MODELLING AND SIMULATION 2010

THE EUROPEAN SIMULATION

AND

MODELLING CONFERENCE

2010

ESM®'2010

EDITED BY

Gerrit K.Janssens

Katrien Ramaekers

An Caris

OCTOBER 25-27, 2010

HASSELT, BELGIUM

A Publication of EUROSIS-ETI

The European Simulation and Modelling Conference 2010

HASSELT, BELGIUM

OCTOBER 25-27, 2010

Organised by

ETI - The European Technology Institute

Sponsored by

EUROSIS - The European Simulation Society

Universiteit Hasselt

Co-Sponsored by

Ghent University

and

The University of Skovde

Hosted by

Hasselt University

Hasselt, Belgium

EXECUTIVE EDITOR

PHILIPPE GERIL
(BELGIUM)

EDITORS

General Conference Chair

Prof. Gerrit Janssens, Hasselt University, Hasselt, Belgium

Past Conference Chair

Prof. Marwan Al-Akaidi, De Montfort University, Leicester, United Kingdom

Journal Publication Chair
Yan Luo, NIST, Gaithersburg, USA

Local Committee Chairs
Tom Bellemans, Hasselt University, Hasselt, Belgium
An Caris, Hasselt University, Hasselt, Belgium
Davy Janssens, Hasselt University, Hasselt, Belgium
Katrien Ramaekers, Hasselt University, Hasselt, Belgium
Jeanne Schreurs, Hasselt University, Hasselt, Belgium
Koen Vanhoof, Hasselt University, Hasselt, Belgium
Geert Wets, Hasselt University, Hasselt, Belgium

INTERNATIONAL PROGRAMME COMMITTEE

Methodology and Tools
Thomas Hanne, Fraunhofer -ITWM, Kaiserslautern, Germany
Bjorn Johansson, Chalmers Univ. of Techn, Gotheburg, Sweden
Moreno Marzolla, Ist.Naz.di Fisica Nucleare, Padova, Italy
Roberto Revetria, University of Genova, Italy
J. Manuel Feliz Teixeira, University of Porto, Porto, Portugal
Bert van Beek, Eindhoven University of Technology, The Netherlands

Simulation and Artificial Intelligence
Helder Coelho, Fac Ciencias, Lisbon, Portugal
Paulo Cortez, University of Minho, Guimareas, Portugal
Adam Galuszka, Silesian Technical University, Gliwice, Poland
Martin Hruby, Brno University of Technology, Brno, Czech Republic
Esko Juuso, University of Oulu, Finland
Wolfgang Kreutzer, Univ. of Canterbury, Christchurch, New Zealand
Damien Olivier, Universite du Havre, France
Leon Rothkrantz, TU Delft, The Netherlands
Franciszek Seredynski, Polish Acad.of Science, Warsaw, Poland
Jim Torresen, University of Oslo, Norway
Frantisek Zboril, Brno University of Technology, Brno, Czech Republic
Morched Zeghal, Nat. Res.Council Canada, Ottawa, Canada

INTERNATIONAL PROGRAMME COMMITTEE

High Performance Large Scale and Hybrid Computing
Jan Broeckhove, University of Antwerp, Antwerp, Belgium
Giancarlo Fortino, Universita della Calabria, Rende, Italy
Jari Porras, Lappeenranta University of Technology, Finland
Simon See, Oracle, Singapore
Pierre Siron, ONERA, Toulouse, France
Behrouz Zarei, Sharif University of Technology, Tehran, Iran

Simulation in Education and Graphics Visualization
Jan Lemeire, VUB, Brussels, Belgium
Marco Roccetti, University of Bologna, Italy

Simulation in Environment, Ecology, Biology and Medicine
Eduardo Ayesa, CEIT, San Sebastian, Spain
Cyrille Bertelle, LITIS, Le Havre, France
Vasco Cadevez, University of Braganca, Braganca, Portugal
Joel Colloc, LITIS, Le Havre, France
Vahid Nassehi, Loughborough University, Loughborough, United Kingdom
Laurent Perochon, INRA, St Genes Champanelle, France
Cezary Orlowski, Technical University Gdansk, Poland

Analytical and Numerical Modelling Techniques
Ana M. Camacho, UNED, Madrid, Spain
Tom Dhaene, Ghent University, Ghent, Belgium
Clemens Heitzinger, University of Vienna, Vienna, Austria
Panajotis Katsaros, Aristotle University, Thessaloniki, Greece

Web Based Simulation
Dimosthenis Anagnostopoulos, Harokopion University of Athens, Greece
Manuel Alfonseca, Universidad Autonoma de Madrid, Spain
Ammar Al-Khani, Nordphysics, Espoo, Finland
Jose Barata, New University of Lisbon, Portugal
Yan Luo, NIST, Gaithersburg, USA
Jose Machado, University of Minho, Braga, Portugal
Mara Nikolaidou, University of Athens, Greece
Krzysztof Pawlikowski, University of Canterbury, Christchurch, New Zealand

Agent Based Simulation
Frederic Guinand, Universite du Havre, France
Zisheng Huang, Vrije Universiteit Amsterdam, The Netherlands
Jean-Luc Koning, INPG-ESISAR-LEIBNIZ, Grenoble, France
Peter Lawrence, Australian Catholic University, Melbourne, Australia
Ioan Alfred Letia, TU Cluj Napoca, Romania
Paulo Novais, Universidade do Minho, Braga, Portugal
Jan-Torsten Milde, FH Fulda, Germany
Isabel Praca, Ist. Superior do Porto, Portugal
Marco Remondino, University of Turin, Italy
Agosthino Rosa, Technical University Lisbon, Portugal

Cosmological Simulation
Philippe Geril, ETI Bvba, Ostend, Belgium

INTERNATIONAL PROGRAMME COMMITTEE

Decision Models for Intermodal Transport
Track Chair: Cathy Macharis, University of Brussels, Brussels, Belgium
An Caris, Hasselt University, Diepenbeek, Belgium
Jonas Flodén, University of Gothenburg, Gothenburg, Sweden
Bart Jourquin, Fucam, Mons, Belgium
Ekki Kreutzberger, Delft University of Technology, Delft, The Netherlands
Ethem Pekin, University of Brussels, Brussels, Belgium
Ellen Van Hoeck, University of Brussels, Brussels, Belgium
Tom van Lier, University of Brussels, Brussels, Belgium

Activity-based Travel demand Modelling and Simulation
Harry Timmermans, Eindhoven Technical University, Eindhoven, The Netherlands
Mario Cools, Hasselt University, Hasselt, Belgium
Liu Feng, Hasselt University, Hasselt, Belgium
Bruno Kochan, Hasselt University, Hasselt, Belgium
Tom Bellemans, Hasselt University, Hasselt, Belgium
Davy Janssens, Hasselt University, Hasselt, Belgium
Katrien Ramaekers, Hasselt University, Hasselt, Belgium
Geert Wets, Hasselt University, Hasselt, Belgium

Simulation in Logistics, Traffic, Transport, Port, Airport and Hospital Logistics Simulation
Tom Bellemans, Hasselt University, Hasselt, Belgium
An Caris, Hasselt University, Hasselt, Belgium
Davy Janssens, Hasselt University, Hasselt, Belgium
Katrien Ramaekers, Hasselt University, Hasselt, Belgium
Jeanne Schreurs, Hasselt University, Hasselt, Belgium
Koen Vanhoof, Hasselt University, Hasselt, Belgium
Geert Wets, Hasselt University, Hasselt, Belgium

Simulation with Petri Nets
Mauro Iacono, University of Naples II, Italy
Juan de Lara, Univ. Autonoma de Madrid, Spain
Hammid Demmou, LAAS CNRS, Toulouse, France
Olivier Grunder, UTBM, Belfort, France
Guenter Hommel, TU Berlin, Germany
Stefano Marrone, Seconda Universita degli Studi di Napoli, Naples, Italy
Alexandre Nketsa, LAAS-CNRS, Toulouse, France
Jean-Claude Pascal, LAAS-CNRS, Toulouse, France
Ivo Vondrak, Technical University of Ostrava, Czech Republic

Bond Graphs Simulation
Rui Esteves Araujo, DEEC-FEUP, University of Porto, Portugal
Jesus Felez, Univ. Politecnica de Madrid, Spain
Aziz Naamane, DIAM-IUSPIM, Marseille, France
Manuel Rodrigues Quintas, FEUP, University of Porto, Portugal
Andre Tavernier, BioSim, Brussels, Belgium

DEVS
Fabrice Bernardi, University of Corsica, Corte, France
Alexandre Muzy, Universite de Corse, Corti, France
Fernando Tricas, Universidad de Zaragoza, Spain

Fluid Flow Simulation
Diganta Bhusan Das, Loughborough University, United Kingdom
H.A.Nour Eldin, University of Wuppertal, Germany
Markus Fiedler, Blekinge Institute of Technology, Sweden

EUROPEAN SIMULATION AND MODELLING CONFERENCE 2010

EUROSIS is a Division of ETI Bvba, The European Technology Institute, Torhoutsesteenweg 162, Box 4, B-8400 Ostend, Belgium

Printed in Belgium by Reproduct NV, Ghent, Belgium
Cover Design by Grafisch Bedrijf Lammaing, Ostend, Belgium

EUROSIS-ETI Publication

ISBN: 978-90-77381-57-1
EAN: 9789077381571

Preface

Welcome back, my friends, to the show that never ends. We are glad, you could attend the 24th annual European Simulation and Modelling Conference, ESM' 2010, organized by EUROSIS. This year, this international European conference presents the recent advances in the fields of simulation and modeling at Hasselt University, Belgium.

Modelling and simulation are of major interest to the academic community, but intended for understanding, managing and controlling organizations in the real world, be it non-profit organisations, service companies, industry or government. On an open forum like the ESM conference, the interested community can learn about new techniques, tools and methodologies how to change their daily behavior to reach the strengthening objectives in a world where the loudspeakers sing 'improve, improve, improve'.

Welcome, maybe for the first time, to the city of Hasselt, to the city of 'good taste'. From its location in the East of Belgium, Hasselt sends a warm invitation to its close neighbours in the Netherlands, in Germany and in Wallonia for all people of good taste. The inner road ring of this small city may be crossed in ten minutes but only by those who have no appetite in charming streets with beautiful fashion shops and with lots of restaurants and cafés spoiling the inner parts of us all. A reasonable amount of 'jenever', the spirit locally brewed but widely known will print Hasselt for a long time in your memory.

Welcome also to the campus of Hasselt University, located in Diepenbeek, within the open nature. The light of the open space will walk with you into the entrance hall where you are warmly invited for this conference.

Unlike the green on and around the campus, an international conference does not grow by itself. The importance of the work of the International Programme Committee should be highlighted. By attracting papers, by refereeing papers or by supervising the review process, the members contribute to the technical excellence of the conference, which is the aim of the EUROSIS conferences. Thanks also to the university management for the use of its facilities and to the dean of the Faculty of Business Economics for his introductory words. A great thanks to all of you who want to share their findings to the world: the keynote speaker, the presenters and the co-authors. Special thanks to my co-editors of the conference proceedings for everything: three names on a cover. Finally, the management and careful planning rests in the hands, mind and heart of Mr. Philippe Geril. A warm word of thanks to him is written in golden characters.

Preface

Welcome again, next time in Hasselt or on our campus, as I am well convinced you will find our place a fruitful and enjoyable place to stay and to visit more than once.

Gerrit K. Janssens
Conference Chairman

CONTENTS

CONTENTS

MODELLING LANGUAGES AND TOOLS

SIMULATION DATA ANALYSIS TOOLS

NETWORK SIMULATION

ELECTRONICS SIMULATION

COMPLEX SOFTWARE SYSTEMS SIMULATION

CONTENTS

ECONOMICS SIMULATION

MANAGEMENT SIMULATION

HEALTH CARE MANAGEMENT

TRANSPORT INFRASTRUCTURE LAYOUT DESIGN

INTERMODAL TRANSPORT SIMULATION

CONTENTS

TRANSPORT MODELLING AND SIMULATION

LOGISTICS SIMULATION

SUPPLY CHAIN SIMULATION

FLUID FLOW SIMULATION

ENVIRONMENTAL SIMULATION

EDUCATIONAL SIMULATION

SCIENTIFIC PROGRAMME

SIMULATION METHODOLOGY

EFFICIENT STATE SPACE-BASED SIMULATION: AVOIDING REDUNDANCIES IN THE PROXEL METHOD

Robert Buchholz
Claudia Krull
Graham Horton

Otto-von-Guericke University
P.O. Box 4120
39016 Magdeburg, Germany
email: {robert|claudia|graham}@isg.cs.uni-magdeburg.de

KEYWORDS
Model analysis, Model evaluation, Approximation techniques, Markov chain, Discrete stochastic model

Abstract

The simulation of discrete stochastic systems is an important tool to make predictions on possible system behavior in science and industry. The most widely used simulation technique, the discrete event simulation, computes possible simulation results by using random numbers. Consequently, the simulation results are also only random numbers. Alternative state space-based simulation techniques can directly compute the actual system behavior. They are, however, computationally infeasible for most bigger models.
In this paper, we present a modification to one promising state-space simulation technique, the Proxel method. Our modification, called MultiProxel, uses a clustering approach on the set of possible discrete system states (markings) to avoid redundancies inherent to the Proxel method.
Experiments were conducted showing that the MultiProxel approach lets the simulation be performed between two and five times faster for realistic models, without any loss in accuracy. If no redundancies in the model can be exploited, the method has been demonstrated to incur only a small computational overhead.
The MultiProxel approach thus has the potential of making deterministic state-space analysis of existing models more efficient, and to enable the analysis of bigger models that more accurately reflect real systems.

INTRODUCTION

Modeling and simulation of discrete stochastic systems (DSMs) is an important tool in Engineering, Logistics and other areas. It is used to analyze possible future behavior of existing systems, evaluate consequences of possible changes to real systems, or even analyze systems that do not yet exist.

