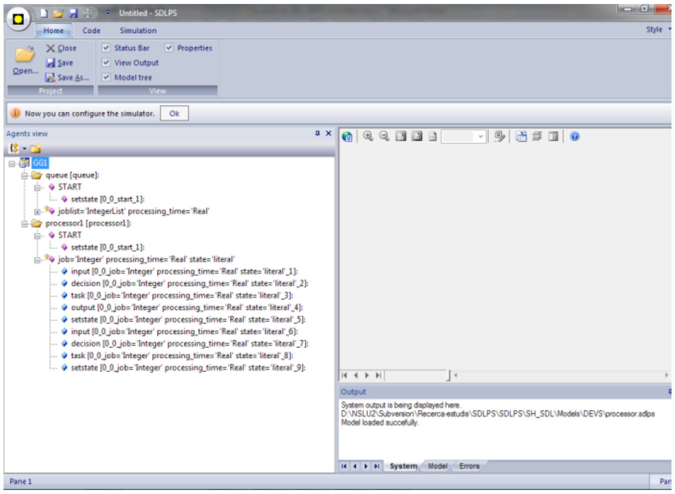
```
<process id="" name="queue" implementation="" IP="" portRead="">
  <DCLS>...</DCLS>
  <procedures>...</procedures>
  <start>...</start>
  <state name = "joblist='IntegerList' processing_time='Real' ">
    <input id="1" name="INT1"></input>
    <decision id="2" name="" iftrue="3" iffalse="5">true</decision>
    <task id="3" name="">
      processing_time="inf";
      joblist=remove_first_list_element(joblist,job_id);
    </task>
    <output id="4" name="INT1" self="yes" to="" via="">
      <param name="delay" value="processing_time"></param>
      <param name="priority" value="0"></param>
    </output>
    <setstate id="5" name="joblist='IntegerList' processing_time='Real' "></setstate>
    <input id="6" name="EXT1"></input>
    <decision id="7" name="" iftrue="3" iffalse="9">is_job(x)</decision>
    <task id="8" name="">
      joblist=add_new_job_to_list(x,joblist);
      processing_time="inf";
    </task>
    <setstate id="9" name="joblist='IntegerList' processing_time='Real' "></setstate>
    <input id="10" name="EXT1"></input>
    <decision id="11" name="" iftrue="12" iffalse="13">true</decision>
    <task id="12" name="">
      processing_time="0";
      joblist=add_new_job_to_list(x,joblist);
      processing_time="inf";
    </task>
    <setstate id="13" name="joblist='IntegerList' processing_time='Real' "></setstate>
  </state>
</process>
```

Figure 13. Process queue definition.

In Figure 13 the XML representation using SDL for the DEVS queue element is shown.

VIII. SIMULATING THE DEVS MODEL ON SDLPS

Regarding the infrastructure used, it is remarkable that SDLPS has been build using C++ and C languages. The code related to the model is represented using a DLL, and the generation of the SDL-XML model is done through a plug-in on Microsoft Visio®.



Now you can configure the simulator.

```
GG1
  queue [queue]:
    START
      setstate [0_0_start_1]:
    joblist='IntegerList' processing_time='Real'
  processor1 [processor1]:
    START
      setstate [0_0_start_1]:
    job='Integer' processing_time='Real' state='literal'
      input [0_0_job='Integer' processing_time='Real' state='literal'_1]:
      decision [0_0_job='Integer' processing_time='Real' state='literal'_2]:
      task [0_0_job='Integer' processing_time='Real' state='literal'_3]:
      output [0_0_job='Integer' processing_time='Real' state='literal'_4]:
      setstate [0_0_job='Integer' processing_time='Real' state='literal'_5]:
      input [0_0_job='Integer' processing_time='Real' state='literal'_6]:
      decision [0_0_job='Integer' processing_time='Real' state='literal'_7]:
      task [0_0_job='Integer' processing_time='Real' state='literal'_8]:
      setstate [0_0_job='Integer' processing_time='Real' state='literal'_9]:
```

```
System output is being displayed here.
D:\NSLU2\Subversion\Recerca-estudis\SDLPS\SDLPS\SH_SDL\Models\DEVS\processor.sdlps
Model loaded succefully.
```

Figure 14. SDLPS system loading the DEVS model.

In Figure 14, we can see the DEVS GG1 model on SDLPS. Note that it is not represented the DEVS model because the Microsoft Visio® plug-in we develop allows the generation of the SDL-XML from a SDL Microsoft Visio® diagram, but the inverse is not yet implemented (we cannot regenerate the diagram form a SDL-XML representation.

On the left side, we can see the tree that contains all the elements that defines the model.