The simulation technique of choice is usually the discrete event simulation (DES) which uses random numbers to generate a possible system development, and then performs numerous simulation runs with different random numbers in order to compute an accurate mean system behavior. The reliance on and influence of random numbers make DES often undesirable to use. For models containing rare events, DES is often unfeasibly slow.
An alternative to DES are state space-based simulation methods. Those can directly compute the mean system behavior without having to resort to replications or random numbers. These techniques, however, have their own disadvantages: they are either limited in their expressiveness (e.g. can only simulate Markovian systems) and thus are not applicable to the majority of discrete systems, or are computationally too expensive to be useful in practice.
One promising state space simulation method is the Proxel method in (Horton 2002, Lazarova-Molnar 2005). Its expressiveness is not limited to specific probability distributions and its superior efficiency on some models has been demonstrated in (Lazarova-Molnar and Horton 2004). However, the method is still unfeasibly slow for most models.
The goal of this paper is therefore to increase the efficiency of said method. This is done by developing an extension to that method which avoids a certain type of computational redundancy. This will make state space-based simulation feasible for more models. It will also allow for the analysis of bigger, more realistic Hidden non-Markovian Models (c.f. (Krull and Horton 2009a)), which use the Proxel method as their primary solution method.

RELATED WORK

Since the influence of randomness on simulation results is sometimes undesirable, state space-based simulation techniques were developed that can compute the desired simulation result directly, without any influence of randomness.

Continuous-Time Markov Chains (CTMCs) are one major state-space modeling and solution approach (Bolch et al. 1998). Their results are not influenced by randomness and result computation is efficient. However, CTMCs limit all state transitions to be memoryless, i.e. to have an underlying exponential probability distribution. This limitation prevents CTMCs from accurately modeling most real systems and thus causes CTMCs to not be applicable to the majority of discrete stochastic systems.

One way to extend the applicability of CTMCs is to convert all general probability distributions of a model to a well-adjusted chain of exponential distributions, a so-called phase-type distribution. This way, many discrete models can be represented as CTMCs and can be solved using CTMC solution algorithms. The downsides of phase-type distributions are that they increase the model's state space, and that the computation of the chain parameters is time-consuming itself (Bobbio et al. 2002, Isensee and Horton 2005).

The Proxel method was developed in order to overcome the limitations CTMCs have. Simply extending CTMCs using the method of supplementary variables from (German 1995) in order to model age-dependent state change probabilities poses algorithmic difficulties and would generate stochastic matrices that are too big to be handled. Therefore, the Proxel method creates only the necessary parts of these matrices on the fly. This allows for the efficient simulation of some discrete stochastic systems, but is often unfeasibly slow.

One big disadvantage of the Proxel method compared to CTMCs is that due to the dynamic generation of relevant parts of the matrix, certain intermediate results need to be computed over and over again. In this paper, we are consequently developing a modification to the Proxel method that removes some of these redundancies. As a first step to this extension, we will briefly review the Proxel method. Since discrete stochastic models can often intuitively be represented as Generalized Stochastic Petri Nets (GSPNs) with arbitrary firing times, the Proxel definition borrows some terms from the definition of GSPNs. So, in the description of the Proxel method, the different discrete states are called "markings" and the state transitions between them are simply called "transitions"

REVIEW OF THE PROXEL METHOD

The Proxel method facilitates state space-based simulation of DSMs with arbitrary probability distributions (Lazarova-Molnar 2005). The basic approach of the Proxel method is to divide simulation time into time steps of equal size and then to trace possible system developments for each time step in parallel.

The possible system developments are stored in tuples (m, τ, Π) called "Proxels". Here, m represents the marking, i.e. the discrete system state. The entry τ is the age vector containing the most recent duration of inactivity for all active or otherwise relevant state transitions, i.e. the time during which a given state change could have happened, but did not. Finally, the entry Π is the probability of the combination of marking and age vector.

Simulation starts by determining a suitable simulation time step and providing a Proxel for the start state, usually with an age vector τ containing only zeros, and with a probability of one.

The simulation is then iteratively performed for each time step: For each Proxel of a time step, the probabilities of all possible single[1] state changes during a time step (and the probability of no state change occurring at all) are computed using numerical integration of ordinary differential equations (Buchholz 2008). The outcomes of these state changes are then stored as Proxels for the next time step.

The pseudo-code for a single Proxel simulation step is shown in Algorithm 1. For a complete simulation, this algorithm simply needs to be executed iteratively for each time step till the desired end of the simulation, with the output *currentStep* of one time step being the input *inputProxels* of the next one.

Algorithm 1: Simulating a single time step using the standard Proxel algorithm

```
Data: inputProxels, Δt
foreach Proxel p ∈ inputProxels do
    Π_trans = getTransitionProbabilities( p.transitions,
    p.ageVector);
    stayProb = 1 - sum(Π_trans);
    if (stayProb > 0) then
        currentStep.addOrMerge
        (createInactivityProxel( p, Δt));
    foreach Transition trans ∈ p.transitions do
        currentStep.addOrMerge(
        createTransitionProxel(p, trans, Π_trans[trans]));
return currentStep;
```

One issue with a naive implementation of the Proxel algorithm is that each Proxel of a given time step causes multiple Proxels for the next time step to be created (one for each possible state change and one for inactivity), resulting in an exponential growth of the number of Proxels per time step. To curb this exponential growth, Proxel merging is used: all Proxels of a given time step that represent the same age vector and marking are merged into a single Proxel by accumulating their probability. Algorithmically, efficient merging requires an efficient way of finding these duplicate Proxels. In the current implementation, this is done by storing all Proxels of a give time step in a binary search tree instead of a generic container. This slightly increases the

[1] The Proxel method does not compute the probability of multiple state changes occurring during a single time step. This is a known source of inaccuracy, especially for big time steps.

time complexity of adding a single Proxel from $O(1)$ to $O(log(n))$, but reduces the time complexity for finding duplicates from $O(n)$ to $O(log(n))$.

Reflections on the Performance of the Proxel Method

The Proxel method can efficiently simulate some models, but for most bigger models, the approach is too slow to be feasible.
To determine the major performance bottlenecks, our Proxel simulation software was profiled using the tool Valgrind (Nethercote and Seward 2007). Profiling showed that about 80% of the computation time of a Proxel simulation run is spent on computing the state transition probabilities and on inserting Proxels into the search tree. Thus, a reduction of the number of Proxels and the number of state transitions probabilities to be computed could significantly speed up Proxel computations.

THE MULTIPROXEL APPROACH

Observations Leading to this Project

The extensions to the Proxel method developed here to speed up Proxel computations are based on two observations on the behavior of a Proxel simulation:
First, it was found that while the state transition probabilities to be computed depend on all active state transitions (which are determined by the marking) and potentially all elements of the age vector, they do not actually depend on the marking itself. In the example of customers waiting in line to be served at a bank counter in Figure 1, different numbers of waiting customers need to be represented by different markings, but these numbers do not influence the probability of the current service ending or a new customer arriving in the next time step.
Thus, one may cluster these markings of a simulation model that agree in the types and parameters of all of their active state transitions. This allows for the transition probabilities to be computed only once for each cluster of markings (instead of for each marking individually, as is done so far). These transition probabilities are then applied to all markings of a cluster, preventing redundant probability computations.
Second, when dealing with these clusters, the target markings for a given state transition for markings of one cluster tend to fall into a single other cluster. Put another way, if the target marking of a marking m_1 through state transition t is m_1', then the target markings of other markings in the same cluster as marking m_1 through state transition t are likely to fall into the same cluster as m_1'. In the bank example in Figure 1, all markings but the one for "0 Customers Waiting" form a cluster, and the state transitions for almost all of them

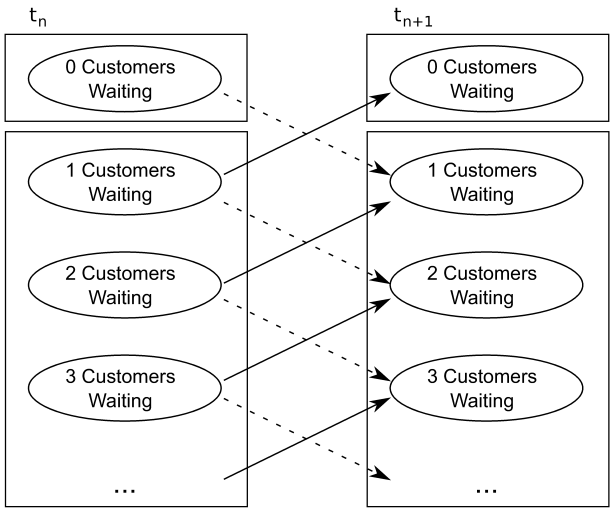


Figure 1: Schematic representation of the state space for the "Bank" example. Ovals represent markings, rectangles group them into clusters. Dashed lines represent the state transitions for "Customer Arriving", solid lines the ones for "Customer Served".

lead to markings inside a single cluster. This behavior can be exploited by modifying the Proxel data structure to contain information on all markings of a cluster instead of just a single marking. Then, many computations can be performed efficiently on clusters of Proxels. Both ideas together, the clustering of markings to prevent redundant computations, and the joint treatment of markings belonging to the same cluster, are the basis of our new MultiProxel approach.

Defining the MultiProxel Approach

Identifying Marking Clusters One necessary preprocessing step for MultiProxel simulations is to find sets of markings with identical state transition behavior, i.e. markings that agree in the types and parameters of the probability distributions of all of their transitions. Our approach is to serially number all state transitions and then to assign a cluster ID number to each marking based on the state transitions active for that marking. Specifically, the cluster ID for a marking is the binary string $B = b_0b_1b_2...b_n$ where each b_i is 1 if state transition i is active for that marking, and is 0 otherwise. All markings with identical ID then form a cluster.

MultiProxel Definition Next, the MultiProxel data structure needs to be defined. While a Proxel was defined as the tuple (m, τ, Π) containing the marking, age vector and the probability of the combination, a MultiProxel should hold data on all markings of a cluster. Therefore, we define a MultiProxel as the tuple $(cluster, \tau, S)$ were $cluster$ identifies the cluster that all markings of a MultiProxel belong to, and S is a set of

(m, Π) pairs, storing the probability Π for each marking m of *cluster*.
At any given point in simulation time, the set of all MultiProxels contains exactly the same information as would the set of all Proxels of a normal Proxel simulation. Each representation can easily be converted to the other one. Consequently, the simulation results will also be identical.

The Algorithm

The usual Proxel simulation algorithm from (Lazarova-Molnar 2005) as shown in Algorithm 1 needs to be modified only slightly to make use of MultiProxels: transition probabilities are computed only once for each MultiProxel, and then are used for each marking that is part of the MultiProxel. And since the target states that can be reached through a single transition from all markings of a MultiProxel are likely to be concentrated in very few MultiProxels, only these few MultiProxels need to be located in the search tree.
The pseudocode for a single MultiProxel simulation step is given in Algorithm 2. As for the Proxel pseudocode in Algorithm 1, for a complete simulation this algorithm simply needs to be executed iteratively for each time step, with the output *currentStep* of one time step being the input *inputMultiProxels* of the next one.

Algorithm 2: Simulating a single time step using the MultiProxel algorithm

```
Data: inputMultiProxels, Δt
foreach MultiProxel mp ∈ inputMultiProxels do
    Π_trans = getTransitionProbabilities(
    mp.cluster.transitions, mp.ageVector);
    stayProb = 1 - sum(Π_trans);
    if (stayProb > 0) then
        currentStep.addOrMerge
        (createInactivityProxel( mp, Δt));
    foreach Transition trans ∈ mp.cluster.transitions
    do
        mp_prev = NULL;
        foreach Marking m ∈ mp.cluster.markings do
            m_target = m.getTarget(trans);
            if (mp_prev.cluster = m_target.cluster) then
                mp_prev.addProbability(m_target,
                mp.probability(m) * Π_trans[trans]);
            else
                mp_prev = currentStep.addOrMerge(
                createTransitionProxel(mp, m_target,
                trans, Π_trans[trans]))
return currentStep;
```

Here, mp_{prev} is used to cache the previous target MultiProxel to prevent multiple successive lookups of the same target MultiProxel and thus allowing to often find target MultiProxels in constant time.
In an efficient implementation, all markings of a cluster would be numbered serially so that the set of all (m, Π) pairs of a MultiProxel can be implemented efficiently as a simple array that stores the probability of marking i at index i, and can be accessed in $O(1)$.

Impact Conducting state space-based simulation using MultiProxels instead of Proxels allows for much more efficient simulations: with MultiProxels, each state transition probability needs only be computed once for each MultiProxel, no matter how many markings are part of the associated cluster. Thus, the overall number of transition probabilities to be computed is reduced substantially.
Also, as was stated before, with respect to a single state transition, the target markings of most markings of a cluster fall into a single other cluster. Consequently, for a given state transition, the target for most markings of a MultiProxel are also only a few or even a single MultiProxel. Thus, only those different MultiProxels needs to be found explicitly in the search tree with a time complexity of $O(log(m))$ in the number of MultiProxels. The probability for each marking can then be added to the MultiProxels in $O(1)$.
Finally, the MultiProxel approach will cause two different types of small computational overhead compared to the normal Proxel approach: First, a constant additional amount of time needs to be spend to create the clusters. Second, overhead by a constant *factor* is caused by MultiProxels, because they are more costly to compare to each other (since they contain more elements) and have to maintain (m, Π) pairs with zero probability. The Proxel approach would simply not have created Proxels with a probability of zero, but MultiProxels need to be created as soon as there is some probability left in at least one of the markings it represents. This overhead, however, is not substantial compared to the potential benefits, as shown in the next section.

EXPERIMENTS

Experiments were conducted to assess benefits and applicability of the MultiProxel algorithm. For that purpose, the following questions were investigated:

- How high is the overhead of the MultiProxel method?
- How high is the performance gain for typical models?
- How high is the performance gain for models particularly suited for the approach?

For each question, a suitable simulation model was chosen, and was simulated with both the Proxel and MultiProxel algorithms. For all experiments conducted,

the actual computed simulation results for the Proxel and MultiProxel simulations were identical, giving additional confidence that the MultiProxel approach indeed solves the problem at hand correctly.

Experiment Setup

Multiple models were simulated to asses the efficiency of the MultiProxel approach. Each model was simulated multiple times, varying the simulation step size and thereby the result accuracy. The particular step sizes for the experiments were chosen in order to yield computation times that are high enough to eliminate the random influence of concurrently running programs on the computation time.
For the same reason, all computation times measures are not the overall simulation duration (wall clock time), but the CPU usage times of the respective simulation processes. All experiments were conducted on a Core2Duo 3 GHz CPU. However, both simulation programs are single-threaded, using only one of the CPU cores. The CPU was manually set to always run at full speed in order to eliminate the influence of dynamic CPU throttling on the computation times.
The implementations of the Proxel and MultiProxel algorithms share most of their source code. They use identical code for Proxel storage, numerical integration, and probability distributions. Different code is used only when the differences in the algorithms make it necessary. This way, the differences in computation time can indeed be attributed to the different theoretical approaches with great confidence.

Computational Overhead

The question of computational overhead is important in order to determine which fraction of markings needs to be clusterable for the model to be simulated more efficiently using the MultiProxel approach. The overhead itself is best determined using simulation models where no markings can be clustered and thus the MultiProxel approach does not provide any benefits, but still imposes the overhead. This way, the difference in computation time between the Proxel and MultiProxel approaches for such a model yields the computational overhead.
One example of such a model is the three marking fully-connected model (cf. Figure 2). As the state transition probability distribution are all different, the model cannot be clustered. Therefore, of the three markings that are reachable in this model, each is the only marking in its respective cluster in a MultiProxel simulation.
The model was simulated with each algorithm and two different simulation step sizes. Each combination was simulated three times and the computation times of the three runs were then averaged to yield the results shown in Table 1.
For both simulation step sizes, the difference in computation time amounts to about 13%. Since clustering is not applicable to this model, these 13% of loss in efficiency represent the computational overhead of the MultiProxel method. And since the relative overhead of about 13% is the same factor for both step sizes, it can be assumed that no overhead by a constant *value* (as opposed to overhead by a constant *factor*) of the MultiProxel approach was observed. Thus, the constant overhead of clustering the markings on the computation time is negligible compared to the overall simulation time.

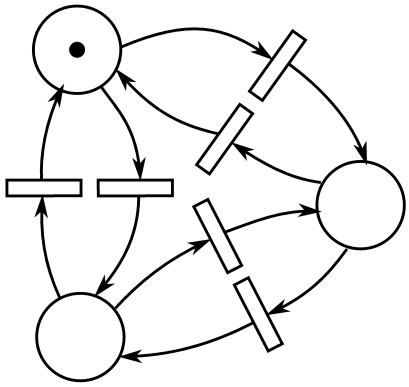


Figure 2: Stochastic Petri Net of the "Fully Connected" model.

Step Size	Computation Time		Overhead(+)
	Proxel	MultiProxel	Gain(-)
2	3.7s	4.2s	+13.9%
1	29.4s	33.1s	+12.6%

Table 1: Computation Times for the "Fully Connected" Model.

Memory Consumption Overhead

The differences in memory consumption between the two algorithms are highly implementation-dependent and therefore are not analyzed here in detail. In a naive implementation, a simulation program would only allocate as much memory as is required to store all (Multi)Proxels, and a MultiProxel would only be as big as necessary to store its probability vector. Under these circumstances, a MultiProxel simulation with its more compact representation will always require equal or less memory than a Proxel simulation.
Our implementation, however, is optimized to reduce memory allocation time overhead and not size. To that end, it always allocates 2^n (Multi-)Proxels at a time, and all MultiProxels of a simulation have identical size, with a probability vector size corresponding to the maximum number of markings in a cluster. Thus, memory consumption overhead or reduction of the MultiProxel approach depends strongly on the number of clusters and the distribution of markings between them. Generally, in our implementation, memory consumption of the

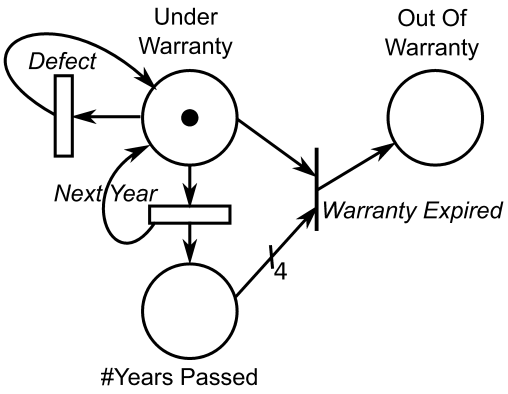


Figure 3: Stochastic Petri Net of the "Warranty" model.

Step Size	Computation Time		Overhead(+)
	Proxel	MultiProxel	Gain(-)
5	12.5s	5.5s	−55.9%
2	79.7s	37.3s	−53.1%

Table 2: Computation Times for the "Warranty" Model.

MultiProxel method tends to be of the same magnitude as for the Proxel method.

Performance on Typical Models

To analyze performance of the approach on typical models, we chose the "Warranty" model previously analyzed using the Proxel method in (Lazarova-Molnar and Horton 2004). It is a real-world simulation model measuring the costs incurred due to defects of a machine part under warranty. The model is actually used in industry and was not developed with MultiProxels in mind, making it a good candidate to assess MultiProxel performance under general conditions. Figure 3 shows a Petri Net of the model.

The model's state space consists of five markings (first, second, third, fourth year of warranty, and "out of warranty"), which can be folded into two clusters. Thus, a theoretical computation time reduction of up to $^3/_5$ can be assumed.

The actual computation times for different simulation step sizes are shown in Table 2. The simulation using the MultiProxel took about 55% less time in both cases, almost reaching the theoretical maximum of 60% for this model. Again, computation times between Proxel and MultiProxel differ by a constant factor, indicating that this is due to the constant-factor overhead of comparing and managing MultiProxel. Therefore, the additive step size-independent clustering overhead appears to be negligible compared to the overall simulation time even for models where clustering does succeed.

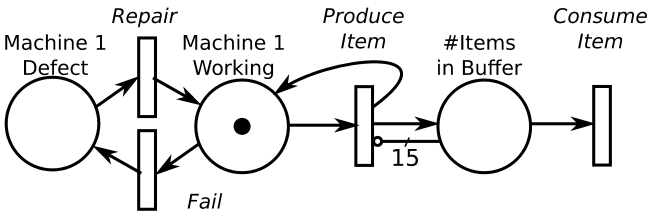


Figure 4: Stochastic Petri Net of the "Buffer" model.

Performance on Suitable Models

Finally, the approach's performance on particularly suited models was assessed. Here, we chose a typical buffer dimensioning problem (cf. Figure 4): one machine in a factory produces items that another machine consumes, with both processes being influenced by randomness. The first machine may fail occasionally, or may sometimes produce items too slowly, so that the second machine would have to wait for input at certain times. In order to compensate this variability, a buffer is installed between both machines, allowing temporary storage of a certain number of items produced by the first machine. This model can be used to analyze the influence of different buffer sizes on the throughput of the whole system.

The model is particularly suited for MultiProxel simulations, because increasing the buffer size will only increase the number of markings, but not the number of clusters. Specifically, for buffers with a capacity of $n \geq 2$ items, the model consists of $2n + 2$ markings that can always be folded into only five clusters. Thus, the performance gain should increase when increasing the buffer size.

In our experiments, the "Buffer" model was simulated using only a single simulation step size, since previous experiments established that the step size does not impact the relative gain in efficiency. The size of the buffer, however, was varied between experiments to determine the influence of the state space size on the efficiency gain.

The results are shown in Figure 5. For very small buffer sizes, computation times of both approaches are almost identical. With increasing buffer size, however, the computation time for the Proxel algorithm increased much faster than that for the MultiProxel one. For a buffer size of 19 items, the computation time of the MultiProxel approach is about 80% less than that of the normal Proxel approach. For higher buffer sizes, the efficiency actually decreases slightly.

For this model, an 80% reduction in computation time seems to be the upper limit. The most likely reason for this limit is that when the buffer size is increased beyond a certain point, most markings of many MultiProxels become impossible to reach. Thus, many MultiProxels will be sparsely populated and represent the same information as would only a few Proxels, reduc-

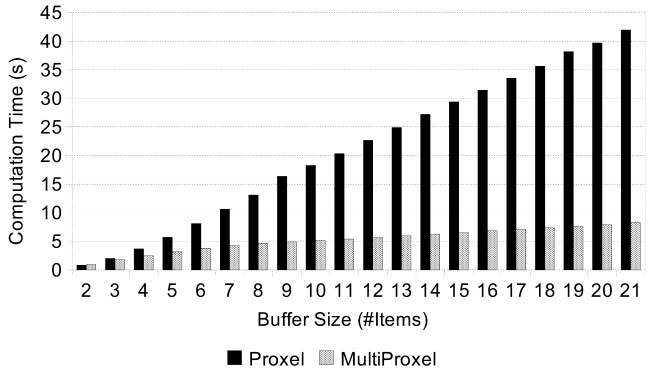


Figure 5: Computation Times for Different Buffer Sizes in the "Buffer" Model. Simulation Step Size is the Same in all Cases.

ing the MultiProxel performance gain over the Proxel approach.

CONCLUSION AND OUTLOOK

In this work, we developed and implemented a variation of the Proxel method for the state space-based simulation of discrete stochastic models. The new MultiProxel approach promises to be more efficient than the standard Proxel approach, since it avoids certain redundant computations of the Proxel method.

The experiments have shown a relatively small computational overhead of the MultiProxel method compared to the Proxel method, if the model contains no redundancies that the MultiProxel approach may exploit. Further experiments have shown a reduction in computation time of about 50% for typical models, and of up to 80% for - still realistic - models with an advantageous state space structure. Memory consumption was explained to be highly implementation dependent, but to be similar for both approaches in comparable implementations.

The new approach is based on a fully automated clustering algorithm, requiring no tuning or additional user input beyond that of Proxel simulations. Consequently, simulation software can transparently replace their core Proxel algorithm by a MultiProxel one. Since the new approach imposes only a slight overhead and only in some cases, and demonstrated a remarkable performance increase otherwise, we recommend the new MultiProxel algorithm to be used as the default algorithm for a general-purpose Proxel simulator.

Model Classes Benefiting from MultiProxel Approach

The MultiProxel approach is applicable to all discrete stochastic models, and for most models, the gains in efficiency will outweigh the slight computational overhead. Furthermore, there are some classes of systems that will particularly benefit from MultiProxel due to the structure of their state space:

Generalized Stochastic Petri Nets Most DSMs that are represented as a generalized stochastic Petri Net with arbitrary firing distributions have a particular model structure. Most transitions in GSPNs exhibit only three different types of behavior:

- They are active
- They are inactive, because there are not enough tokens in a place to satisfy firing conditions
- They are inactive since an inhibitor arc blocked them due to too many tokens being in a place

Thus, independent of how many tokens may populate a place, such a part of a model can be represented by three state clusters, while a standard Proxel simulation requires as many markings as there can by tokens in a place.

Queuing Systems Queuing systems (Law and Kelton 2000) are discrete stochastic systems that model queues (waiting lines) or networks thereof. Here, a queue consists of a buffer to store items, and one or multiple servers that process these items individually and afterwards either remove them from the system or forward them to another queue. Queuing systems have manifold applications, among others in the simulation of computer networks and logistics.

Similar to stochastic Petri Nets, the types of possible state transitions for queuing systems follow a simple structure. A queue may be:

- empty and thus cannot produce output.
- partially used and thus may receive input and produce output.
- full and thus cannot receive input.

A standard Proxel simulation would need to represent each possible number of items in a queue as an individual marking. The MultiProxel approach on the other hand requires only three clusters for the three cases mentioned above, independent of the queue's capacity.

Hidden non-Markovian Models Hidden non-Markovian Models (HnMM) are a new paradigm to model and analyze systems whose runtime behavior cannot be observed directly, but for which some system behavior produces observable output (Krull and Horton 2009a). Unlike older approaches, HnMMs allow for the hidden model to be any discrete stochastic system with arbitrary firing times. HnMMs enable the analysis of certain characteristics of the system behavior that produced a given sequence of observed output.

Currently, the majority of HnMMs is analyzed using a slightly modified Proxel method (Krull and Horton 2009b). Here, in many cases, the result of interest is some aggregate of the possible system behavior over a given simulation time, subject to the observed output. For example, in (Buchholz et al. 2010), the simulation tried to find the probable number of defective items produced by a certain machine. The current number of defects at a given situation (Proxel) and simulation time was encoded into the discrete state of each Proxel. This, however, means that the number of discrete systems states is increased by a factor corresponding to the quantity measured (in the example the maximum number of defective items produced by one machine). This quantity, however is only stored for statistical analysis of the simulation results. It has no influence on further system behavior. Consequently, using the MultiProxel approach to solve HnMMs, all system states that encode the same marking with different histories could be represented by a single state cluster. Thus, with the MultiProxel method, the computational overhead of HnMMs on the Proxel method can almost be eliminated.

Benefits

The increased efficiency of computations due to the MultiProxel approach result in shorter computation times to obtain the same accuracy as before, so that results are now available and may be used sooner. By reducing the simulation step size, results may also be obtained at higher accuracy in same time and thus allow for more accurate predictions within the same time frame. Finally, using the gain in efficiency, bigger, more realistic models may be simulated using state space-based techniques for the first time.

Future Work

The MultiProxel approach captures and prevents redundancies in situations where the transition probabilities for different markings for a given age vector are identical. It does not yet prevent redundancies for cases where age vectors between Proxels may differ, but transition probabilities are nevertheless identical, either by coincidence, or because the age vector contains ages of state transitions that are not currently active. Further research may also show ways to reduce redundancies in computations performed for different time steps, or even between slightly different models. It might also be advantageous to identify simulation intermediate results that are not necessarily identical, but very similar, and to exploit these similarities.

References

Bobbio A.; Horváth A.; and Telek M., 2002. *Phfit: A general phase-type fitting tool.* In *Proceeding of 12th Performance TOOLS*. 82–91.

Bolch G.; Greiner S.; de Meer H.; and Trivedi K.S., 1998. *Queuing Networks and Markov Chains.* John Wiley & Sons, New York.

Buchholz R., 2008. *Improving the Efficiency of the Proxel Method by using Variable Time Steps.* Master's thesis, Otto-von-Guericke Universität Magdeburg.

Buchholz R.; Krull C.; Horton G.; and Strigl T., 2010. *Using Hidden non-Markovian Models to Reconstruct System Behavior in Partially-Observable Systems.* In *3rd International ICST Conference on Simulation Tools and Techniques.*

German R., 1995. *Transient Analysis of Deterministic and Stochastic Petri Nets by the Method of Supplementary Variables.* In *Proceedings of the 3rd International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 95).* IEEE Computer Society, 394–398.