IX. DISCUSSION

Several formal languages exists that can be used to represent a simulation model, like SDL, DEVS, PetriNets [15], or SysML among others. The use of this kind of languages in a simulation project is very desirable, because clearly differentiates the model form the implementation that finally represents the model. Also helps in the understanding of the model and helps in the Validation and Verification process. However, only few simulators allow working with different formal languages in the same environment.

In this paper, we presented a XML representation for atomic and coupled DEVS models with the main goal to serve as a starting point to achieve a complete representation of a DEVS model. This allows the construction of tools that works with DEVS. Also we shown that thanks this representation we can implement a transformation algorithm between DEVS and SDL, allowing that in a single model we can use both formalisms. This simplifies the reuse of simulation models, and the collaboration between different groups that can use the formal language they prefer to define the models. The first issue that is needed to be fixed is that only few of them have been standardized, this often implies that the XML representation of the models needs to assure the inclusion of new features to the language. Also the textual representation of these models, needed in order to be used in a computer simulator, sometimes does not exists.

Also, we presented an infrastructure that allows the simulation of DEVS and SDL models. This combination of both languages can be done thanks the XML representation used for DEVS and SDL models. In this infrastructure we show that the final user can define the models using common simulation tools, like Microsoft Visio®, and thanks a plug-in the XML representation can be obtained.

Now, this infrastructure is currently used in a production environment in real simulation projects for different well known industries. Those projects help us in the Verification of the tool and in the development of some missing plug-ins for some of the more common used computer programs in the industry.

ACKNOWLEDGMENT

Many thanks, for his support in the development of this project, to the Computing Laboratory of the Barcelona Informatics School.

REFERENCES

[1] M. Ho Hwang. (2009, April) DEVS++: C++ Open Source Library of DEVS Formalism. Document. [Online]. http://odevspp.sourceforge.net <retrieved: 10, 2011>

[2] G. Wainer, "CD++: a toolkit to develop DEVS models," *Software, Practice and Experience*, vol. 32, no. 3, November 2002, pp. 1261-1306.

[3] J. Dávila, E. Gómez, K. Laffaille, K. Tucci, and M. Uzcátegui, "MultiAgent Distributed Simulation with GALATEA," in *Procediings of the 9-th IEEE International Symposium on Distributed Simulation and Real Time Applications*, Montreal, 2005, pp. 165-170.

[4] CINDERELLA SOFTWARE. (2007) Cinderella SDL. [Online]. http://www.cinderella.dk <retrieved: 10, 2011>

[5] IBM. (2009) TELELOGIC. [Online]. http://www.telelogic.com/ <retrieved: 10, 2011>

[6] Telecommunication standardization sector of ITU. (1999) Series Z: Languages and general software aspects for telecommunication systems. [Online]. http://www.itu.int/ITU-T/studygroups/com17/languages/Z100.pdf <retrieved: 10, 2011>

[7] L. Doldi, Validation of Communications Systems with SDL: The Art of SDL Simulation and Reachability Analysis.: John Wiley & Sons, Inc., 2003.

[8] R. Reed, "SDL-2000 form New Millenium Systems," *Telektronikk 4.*2000, pp. 20-35.

[9] P. Fonseca i Casas, "SDL distributed simulator," in *Winter Simulation Conference 2008*, Miami, 2008, pp. 2943-2943 http://wintersim.org/abstracts08/POS.htm#fonsecaicasasp84590 . <retrieved: 10, 2011>

[10] P. Fonseca i Casas, "Towards an automatic transformation from a DEVS to a SDL specification," in *Procediings of the 2009 Summer Simulation Multiconference*, Istanbul, Turkey, 2009, pp. 348-353.

[11] P. Fonseca i Casas. (2011) Pau Fonseca i Casas. [Online]. http://www-eio.upc.es/~pau/index.php?q=node/30 <retrieved: 10, 2011>

[12] B.P. Zeigler, H. Praehofer, and D. Kim, *Theory of Modeling and Simulation*.: Academic Press, 2000.

[13] J.L. Risco-Martín, S. Mittal, M.A. López-Peña, and J.M. De la Cruz, "A W3C XML Schema for DEVS Scenarios," in *Spring Simulation Multiconference 2007*, vol. DEVS Symposium, Norfork, Virginia, 2007, pp. 279-286.

[14] P. Fonseca i Casas and Josep Casanovas Garcia, "Using SDL diagrams in a DEVS specification," in *The Fifth IASTED International conference on Modeling Simulation and Optimization*, 2005, pp. 67-72.

[15] L Recalde, E Teruel, and E Silva, "Autonomous continuous P/T systems. Application and Theory of Petri Nets," *Lecture Notes in Computer Science*, 1999, pp. 107-126.