Horton G., 2002. *A New Paradigm for the Numerical Simulation of Stochastic Petri Nets with General Firing Times.* In *European Simulation Symposium.* SCS European Publishing House, Dresden, Germany.

Isensee C. and Horton G., 2005. *Approximation of Discrete Phase-Type Distributions.* In *Annual Simulation Symposium 2005.* San Diego.

Krull C. and Horton G., 2009a. *Hidden non-Markovian Models: Formalization and Solution Approaches.* In *Proceedings of 6th Vienna International Conference on Mathematical Modelling.* Vienna, Austria.

Krull C. and Horton G., 2009b. *Solving Hidden non-Markovian Models: How to Compute Conditional State Change Probabilities.* In *21st European Modeling and Simulation Symposium.* Santa Cruz de Teneriffe, Spain.

Law A.M. and Kelton W.D., 2000. *Simulation, Modelling and Analysis*, McGraw-Hill, chap. 1. 3rd ed., pp. 94–98.

Lazarova-Molnar S., 2005. *The Proxel-Based Method: Formalisation, Analysis and Applications.* Ph.D. thesis, Otto-von-Guericke Universität Magdeburg.

Lazarova-Molnar S. and Horton G., 2004. *Proxel-Based Simulation of a Warranty Model.* In *European Simulation multiconference.* SCS European Publishing House 2004.

Nethercote N. and Seward J., 2007. *Valgrind: A Framework for Heavyweight Dynamic Binary Instrumentation.* In *Proceedings of ACM SIGPLAN 2007 Conference on Programming Language Design and Implementation (PLDI 2007).* San Diego, California, USA.

About Possibilities for Real and Alleged Uncertainties within Numerical Optimization Processes

Olaf Frommann
University of Applied Sciences Bremen
Flughafenallee 10
D-28199 Bremen, Germany
E-mail: olaf.frommann@hs-bremen.de

KEYWORDS

Optimization, Uncertainty, Fuzzy Logic, Objective function, CFD, Multi-Objective, Multipoint Design, Simulation

ABSTRACT

Within the usual product design process uncertainties are quite common and can be handled by well standardized procedures like Six Sigma or Robust Design methods. During the last years, numerical optimization is increasingly applied for the improvement of product qualities. However, there exist some kinds of real and also alleged uncertainties in this context that stem from different sources and may harm the successful utilization of the latter. It is necessary to understand theses coherencies in order to work properly within the optimization design process. This survey points out possible causes of uncertainties within the optimization process, their sources and impacts with respect to simulation, as well as alleged uncertainties, which are often dealt with in an inappropriate way. The focus lies on the design process itself, represented by the simulation and optimization workflow, as well as human factors, represented by uncertain goals respectively knowledge deficiency. The impact of these factors will be shown and possible solutions presented.

INTRODUCTION

In the past, many studies have been conducted concerning the design under consideration of uncertainties. They were motivated in early times by manufacturing problems, where variability within the manufacturing process itself and material properties mainly affect the final product quality. The tools used are based upon statistical methods in order to cope with stochastic fluctuations. This approach has subsequently been adopted for the treatment of the design process, above all for the design optimization, in order to cope with more or less alleged uncertainties. However, while in real life variability and therefore uncertainties are identified rather easily, they stem from different and sometimes hard to identify sources within the design process. The main sources of uncertainty in this context are the following.

1. Inaccurate or only partially modelled impacts of manufacturing aspects
2. Unknown inaccuracies of the simulation, e.g. through grid deterioration and subsequently convergence degradation during the optimization process, caused by unpredictable, although sometimes reproducible changes of the design variables
3. Inappropriate design procedures caused by knowledge deficiency of the designer, e.g. disregard of physical and/or operational aspects
4. Imponderability of design goals

Whereas the first point is the most obvious case of uncertainties and procedures to cope with them are well established, much effort is currently put into the remaining three topics. In the following, these are discussed in detail in relation to current scientific work. Section 2 deals with uncertainties of simulations within the optimization process, section 3 shows some misunderstandings of the design process, and section 4 clarifies the role of design objectives, which are due to human decisions.

EFFECT OF INACCURACIES FROM SIMULATION CODES

As an example the design of an airfoil for an aircraft may serve to describe the possibilities and impacts of uncertainties based upon a priori unknown simulation inaccuracies. The simulation of the flow around an airfoil is based upon the CFD (Computational Fluid Dynamics) simulation, where the physical conservation equations for mass, momentum and energy are solved within a domain, which is discretized by a computational grid, Fig. 1. The equations resp. the derivatives are approximated by finite differences and the accuracy strongly depends on the grid quality. The question, when a simulation is finished, is generally answered by the so called residual, which is an indicator for the difference of the actual solution compared to the exact one, called the convergence criterion. If the qualities of an airfoil, i.e. the forces described by the lift and drag coefficient cl and cd, are calculated, one notices that these values in general are varying during the simulation depending on the current convergence. The development of the drag and lift coefficient in dependence on the residual for an airfoil is shown in Fig. 2 and 3. It can be seen that a more or less converged result with respect to the forces approximately occurs only after the residual has reached a value of at least 10^{-5}. This makes clear that the mere concentration on the residual may be no guaranty for a reliable solution. It is more convenient to concentrate on ongoing changes in the force coefficients. If they don't change significantly any more, the solution has reached equilibrium. The question left is, whether less converged solutions may at least serve as a trend indicator, saving time for the computations and leading to comparable results when calculating the optimized design to full convergence.

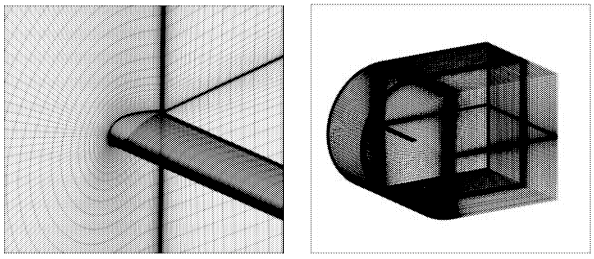


Figure 1: Example of a Surface and Volume Grid for the Rotor of a Wind Energy Turbine

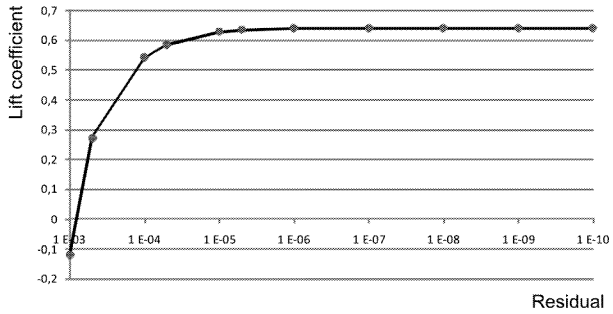


Figure 2: Lift Coefficient in Dependence on the Residual

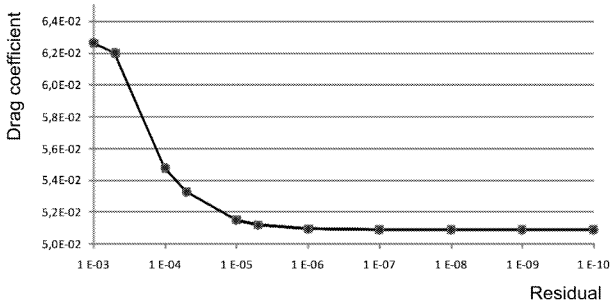


Figure 3: Drag Coefficient in Dependence on the Residual

In the context of numerical optimization computational grids are created automatically for various geometries over and over again, Fig. 4, leading to the fact that the convergence properties, which strongly depend upon the grid quality which in turn depends on the design variables, are unknown. At least in the case where stochastic optimization algorithms like evolution strategies or simulated annealing are applied, this produces some kind of uncertainty, because the created geometries may be out of the expected area and therefore the grids may be inadequate, leading to convergence problems or inexact results. The question is how this affects the design resp. optimization process. In order to answer this question, exactly the same optimization loop has been used for various convergence criteria, applying the deterministic Downhill-Simplex optimization strategy after Nelder-Mead, ensuring reproducible and comparable results.

The goal was to minimize the aerodynamic drag of an airfoil for a given lift coefficient of $cl = 0.5$. In order to attain the desired lift coefficient, which depends on the angle of incidence α, Fig. 5, the derivative $\partial cl/\partial\alpha$ has been computed from two angles of incidence and then the required angle for the desired lift was calculated, resulting in three CFD simulations for each geometry. This additional effort yields an achieved lift coefficient within ±1.3% and is necessary in order to be able to compare the drag values in contrast to other studies (Shimoyama 2006, Duvigneau 2007) where only the angle of incidence was prescribed. This is an inappropriate condition, which will be described in the next section.

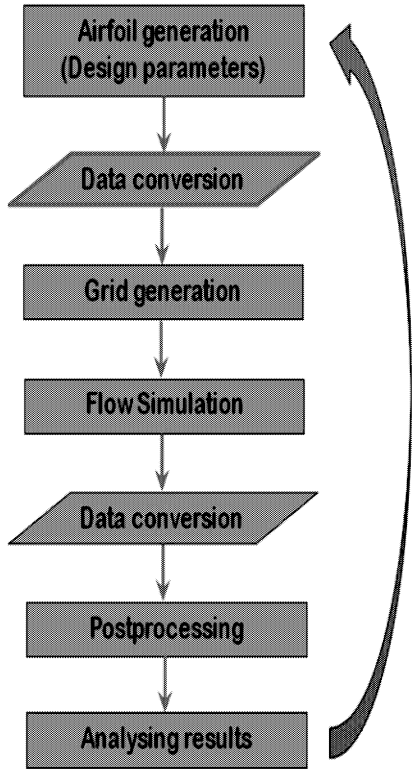


Figure 4: Optimization Work Flow for an airfoil Design

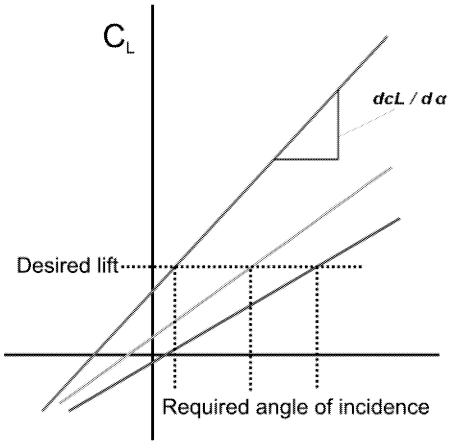


Figure 5: Lift Coefficient in Dependence on the Angle of Attack

The residuals have been limited for each run from 10^{-3} down to 10^{-9} respectively. Fig. 6 shows the development of the objective function (drag) and Fig. 7 the best airfoil geometries with the corresponding pressure distributions, while Tab.1 provides the respective values for lift and drag.

It is obvious that for residuals of 10^{-3} and 10^{-4} the convergence in optimization is not reached. Even very small changes in geometry cause severe differences in drag and lift due to not converged solutions, leading to the erratic trend shown in Fig. 6. Above all, the desired lift has not been reached in these cases, even in the case of partial convergence, showing the uselessness of less converged solutions.

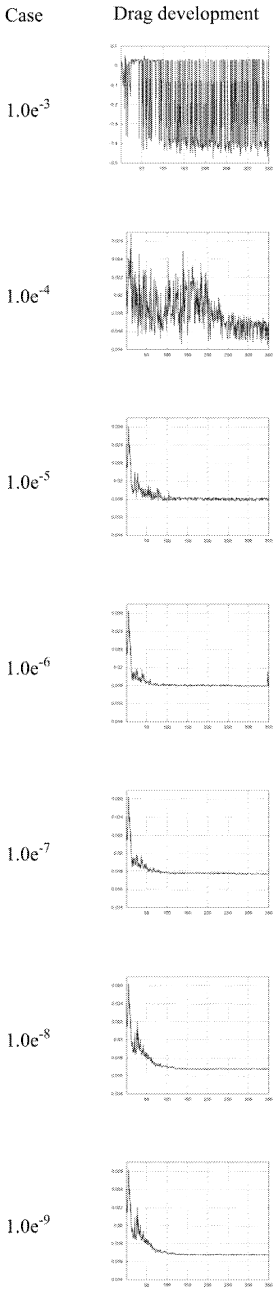


Figure 6: Objective Function for Different Residuals

The best airfoils of every optimization have been recalculated to a residual of about 10^{-9} in order to get the correct lift and drag for them. The values in Tab. 1 indicate an interesting trend. With increased convergence the resulting solution, i.e. drag minimization, is also improved. The airfoils with the best performance were found in the case that the residual was very small. Even in the case of 10^{-7}, where the forces are quite stable as seen in Fig. 2 and 3, the resulting drag is larger than in the case of 10^{-8} or 10^{-9}, indicating the influence of accuracy during the optimization process. Because of the varying und unpredictable grid quality due to the automatic grid generation process, and therefore the changes of the convergence properties, which cannot be guaranteed throughout an optimization process (particularly when applying stochastic) means some kind of real uncertainty within the design process, which is very difficult to get under control. The only advice that can be drawn from this numerical experiment is, to converge the solution as much as possible respectively until the force values do not change significantly any more, and to inspect all solutions of the optimization process to see, whether they are converged. Another possibility may be to introduce the convergence rate into the objective function, allowing the optimization strategy to avoid problematic geometries.

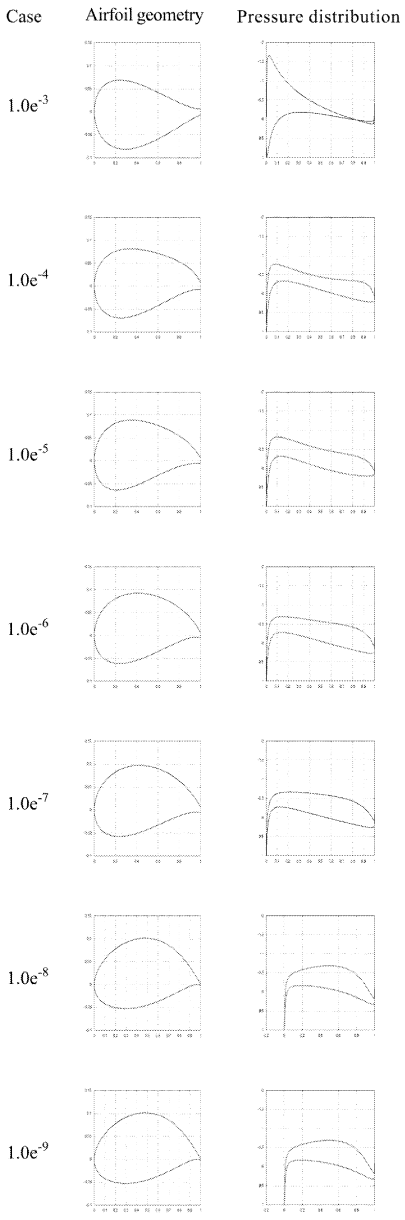


Figure 7: Development of the Geometry and Pressure Distribution for Different Residuals

Table 1: Computed Drag and Lift in Dependence on the Residual

Case	Best dataset	cd	cl	Converged cd	Converged cl
1.0E-3	218	-4.750E+03	-4.119E+03	2.421E-02	4.509E-01
1.0E-4	355	1.444E-02	5.094E-01	1.802E-02	4.066E-01
1.0E-5	282	1.779E-02	5.001E-01	1.830E-02	4.876E-01
1.0E-6	159	1.790E-02	4.959E-01	1.793E-02	4.950E-01
1.0E-7	268	1.766E-02	4.940E-01	1.766E-02	4.941E-01
1.0E-8	340	1.669E-02	4.935E-01	1.669E-02	4.935E-01
1.0E-9	340	1.669E-02	4.935E-01	1.669E-02	4.935E-01

EFFECT OF INAPPROPRIATE DESIGN PROCEDURES

Recently, there appeared several articles dealing with uncertainties in the context of the design optimization procedure itself. As an example the airfoil design for an aircraft wing, considering the flight speed or Mach number as an uncertainty may serve. In these cases, optimizations were carried out assuming a stochastic fluctuation of the Mach number M with the goal to reduce the drag coefficient (Duvigneau 2007), or to maximize the lift-to-drag ratio (Shimoyama 2006), for a fixed angle of incidence. The lift was only considered as a constraint, which makes it impossible to compare optimization results. The drag of an airfoil strongly depends on the generated lift, Fig. 8. If during the optimization process an airfoil will be generated, that accidentally has a larger lift at the prescribed angle of incidence due to its camber, it will certainly produce a larger drag than with the correct angle. Because of this it will be sorted out and thus reducing the chance to find the best solution. This one may have been the best one at the correct conditions. That is the first mistake here.

These researches also claim to address problems related to practice. However, during the about 30 year lifecycle, aircrafts are operating on diverse flight planes respectively altitudes, which are allotted by the air traffic control and are defined with a distance of 500 ft, allowing for a maximum variability in altitude of ± 250 ft. That means the aircraft needs to operate in a fixed altitude during one mission. Keeping in mind, that the air density is dependent on the altitude, leads to the fact that this requires diverse Mach numbers and lift coefficients. Furthermore, the pay load varies a lot from mission to mission. The lift coefficient c_L , necessary to compensate the weight W of the aircraft, is calculated by

$$c_L = W / (\rho V^2 S / 2) \quad (1)$$

where ρ denotes the air density, V the Velocity ($V = M \cdot a$, with a being the speed of sound) and S the wing area. Equation (1) can be transposed to the following form:

$$\rho = W / (c_L V^2 S / 2) \quad (2)$$

Even considering a constant lift coefficient (and not only a constraint on minimum lift) makes evident, that the flight altitude needs to vary due to changes in the Mach number, which is only possible in a small range, or the lift coefficient needs to be adjusted, which in turn changes the minimum lift constrained, which was not considered.

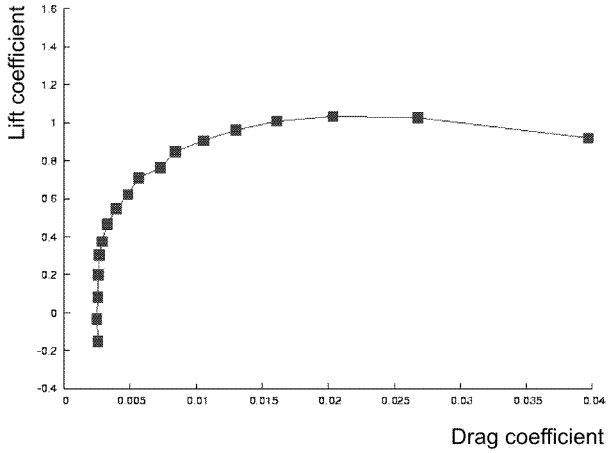


Figure 8: Dependence of Drag on Lift

Based upon the ICAO standard atmosphere the allowable Mach changes for an Airbus A340 can be calculated. The possible range of the Mach number for a prescribed flight plane is from *0.824* up to *0.834* and by far less than that assumed in the mentioned studies, Fig. 9. Furthermore, a varying Mach number requires the lift coefficient to be adjusted in order to keep the altitude, which is not considered there. This is far away from real life. In practice, this problem is addressed by aircraft manufacturers by a so-called multipoint design, where multiple operating conditions are considered. This leads to a compromise for all design points and therefore automatically covers possible fluctuations of the Mach number, but in a much wider and more appropriate way.

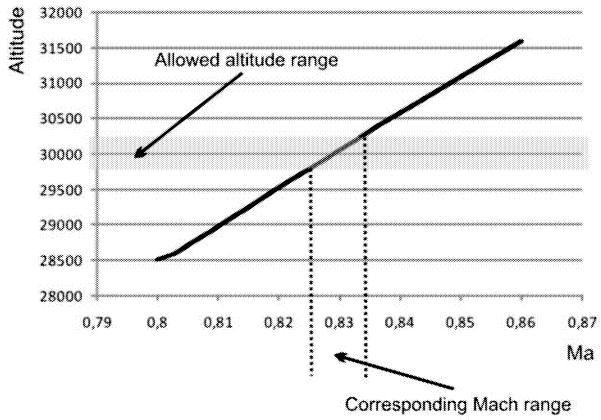


Figure 9: Correlation of Altitude and Mach number

The result for the proposed optimizations with a fluctuating Mach number was, that the best airfoils perform slightly worse in design condition, but much better in off-design conditions. This is not surprising, because off-design points are considered by fluctuating the Mach number, but in an arbitrary and inappropriate way. In contrast, considering operational aspects by doing a multipoint design should yield a comparable result, but with a more reasonable approach.

In order to assess the impact of a multipoint design in contrast to a single point design, several flight conditions have been considered. As an example, the maximum take-off

weight of an A340 is *368t,* while the maximum landing weight is *259t*. This yields an average weight during flight of about *310t*, which is the basis for the single point design for a standard flight height of 30,000 ft. On the other hand, different weights and altitudes were considered within an optimization for an airfoil, resulting in varying lift coefficients, Equation (1). Details are shown in Fig. 10 and Tab. 2.

Table 2: Design Points for an Airbus A340

Design point	Aircraft weight [t]	Altitude [ft]	Mach number [-]	Lift coefficient [-]
1	310	30.000	0.830	0.480
2	360	30.000	0.830	0.560
3	259	30.000	0.830	0.400
4	310	35.000	0.830	0.605
5	310	30.000	0.800	0.520
6	310	30.000	0.860	0.447

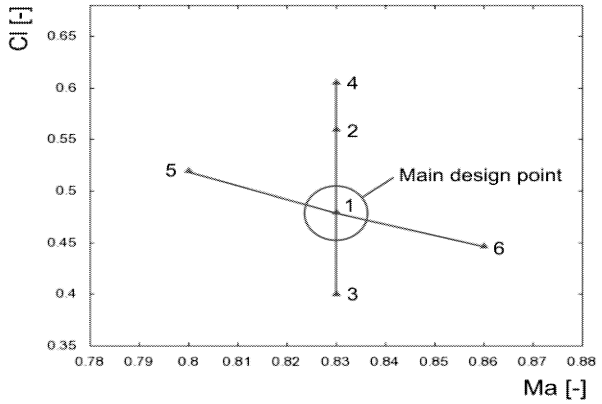


Figure 10: Design Points for an Airbus A340 shown as Lift and Mach Number Dependence

Optimizations were carried out for the single-point (design point 1) and the multi-point case, using the deterministic Downhill-Simplex method after Nelder-Mead in order to ensure comparability. The lift coefficients were attained with an accuracy of at least 2% in contrast to just considering lift as a constraint. From the results, it is obvious that the multipoint airfoil performs only 1% less with respect to drag in the lower off-design Mach range than the single point airfoil. Moreover, it performs up to 8% better in the higher Mach range, Fig. 11. This makes very clear that a multi point design already covers possible fluctuations in the Mach number and considering them as an uncertainty does not make sense. Furthermore, by just considering random or at least arbitrary fluctuations, prevents an optimal design which is exactly adjusted to the products mission.

It is obvious that considering speed fluctuations as uncertainties is not only far from practice, it is also a waste of time. Furthermore, comparing airfoils with respect to drag having different lift coefficients, like in the studies mentioned, is useless, because drag depends on lift. That may yield a bad assessment of a potentially good airfoil just only because it has been calculated with an inappropriate angle of attack. This is a clear example of an alleged uncertainty, stemming from a lack of knowledge and experience.

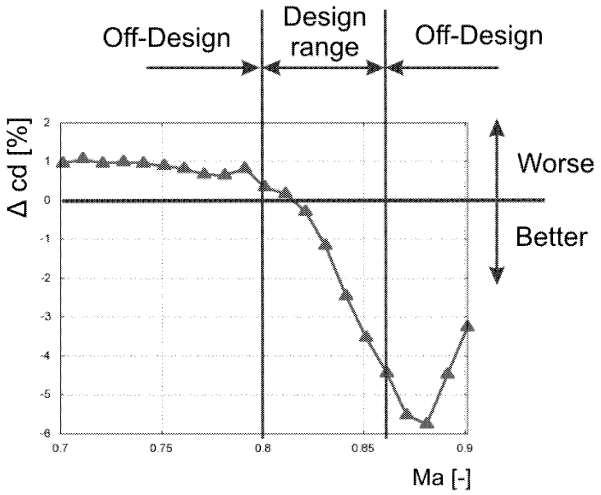


Figure 11: Relative Drag of the Multi-point Airfoil Compared the Single-point Airfoil

IMPONDERABILITY OF DESIGN GOALS

Design objectives are generally defined by human beings through an objective function, depending on several variables respectively criteria. These functions create a hyper-dimensional surface, where a minimum or maximum is predefined, although unknown a priori. Due to the fact that the setup of these functions is more or less arbitrary, and human beings are different and have different experiences, there exists some kind of uncertainty here. The impact of the objective function onto the optimization result has been discussed in several articles (Frommann 1998, Frommann 2000). As an example the simple design of a beam may serve, which is shown in Fig. 12.

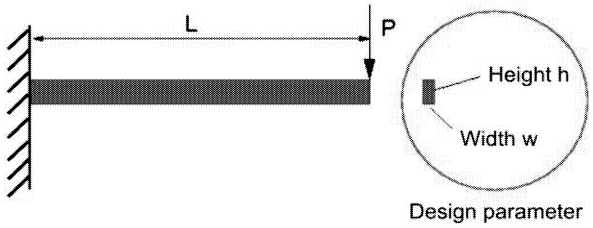


Figure 12: Beam Optimization Example

This beam is loaded by the force *P,* leading to some stress at the mounting location. The goal is to maximize the profit (length L) and concurrently to minimize the costs (volume V). These qualities are simply calculated, i.e. the volume by

$$V = h \cdot w \cdot L \quad (3)$$

and the length in dependence on the allowable stress at the mounting location and the moment of inertia by

$$L = \sigma_{allowed} \cdot w \cdot h^2 / (6 \cdot P) \quad (4)$$

A usual form of the objective function is a weighted sum with the weights w_1 and w_2:

$$min\, f = w_1 V - w_2 L \quad (5)$$

An unanswered question is how to choose the weighting factors. Tab. 3 lists three slightly different weighting combinations and Fig. 13 shows the resulting solution space topographies.

Table 3: Weighting Combinations for the Beam Objective Function

	Case 1	Case 2	Case 3
V	**50%**	**60%**	**40%**
L	**50%**	**40%**	**60%**

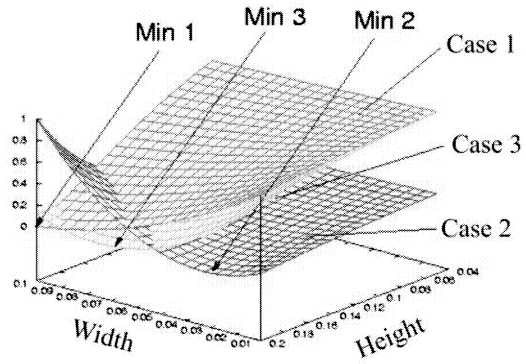


Figure 13: Solution Space Topographies for the Weighting Combinations

Different objective functions with varying weighting factors (more or less arbitrarily chosen) obviously lead to different results. With respect to some researches (Duvigneau 2007), where weighting factors are introduced within the objective function for the mean value and the variance, it is not astonishing that the results differ significantly in dependence on these factors. Changing the objective function means changing the shape of the solution space, with the result of different optima. In contrast to reducing uncertainty, this procedure introduces additional, human related uncertainty. Calling this a robust design is at least questionable. Uncertainty here again comes from a knowledge deficiency.

A solution to this problem may be the application of Fuzzy Logic, introduced by Zadeh (Zadeh 1965) for the definition of the optimization goal (Frommann 1998, Frommann 2000). This approach exploits the capabilities of human beings, i.e. working on two-dimensional dependencies and assessing dependencies in a natural way. Starting from a current solution one is able to define, in which direction each criterion improves. These trends can be represented by so-called membership functions, which describe the grade of membership, i.e. how much does a criterion belong to the desirable, tolerable or unacceptable solutions. Based upon this the user can define logic rules like: *If criterion1 is desirable and criterion2 is unacceptable than the solution is unacceptable*. This way, feelings can be converted into an exact mathematical function, describing the wishes for the design. Modern optimization software (CAOtec 2010) like ***CAOne***® offers the possibility to define these in a simple manner, Fig. 14.

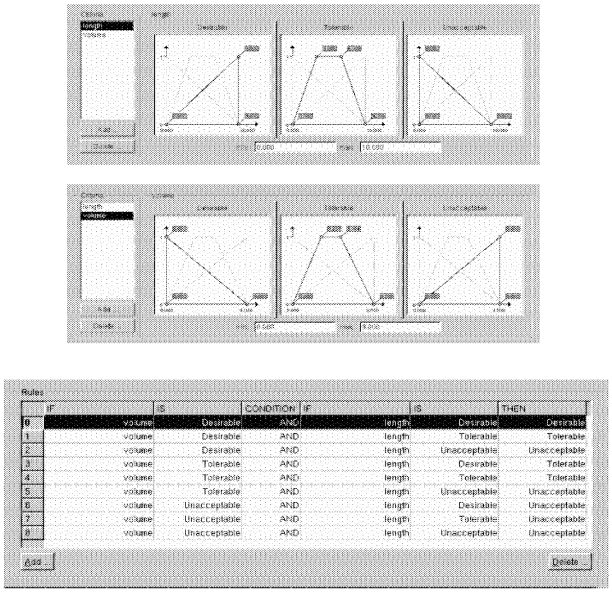


Figure 14: Definition of Membership Functions and Logic Rules for the Length and Volume within ***CAOne*** ®

Applying this methodology to the above mentioned problem of the beam yields the result depicted in Fig. 15. Now it becomes clear that, in this case, the goal was to optimize both, profit and costs to the same extend. This clearly shows that, by definition of the objective function in a more or less arbitrary way by applying weighting factors, some kind of uncertainty is introduced into the optimization process. It can be avoided by utilization of sophisticated approaches like Fuzzy Logic based objective functions, which yield a superior solution, even compared to Pareto-fronts. The latter can only be an aid in case one does not know where to go to, and, furthermore, this is limited to two or three criteria due to the limit of human spatial sense. Therefore, design objectives may be a source of uncertainty, which is worth to be considered.

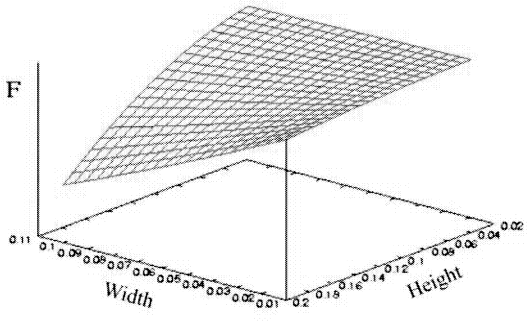


Figure 15: Solution Space Topography for the Fuzzy Logic Objective Function

CONCLUSIONS

Uncertainties are possible due to diverse reasons, which have been discussed in this article. Some of them are just alleged ones and only consume time and effort, if considered. Others, like convergence or manufacturing uncertainties, need to be handled. It has been shown that different convergence criteria lead to different, even worse results. While these problems can be overcome in a rather simple way, just by increase of computational effort, manufacturing uncertainties should be included into optimization process. A straight forward approach may be the random distortion of the geometry during the design process, representing stochastic fluctuations of material properties and manufacturing variability. This is currently part of researches by the author. Uncertainties with respect to the design goal are rather difficult to assess, because they stem from the feelings of human beings, and therefore elude themselves to some extend from being describable. The proposed Fuzzy Logic approach is a promising way out of this dilemma.

REFERENCES

CAOtec Software GmbH. 2010. "CAONE Users guide and tutorial". www.caotec.com.

Duvigneau, R. 2007. "Robust Design of a Transonic Wing with Uncertain Mach Number". *Evolutionary Methods for Design, Optimization and Control*, CIMNE, Barcelona, Spain.

Frommann, O. 1998. "Conflicting Criteria Handling in Multiobjective Optimization Using the Principles of Fuzzy Logic". *Applied Aerodynamics Conference*, Albuquerque, NM, AIAA Paper 98--2730.

Frommann, O. 2000. "Bewertung multipler und gegensätzlicher Qualitätskriterien in Multidisziplinärer Optimierung". *Proc. Deutscher Luft- und Raumfahrtkongress*, Leibzig.

Shimoyama, K. 2006. "Robust Aerodynamic Airfoil Design Optimization Against Wind Variations for Mars Exploratory Airplane". , *IAC-06-A3.P3.07, 57th International Astronautical Congress*, Valencia.

Zadeh, L.A. 1965. "Fuzzy Sets". *Information and Control, 8,* 338-352.

AUTHOR BIOGRAPHY

OLAF FROMMANN studied Aerospace Technology at the Technical University Braunschweig, Germany, with the focus on aerodynamics and aircraft design. Afterwards, from 1991 to 1996 he was employed as a member of the research staff at the Technical University Hamburg-Harburg. There, he researched high speed pipe flows with wave propagations, causing safety valve installations to flutter with the result of destruction. After receiving his Phd in 1996 he went to Airbus in Bremen, where he developed the core of a numerical optimization system for the design of wings for transonic aircrafts based upon solvers of the Reynolds averaged Navier-Stokes equations (DLR FLOWer code). During this time the Fuzzy Logic approach for objective functions was discovered and successfully applied to several problems. In 1998 he founded his own company, the Synaps Ingenieur-Gesellschaft mbH in Bremen. The core business of this company was the development of a general purpose optimization framework, called *SynapsPointer Pro*. Besides, many engineering services have been conducted, e.g. for Airbus and McLaren Formula 1. The software has been sold to CAOtec Software GmbH in 2005 and is now distributed and further developed under the brand *CAOne*®. In 2003 he became Professor for Aerospace Technology at the University of Applied Sciences Bremen. The main area of his current work is aerodynamics, aircraft design, Computational Fluid Dynamics (CFD) and Computer Aided Optimization (CAO). He was the first to establish a lecture on CAO within the study course Computer Based Mechanical Engineering (CBME).

During the last 12 years many R&D projects have been successfully finished, e.g. MEGADESIGN, AEROSHAPE. Currently, he works on a research project funded by the German Federal Ministry of Education and Research, focusing on the optimization of very thick airfoils (greater 35% relative thickness) for rotors of horizontal axis wind energy turbines. Other areas of research address the consideration of manufacturing uncertainties within optimization processes, improvement of grid and geometry generation.

COMPARATIVE SIMULATION METHODOLOGY

COMPARISON OF FORMAL MODELS FOR PROCESSES WITH SCATTERED INTER-DEPENDENCIES

Šárka Květoňová and Dušan Kolář
Department of Information Systems,
Faculty of Information Technology, Technical University of Brno
Božetěchova 2, Brno 612 66, Czech Republic
E-mail: {kvetona, kolar}@fit.vutbr.cz

KEYWORDS
Process Management, Scattered Context Grammars, Petri Nets, Context Dependencies, Parsing, Comparision of Models

ABSTRACT

The paper deals with comparison of two models used for modeling of processes. We used scattered context grammars Greibach and Hopcroft (1969) to model situation, when there are several sub-processes consisting of several actions. Now, we investigate Petri net Girault (2003), Rozenberg (1991) model's features and compare them with grammar ones. We focus on five main properties of the models that we compare.

INTRODUCTION

Business processes are composed from various activities that are strongly connected with particular tasks, task resolution, respectively. Process flow may be either fixed for all similar occurrences (e.g. tire pressure adjusting) or it is possible to have various possibilities of process flow even if service delivered is the same.
This paper presents Petri nets approach for such a formal description. On the other hand, the description via scattered context grammars (SCG) is easy to define (Kolář and Květoňová (2010)). Such a grammar description can be used efficiently in several ways. One way is for generation of all correct possibilities (see Kolář (2009), Kolář and Květoňová (2009)). The other possibility is to use the description to analyze and, partially, predict the process flow when the activities are planned according to actual availability of resources and not prior to service delivery. Similar features can be observed in Petri net description as well.
Any formal description and processing of the commitment of a certain process/subprocess/activity increases safety and reliability not just of the process but of involved actors as well. The particular algorithm used for construction of an SCG used for modeling of all possible correct process flows is described in Kolář and Květoňová (2010).
To model the situation, we stick to car repair station without any loss of generality, just to express the situation. In case of Petri net model, we use the P/T Petri net model (black and white) where places will model state of process execution and transitions synchronize and order particular subprocesses. Petri nets offer more compact and much readable description than finite automata that could be used for the same purpose by straightforward "enumeration" of all possible execution plans. This fact will be very easy to observe and provided with no proof. This is similar observation as relation between scattered context grammars and regular grammars Kolář (2009), Kolář and Květoňová (2009).

CONTRIBUTION

In this section, we describe construction of a Petri net for particular case (detailed grammar construction can be found in Kolář and Květoňová (2010)). Next, we compare both models from several viewpoints.

CAR REPAIR STATION

Business process modeling and basic motivation on transformation to formal models can be found in Kolář and Květoňová (2010) and in references cited therein.
Our use case is based on a simple car repair process in simplified car repair station. A car repair process is performed on the basis of customer's requirement that is recorded. After that, a car is taken into the car repair service and it is prepared for a repair process initialization. The own reparation is executed by a mechanic or engineer from the relevant department. Each department has just one mechanic in charge and he/she is fully responsible for it (reparation execution and its final quality). If a car reparation requires some external inputs (materials, specific services or car spare parts), a mechanic must ensure their commitment (e.g. tire service, technical inspection, upholstery works etc.).
During a reparation, a stockman draws up material requisitions which are approved by a supervisor (manager of a car repair service). He/she checks meeting the conditions, too. In particular, time keeping and cost. In case of any variances or difficulties, a supervisor informs a customer and he/she requires an agreement from

him/her.
We take into account a car repair station with four basic services. Every service can provide its own particular services, for example, the following way:

- A: tire services
  - $< a1 >< a2 >< a3 >$: tire change — consists from three tightly connected activities, removal of tire from wheel, put on a new tire, balance and pressure the wheel
  - $<a4> <a5> <a6>$: tire repair
  - $<a7> <a8>$: wheel change
- B: engine services
  - $<b1> <b2>$: ignition cable change
  - $<b3> <b4>$: ignition spark change
  - $<b5> <b6> <b7> <b8>$: oil change
- C: chassis services
  - $<c1> <c2> <c3>$: geometry setup
  - $<c4> <c5> <c6> <c7>$: absorber change
  - $<c8> <c9>$: suspension change
- D: hand car wash service
  - $<d1> <d2> <d3> <d4>$: body wash
  - $<d5> <d6> <d7>$: interior wash

Besides tight coupling of some activities, there are natural long-distant context limits. For instance *(a)* wash services are always the last ones, and *(b)* suspension change must be followed by a geometry setup.

MODELING VIA PETRI NETS

Broader analysis can be found in Kolář and Květoňová (2010), we start directly with description of building a Petr net model.
Number of all possible (even erroneous) orders of activities is finite. Thus, we can describe the situation by finite automaton (every finite language is regular one and every regular language can be recognized by a finite automaton). Even if this is possible, it is not a feasible way as the number of all possibilities is quite large even just for correct combinations. We use Petri nets instead. They provide better synchronization features, mutual exclusion, alternative ways of processing, etc.

PETRI NET CONSTRUCTION

The Petri net is constructed from the situation description step by step.
A set of places P is initialized to all activities presented above

$$P = \{<a1>, <a2>, \ \dots \ , \ <d6>, <d7>\}$$

Next, synchronization places for all particular services are added to the set of places. These newly added places enable just single execution of the service within particular order. Set P:

$$P = P \cup \{U_{A1}, U_{A2}, U_{A3}, U_{B1}, U_{B2}, U_{B3}, U_{C1}, U_{C2}, U_{D1}, U_{D2}, U_{C13}, U_{CD}\}$$

where place U_{C13} is introduced instead of simple U_{C3} as there is sequentiality necessary between these two services. Similarly, place U_{CD} enables finishing of services only when geometry setup is placed after required car suspension change.
Finally, we add three places representing core services preparation S, washing activity preparation D, and finalization of the process, FIN. Set P:

$$P = P \cup \{S, D, FIN\}$$

Thus, set of places, P, contains $32 + 12 + 3$ places that is 47 elements.
Transitions must be defined next. First of all, transitions following every service activity are introduced and set of transitions, T, is initialized:

$$T = \{T_{a1}, T_{a2}, \ \dots \ , \ T_{d7}\}$$

Next, transitions blocking services just for single execution are added. Set T:

$$T = T \cup \{T_{UA1}, T_{UA2}, T_{UA3}, T_{UB1}, T_{UB2}, T_{UB3}, T_{UC2}, T_{UD1}, T_{UD2}, T_{C3}, T_{C1a}, T_{C1b}, T_{UCDa}, T_{UCDb}\}$$

where places with a and b indicants in their name represent variants in the service execution — T_{C1a} stays for car geometry when no suspension was change, T_{C1b} stays for car geometry after suspension change, T_{UCDa} stays for finalization of services without application of geometry service, and T_{UCDb} stays for finalization with application of geometry service.
Finally, transition that closes service delivery must be added. Set T:

$$T = T \cup \{T_F\}$$

Thus, we have 47 elements in the set T (the same value as for set P).
Flow relation is the following way, initialize F with:

$$F = \{(<a1>, T_{a1}), (<a2>, T_{a2}), \ \dots \ , (<d7>, T_{d7})\}$$

(simple flow in sub-services).
Next, completion of simple flows is added. Thus, set F:

$$F = F \cup \{(T_{a1}, <a2>), (T_{a2}, <a3>), (T_{a4}, <a5>), (T_{a5}, <a6>), (T_{a7}, <a8>), \ \dots \ , (T_{d5}, <d6>), (T_{d6}, <d7>)\}$$

Next, return from services to preparation places is added. Set F:

$$F = F \cup \{(T_{a3}, S), (T_{a6}, S), (T_{a8}, S), \ \dots \ , (T_{c8}, S), (T_{d4}, D), (T_{d7}, D)\}$$

Next, links from preparation places to transitions starting services are added. Set F to be:

$$F = F \cup \{(S, T_{UA1}), (S, T_{UA2}), \ \dots \ , (S, T_{UB3}), (S, T_{UC1a}), (S, T_{C1b}), (S, T_{UC2}), (S, T_{C3}), (D, T_{UD1}), (D, T_{UD2})\}$$

and links from transitions starting services to places representing the first sub-service of particular service are added. Thus set F:

$$F = F \cup \{(T_{UA1}, <a1>), (T_{UA2}, <a4>), (T_{UA3}, <a7>), (T_{UB1}, <b1>), (T_{UB2}, <b3>), (T_{UB3}, <b5>), (T_{C1a}, <c1>), (T_{C1b}, <c1>), (T_{UC2}, <c4>), (T_{C3}, <c8>), (T_{UD1}, <d1>), (T_{UD2}, <d5>)\}$$

Next, we add links from single execution synchronization places to appropriate transitions. Set F:

$$F = F \cup \{(U_{A1}, T_{UA1}), (U_{A2}, T_{UA2}), \ \dots \ , (U_{B3}, T_{UB3}), (U_{C1}, T_{C1b}), (U_{C2}, T_{UC2}), (U_{C13}, T_{C1a}), (U_{C13}, T_{C3}), (U_{D1}, T_{UD1}), (U_{D2}, T_{UD2})\}$$

Next, links assuring proper ordering of services, in particular suspension change and geometry setup, are added. Thus, set F:

$$F = F \cup \{(T_{c9}, U_{C1}), (T_{c3}, U_{CD})\}$$

Finally, links for transition to washing services and to finalize the car repair service are added. Set F:

$$F = F \cup \{(S, T_{UCDa}), (S, T_{UCDb}), (U_{CD}, T_{UCDa}), (U_{C13}, T_{UCDb}), (T_{UCDa}, D), (T_{UCDb}, D), (D, T_F), (T_F, FIN)\}$$

We have 108 edges at all for the use case.
To finish definition of the Petri net, we have to specify capacity of places weight of edges and initial marking. Capacity of every place is 1 and weight of every edge is 1 as well. Initial marking is set the following way:

- place S contains one token—initial marking is 1—this is the starting point of service delivery, preparation of services that can be delivered at first;
- places U_{A1}, U_{A2}, …, U_{B3}, U_{C13}, U_{C2}, U_{D1}, and U_{D2} contain one token—these are places modeling at most single service delivery, including places for serialization of services;
- all other places contain no token during initialization.

SUMMARY ON PETRI NET CONSTRUCTION
Petri net construction is quite straightforward in many aspects—places representing sub-services, direct chaining of sub-services into a service, including transitions and edges. It is quite easy to detach services delivered independently before others. On the other hand, modeling requiring sequential dependency and, optional, synchronization with group of other independent services is quite tricky and the more complex the scattered dependency is, the more trickier the Petri net may be. Even if we can easily discover certain patterns in modeling of scattered dependencies, their introduction is not straightforward.

COMPARISON OF THE MODELS: SCG AND PETRI NETS

We want to study the two models from these five viewpoints:

- number of states/symbols—number of core elements necessary to define a model informs about overall size, nevertheless, small differences are not very important in this comparison;
- number of relationship description elements—edges in Petri nets and production rules in grammar define relations and, in a fact, they form functionality of the model;
- simpleness of creation from informal description—is the way of model definition straightforward or do we need some reasoning to define the model;
- modeling features—is it possible to verify that a process went correct way, can we predict certain behavior/runtime live overview;
- context bindings incorporation—we especially concentrate on scattered dependency and its easiness of introduction.

Before we get to comparison, we present a complete scattered context grammar model of the same use case, its creation algorithm is in Kolář and Květoňová (2010).

$G = (V, T, P, S)$
$T = \{<a1>, <a2>, \ \dots, \ <d6>, <d7>\}$
$V = T \cup \{S, Y, D, Z, A1, A2, \ \dots \ , C3, U_{A1} \ U_{A2} \ \dots \ U_{D2}, Z_{C1}, Z_{C3}\}$
$P = \{S \ \rightarrow \ Y \ D \ Z \ U_{A1} \ U_{A2} \ \dots \ U_{D2},$
$U_{A1} \ \rightarrow \ \varepsilon,$
$\vdots$
$U_{D2} \ \rightarrow \ \varepsilon,$
$D \ \rightarrow \ \varepsilon$
$(D, U_{D1}) \ \rightarrow \ (<d1> <d2> <d3> <d4> \ D, \varepsilon)$
$(D, U_{D2}) \ \rightarrow \ (<d5> <d6> <d7> \ D, \varepsilon)$
$A1 \ \rightarrow <a1> <a2> <a3>$
$A2 \ \rightarrow <a4> <a5> <a6>$
$\vdots$
$C3 \ \rightarrow <c8> <c9>$
$Y \ \rightarrow \ \varepsilon$
$(Y, U_{A1}) \ \rightarrow \ (A1 \ Y, \varepsilon)$
$(Y, U_{A2}) \ \rightarrow \ (A2 \ Y, \varepsilon)$
$\vdots$
$(Y, U_{B3}) \ \rightarrow \ (B3 \ Y, \varepsilon)$
$(Y, Z, U_{C1}) \ \rightarrow \ (C1 \ Y, Z_{C1} \ Z, \varepsilon)$
$(Y, U_{C2}) \ \rightarrow \ (C2 \ Y, \varepsilon)$

Table 1: Comparison results summary

Formalism	SCG	Petri Nets
Number of states/symbols	58	94
Number of relation descriptors	35	108
Simpleness of creation	very good	good
Modeling features	good	very good
Context bindings incorporation	very good	good

$$(Y, Z, U_{C3}) \rightarrow (C3\ Y, Z_{C3}\ Z, \varepsilon)$$
$$Z \rightarrow \varepsilon$$
$$Z_{C1} \rightarrow \varepsilon$$
$$(Z_{C3}, Z_{C1}) \rightarrow (\varepsilon, \varepsilon)$$

We have 32 terminals, 26 non-terminals, and 35 propagation rules.

Comparison results are summarized in Table 1.

SUMMARIZED ADVANTAGES AND DISADVANTAGES OF THE MODELS

As the Table 1 shows, number of terminals plus number of non-terminals is lower than number of places plus number of transitions. Even if some transitions and places are introduced really formally, the same holds for terminals and it seems that total number of building components is lower for SCGs.

Similar relation is for propagation rules and edges. Again, even if some edges are really formal ones, number of edges probably always exceeds number of propagation rules a lot.

Simpleness of creation is almost the same for both models. Unfortunately, Petri net has more formal parts and, thus, it is more resource consuming during creation. Nevertheless, overall performance of both is nice.

Both formal models can decide that a process went OK (grammar a bit easier), both models can provide all possible correct cases (again, grammar a bit easier), but in run-time (live) checking is Petri net better.

Incorporation of context bindings is simpler in grammar. We have to add two extra rules, extend two other, while working with two extra non-terminals. Having three or more sub-services in similar relation is not a problem, just instead of 2 elements we will have 4. Moreover, detaching the washing services to the end of the process is straightforward. The same task for Petri net is not impossible, nevertheless, it is not that straightforward and it is more tricky. We also have to add helping structure, but it does require copying on detaching part of washing services. Overall evaluation for this item is slightly worse for Petri nets.

CONCLUSION

Petri nets are a very powerful tool having many useful properties (synchronization, parallel processing, resource modeling, timing etc.), but for our purpose, they are too complex and sophisticated and we do not use all aspects/means of them. Therefore, we chose Scattered Context Grammars. SCG enable fully sufficient and compact description of problems described above with algorithmic derivation from a very compact definition. Moreover, following the features required by parser of SCG, we can easily (= algorithmically) derive a parser that can validate a sequence of activities/sub-processes (see, Kolář (2008)). A simple modification of the parser can help us in prediction of what activities/sub-processes should be committed for the particular process in the future, as they have not been completed yet.

ACKNOWLEDGEMENT

This work was partially supported by the BUT FIT grant FIT-10-S-2 and the research plan No. MSM0021630528.

REFERENCES

Girault C., 2003. *Petri Nets for Systems Engineerin.* Springer. ISBN 3540412174.

Greibach D. and Hopcroft J., 1969. *Scattered Context Grammars. Journal of Computer and System Sciences*, , no. Vol 3, 233–247.

Kolář D., 2008. *Scattered Context Grammars Parsers.* In *Proceedings of the 14th International Congress of Cybernetics and Systems of WOCS.* PWR WROC. ISBN 978-83-7493-400-8, 491–500.

Kolář D., 2009. *Exploitation of Scattered Context Grammars to Model Constraints between Components.* In *Proceedings of 31st Autumn International Colloqium ASIS 2009, Advanced Simulation of Systems.* MARQ. ISBN 978-80-86840-47-5, 13–18.

Kolář D. and Květoňová Š., 2009. *Optimization of Car Repair Processes by Scattered Context Grammars Application.* In *The 2009 European Simulation and Modelling Conference.* EUROSIS. ISBN 978-90-77381-52-6, 146–149.

Kolář D. and Květoňová Š., 2010. *Process Modeling & Optimization of Complex Systems by Scattered Context Grammars.* In *Proceedings of The Seventh International Conference on Engineering Computational Technology.* in print, 12.

Rozenberg G., 1991. *Advances in Petri Nets.* Springer. ISBN 0387538631.

Comparing two sampling methods in Monte Carlo simulation

Megdouda OURBIH-TARI and Sofia GUEBLI
Laboratory of Applied Mathematics
Department of Mathematics
University of Bejaia
Algeria

Abstract

This paper gives a generalization of the use of refined descriptive sampling (RDS) method for K input variables. The estimate of RDS is shown to be unbiased and a comparison of RDS and random sampling variances is also given.

Keywords: Sampling method; Variance; Monte Carlo, Expectation.

1 Introduction

Suppose we have some device, the behavior of which depends on a random vector $\mathbf{X} = (X_1, X_2, ..., X_K)$ of fixed length K with known probability density function $f(x)$ and known cumulative function $F(x)$ for $x \epsilon \mathbb{R}^K$. A mathematical model for the device is developed from which we can simulate the behavior of the device on a computer. So experiments are carried out on the model built and unknown parameter θ of the output random variable Y of interest denoted as the unknown but observable univariate transformation of $\mathbf{X}$ given by the function $Y = h(\mathbf{X})$ is estimated. Thus, we have the problem of approximating θ.

Since $h(\mathbf{X})$ may be difficult to compute for each new value of $\mathbf{X}$, it is important to pick a sampling scheme that allows us to estimate $h(\mathbf{X})$ well while keeping N, the number of replication, to a minimum. A lot of methods for choosing $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_N$ exist. The simplest is simple random sampling which is used to generate N iid random vectors with the distribution of $\mathbf{X}$. Tari and Dahmani [7] suggested an alternative method of generating $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_N$ that they call Refined Descriptive Sampling (RDS). Let T_{RS} denote the estimate of θ using random sample of size N and T_{RDS} denote the estimate of θ using refined descriptive sample of size N. In section 2, we describe random sampling and give the considered class of estimators. In section 3, RDS procedure is described and its generalization for K input random variables is proposed in section 4. In section 5, the estimator T_{RDS} is shown to be unbiased and efficient by studying its variance which is proved to be less than that obtained with simple random sampling.

2 Estimators

In this paper, RDS is examined and compared to random sampling with respect to the class of estimators of the form

$$\mathbf{T} = \mathbf{T}(Y_1, Y_2, ..., Y_N) = \frac{1}{N} \sum_{j=1}^{N} g(Y_j),$$

where $g(.)$ is an arbitrary known function.

If $g(Y) = Y$ then $\mathbf{T}$ represents the sample mean which is used to estimate $E(Y)$. If $g(Y) = Y^{\,r}$ we obtain the r^{th} sample moment. By letting

$$g(Y) = \begin{cases} 1 & \text{for } Y \leq y \\ 0 & \text{otherwise} \end{cases},$$

we obtain the usual empirical distribution function at the point y. Our interest is focused around these particular statistics to compare random sampling (RS) and refined descriptive sampling methods.

It is often impractical or impossible to use deterministic methods when the number of variables K is large. Hence Monte Carlo methods [1] are usually used for high-dimensional problems. That is, N values of the input random vector, $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_N$ are generated in some manner such that the parameter θ can be estimated by

$$T_{RS} = \mathbf{T}(Y_1, Y_2, ..., Y_N) = \frac{1}{N} \sum_{j=1}^{N} g(Y_j)$$

when the arguments $Y_1, Y_2, ..., Y_N$ constitute a random sample of Y.

The mean and variance of T_{RS} are denoted by θ and $\frac{\sigma^2}{N}$ where σ^2 is the variance of $g(Y)$ obtained using random sampling.

3 The proposed generalization of RDS for K input variables

RDS is suggested in order to reduce the problem of sampling bias which can be introduced by using descriptive sampling (DS) [4] in Monte Carlo simulation and by the way it eliminates the problem related to the sample size of descriptive sampling. It is concerned with a block that must be situated inside a generator aiming to distribute regular subsets of randomly chosen prime number sizes p_j, in a random order as required by the simulation. We stop the process when the simulation terminates, say when m prime numbers have been used which derives m subruns. This approach removes the second problem of DS which is the need to determine in advance the sample size N.

Either a discrete or a continuous or even a mixed distribution can be represented, provided that the respective inverse of the distribution function is available. This inverse function is always defined, although, in most cases, a numerical approximation may be necessary, as in the case of a normal distribution [3].

3.1 Sample space

Let the sample space Ω of $\mathbf{X}$ be partitioned into m disjoint K dimensional space A_j of size P_j^K where each A_j is again partitioned into P_j^K disjoint K dimensional hypercubes labelled by $S_{i+\sum_{q=1}^{j-1} P_q^K}$

$i = 1, 2, ..., P_j^K$ and $\quad j = 1, 2, ..., m$ with the convention $\sum_{q=1}^{0} P_q^K = 0$.

In the j^{th} subrun, we have P_j^K possible outcomes and in a simulation experiment defined by m subruns, we have then, $\sum_{q=1}^{m} P_q^K$ possible outcomes of K dimensional vectors $\mathbf{r}_{i+\sum_{q=1}^{j-1} P_q^K}$

Finally, the sample space Ω of $\mathbf{X}$ can be written by

$$\Omega = \cup_{j=1}^{m} A_j = \cup_{j=1}^{m} \cup_{i=1}^{p_j^K} S_{i+\sum_{q=1}^{j-1} P_q^K}$$

3.2 Set values generation

In the center of each hypercube $S_{i+\sum_{q=1}^{j-1} P_q^K}$ we can find a K dimensional vector $\mathbf{r}_{i+\sum_{q=1}^{j-1} P_q^K}$ of regular numbers where each component is defined by

$$r_{l+\sum_{q=1}^{j-1} P_q} = P_j^{-1}(l - 0.5), l = 1, 2, ..., P_j.$$

Let $\mathbf{X}_{l+\sum_{q=1}^{j-1} P_q}$ be the l^{th} simulated value of the j^{th} sub-run and $\mathbf{R}_{l+\sum_{q=1}^{j-1} P_q}$ be the l^{th} vector of the j^{th} sub-run, where each component is randomly selected from the following K same subsets of regular numbers

$$\left\{ r_{1+\sum_{q=1}^{j-1} P_q}, r_{2+\sum_{q=1}^{j-1} P_q} ..., r_{\sum_{q=1}^{j} P_q} \right\}^K$$

These subsets $\left\{ r_{1+\sum_{q=1}^{j-1} P_q}, r_{2+\sum_{q=1}^{j-1} P_q} ..., r_{\sum_{q=1}^{j} P_q} \right\}$ of prime size p_j are uniformly distributed between $[0, 1[$ and $\mathbf{R}_{l+\sum_{q=1}^{j-1} P_q}$ is situated in the center of the hypercube.

In refined descriptive sampling, the subset values are generated as required by the simulation and a refined descriptive sample vectors of size N is defined for the input random vector $\mathbf{X}$ using the inverse transform method as successive samples vectors of size P_j, $j = 1, 2, ..., m$ such as

$$\mathbf{X}_{l+\sum_{q=1}^{j-1} P_q} = F^{-1}\left(\mathbf{R}_{l+\sum_{q=1}^{j-1} P_q}\right) \text{ where } l = 1, 2, ..., P_j$$

3.3 Bernouilli random variables

To have the appropriate form of the estimator T_{RDS}, we introduce $\sum_{q=1}^{m} p_q^K$ Bernouilli random variables

$$\begin{aligned} w_{i+\sum_{q=1}^{j-1} P_q^K} &: S_{i+\sum_{q=1}^{j-1} P_q^K} \to R \\ \text{where } i &= 1, 2, ..., p_j^K \text{ and } \quad j = 1, 2, ..., m \end{aligned}$$

such as

$$w_{i+\sum_{q=1}^{j-1} P_q^K} = \begin{cases} 1 & \text{if } \{\mathbf{X}_{l+\sum_{q=1}^{j-1} P_q} \\ & \text{used } \mathbf{r}_{i+\sum_{q=1}^{j-1} P_q^K} \in S_{i+\sum_{q=1}^{j-1} P_q^K}\} \\ 0 & \text{otherwise} \end{cases}.$$

where each random variable having the following properties which are immediate:

1. $p(w_{i+\sum_{q=1}^{j-1} P_q^K} = 1) = \frac{1}{P_j^{k-1}}$
$= E\left(w_{i+\sum_{q=1}^{j-1} P_q^K}\right) = E\left(w^2_{i+\sum_{q=1}^{j-1} P_q^K}\right)$
2. $Var(w_{i+\sum_{q=1}^{j-1} P_q^K}) = \left(\frac{1}{P_j^{k-1}}\right)\left(1 - \frac{1}{P_j^k - 1}\right).$
3. $E(w_{i+\sum_{q=1}^{j-1} P_q^K} \times w_{t+\sum_{q=1}^{j-1} P_q^K}) = \frac{1}{P_j^{K-1}(P_j-1)^{K-1}}$ if $i \neq t$
4. $E(w_{i+\sum_{q=1}^{j-1} P_q^K} \times w_{t+\sum_{q=1}^{j-1} P_q^K}) = 0 \quad$ if $i = t$

3.4 RDS estimate

Finally, $\sum_{q=1}^{m} p_q$ values of the input refined descriptive vectors,

$$\mathbf{X}_1, .., \mathbf{X}_{p_1}, \mathbf{X}_{1+p_1}, .., \mathbf{X}_{p_2+p_1}, .., \mathbf{X}_{1+\sum_{q=1}^{m-1} p_q}, .., \mathbf{X}_{\sum_{q=1}^{m} p_q}$$

are generated such that the parameter θ can be estimated in each sub-run by the following m estimates

$$\overset{\Lambda}{\theta}_1 = \mathbf{T}(Y_1, Y_2, ..., Y_{p_1}) = \frac{1}{p_1}\sum_{i=1}^{p_1^K} w_i \times g(y_i)$$

$$\begin{aligned}\overset{\Lambda}{\theta}_2 &= \mathbf{T}(Y_{1+p_1}, Y_{2+p_1}, ..., Y_{p_2+p_1}) \\ &= \frac{1}{p_2}\sum_{i=1}^{p_2^K} w_{i+p_1^K} \times g(y_{i+p_1^K}) \\ &\vdots \\ \overset{\Lambda}{\theta}_m &= \mathbf{T}\left(Y_{1+\sum_{q=1}^{m-1} p_q}, Y_{2+\sum_{q=1}^{m-1} p_q}, ..., Y_{\sum_{q=1}^{m} p_q}\right) \\ &= \frac{1}{p_m}\sum_{i=1}^{p_m^K} w_{i+\sum_{q=1}^{m-1} p_q^K} \times g(y_{i+\sum_{q=1}^{m-1} p_q^K})\end{aligned}$$

and in the simulation experiment, by the following T_{RDS} estimate defined by the average of those subruns estimates

$$\begin{aligned}T_{RDS} &= \mathbf{T}(\overset{\Lambda}{\theta}_1, \overset{\Lambda}{\theta}_2, .., \overset{\Lambda}{\theta}_m) \\ &= \frac{1}{\sum_{q=1}^{m} p_q}\sum_{j=1}^{m} p_j \times \overset{\Lambda}{\theta}_j \\ &= \frac{1}{\sum_{q=1}^{m} p_q}\sum_{j=1}^{m}\sum_{i=1}^{p_j^k} w_{i+\sum_{q=1}^{j-1} p_q^K} \times g(y_{i+\sum_{q=1}^{m-1} p_q^K})\end{aligned}$$

when the arguments $\left(Y_{1+\sum_{q=1}^{j-1} p_q}, Y_{2+\sum_{q=1}^{j-1} p_q}, ..., Y_{\sum_{q=1}^{j} p_q}\right)$, $j = 1, ..., m$ constitute the m descriptive samples of size p_j of Y observed in each subrun and the arguments $Y_1, .., Y_{p_1}, Y_{1+p_1}, .., Y_{p_2+p_1}, .., Y_{1+\sum_{q=1}^{m-1} p_q}, .., Y_{\sum_{q=1}^{m} p_q}$ constitute a refined descriptive sample of size N of Y .

The convention $\sum_{q=1}^{0} p_q = 0$ still hold and suppose also that all regular numbers generated from the last prime p_m are all used by the simulation. To compare this sampling method with random sampling we assume $\sum_{q=1}^{m} p_q = N$. The mean and variance of T_{RDS} are now estimated.

Remark 1 *Given that a software component "getRDS" is designed to generate numbers using refined descriptive sampling procedure and it is fully tested for both criteria: independence and uniformity between* 0 *and* 1 *[2] then a refined descriptive sample can be considered as one in which observations are independent random variables and each one following the population distribution.*

4 Properties of RDS estimate

4.1 T_{RDS} is unbiased

It has been shown in Tari and Dahmani [7] that $Bias(T_{RDS})$ is insignificant if the underlying frequency $f_w$of the output random variable Y is different from $M \times \prod_{q=1}^{m} p_q$ where $M \in N^*$. This condition is usually verified because the simulation will certainly terminate before the product of all prime numbers used in a run or a multiple of it can be equal to the underlying frequency as this product $M \times \prod_{q=1}^{m} p_q$ has a very high frequency. If the sample size $N = \sum_{q=1}^{m} p_q \to +\infty$ then $\prod_{q=1}^{m} p_q \to +\infty$ faster and $M \times \prod_{q=1}^{m} p_q$ tends to infinity even faster and since f_w is a finite frequency, then $f_w \neq M \times \prod_{q=1}^{m} p_q$. As a consequence, $lim_{N\to+\infty}\ Bias(T_{RDS}) = 0$ then, T_{RDS} is asymptotically unbiased estimate. We can then suppose that T_{RDS} is an unbiased estimator since sample sizes are always large in a simulation study, then the mean of T_{RDS} is θ.

4.2 The variance of T_{RDS}

Theorem 2 *If $Y = h(X_1, ..., X_K)$ is monotonic in each of its arguments and if $g(Y)$ is monotonic function of Y, we have then*

$$Var(T_{RDS}) \leq Var(T_{RS})$$

Proof. Firstly, we notice that

$$P(\ \mathbf{X}_{l+\sum_{q=1}^{j-1} P_q} \text{used } \mathbf{r}_{i+\sum_{q=1}^{j-1} P_q^K} \in S_{i+\sum_{q=1}^{j-1} P_q^K}) = \frac{1}{p_j^K}$$

and the marginal density function of

$$\mathbf{X}_{l+\sum_{q=1}^{j-1} p_q} \text{given } \left\{\mathbf{X}_{l+\sum_{q=1}^{j-1} P_q} \text{ used } \mathbf{r}_{i+\sum_{q=1}^{j-1} P_q^K}\right\}$$
is $P_j^K \times f(x)$,

Let $\mu_{i+\sum_{q=1}^{j-1} p_q^K}$ to be the means of $g(Y_{i+\sum_{q=1}^{j-1} p_q^K})$ in the K dimensional hypercubes.

Let σ_j^2 to be variances of $g(Y_j)$ in the subruns defined by

$$\sigma_j^2 = \sum_{i=1}^{P_j^K} \int_{S_{i+\sum_{q=1}^{j-1} P_q^K}} (g(y_{i+\sum_{q=1}^{j-1} P_q^K}) - \theta)^2 f(x)dx \quad (1)$$

where $j = 1, ..., m$

Let σ^2 the variance of $g(Y)$ to be defined in refined descriptive sampling method by

$$\sigma^2 = var(g(Y)) = \frac{1}{N}\sum_{j=1}^{m} P_j \sigma_j^2. \tag{2}$$

Apply the independence of the subruns, the variance of the general form of T_{RDS} is written

$$\begin{aligned} & Var\,(T_{RDS}) \\ = \; & \frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{P_j^k} Var(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K})) \\ + & \frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{P_j^K}\sum_{\substack{t=1 \\ i\neq t}}^{P_j^K}[Cov(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K}), \\ & w_{t+\sum_{q=1}^{j-1} P_q^K} \times g(y_{t+\sum_{q=1}^{j-1} P_q^K}))] \end{aligned} \tag{3}$$

1) Let us calculate the first part of the the right hand side of the last equality, according to the variance properties, it follows that,

$$\begin{aligned} & \sum_{i=1}^{P_j^k} Var(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K})) \\ = \; & \sum_{i=1}^{P_j^k} E(w^2_{i+\sum_{q=1}^{j-1} P_q^K}) Var(g(y_{i+\sum_{q=1}^{j-1} P_q^K}) \\ & + \sum_{i=1}^{P_j^k} E^2(g(y_{i+\sum_{q=1}^{j-1} P_q^K})) Var(w_{i+\sum_{q=1}^{j-1} P_q^K}) \end{aligned}$$

where E design the mathematical expectation.

By virtue of the 1^{st} and 2^{nd} probability properties of $w_{i+\sum_{q=1}^{j-1} P_q^K}$, we have,

$$\sum_{i=1}^{P_j^k} Var(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K})) \tag{4}$$

$$\begin{aligned} = \; & \sum_{i=1}^{P_j^k} \frac{1}{P_j^{k-1}} Var(g(y_{i+\sum_{q=1}^{j-1} P_q^K}) \\ & + \sum_{i=1}^{P_j^k} \mu^2_{i+\sum_{q=1}^{j-1} P_q^K} (\frac{1}{P_j^{k-1}})(1 - \frac{1}{P_j^{k-1}}) \end{aligned}$$

In accordance with the variance properties, we have

$$Var(g(y_{i+\sum_{q=1}^{j-1} P_q^K}) = E(g(y_{i+\sum_{q=1}^{j-1} P_q^K}) - \mu_{i+\sum_{q=1}^{j-1} P_q^K})^2$$

By adding and reducing θ in the expectation, we obtain

$$\begin{aligned} & Var(g(y_{i+\sum_{q=1}^{j-1} P_q^K}) = E(g(y_{i+\sum_{q=1}^{j-1} P_q^K}) - \theta)^2 \\ & \qquad -(\mu_{i+\sum_{q=1}^{j-1} P_q^K} - \theta)^2 \\ = & \int_{S_{i+\sum_{q=1}^{j-1} P_q^K}} (g(y_{i+\sum_{q=1}^{j-1} P_q^K}) - \theta)^2 \; P_j^k f(x)dx \\ & \qquad -(\mu_{i+\sum_{q=1}^{j-1} P_q^K} - \theta)^2 \end{aligned} \tag{5}$$

Substituting the expression 5 in the relation 4, we write

$$\begin{aligned} & \sum_{i=1}^{p_j^k} Var(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K})) \\ = \; & P_j \sum_{i=1}^{p_j^k} \int_{S_{i+\sum_{q=1}^{j-1} P_q^K}} (g(y_{i+\sum_{q=1}^{j-1} P_q^K}) - \theta)^2 \; f(x)dx \\ & -\frac{1}{P_j^{k-1}}\sum_{i=1}^{p_j^k}(\mu_{i+\sum_{q=1}^{j-1} P_q^K} - \theta)^2 \\ & +\frac{1}{P_j^{k-1}}(1 - \frac{1}{P_j^{k-1}})\sum_{i=1}^{p_j^k} \mu^2_{i+\sum_{q=1}^{j-1} P_q^K} \end{aligned}$$

Considering 1 and using

$$\theta = \frac{1}{P_j^K}\sum_{i=1}^{P_j^K} \mu_{i+\sum_{q=1}^{j-1} p_q^K}$$

we have then,

$$\begin{aligned} & \frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{p_j^k} Var(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K})) \\ = \; & \frac{1}{N}\frac{1}{N}\sum_{j=1}^{m} P_j \sigma_j^2 \\ & -\frac{1}{N^2}\sum_{j=1}^{m}(-p_j\theta^2 + \frac{1}{p_j^{2k-2}}\sum_{i=1}^{p_j^k} \mu^2_{i+\sum_{q=1}^{j-1} P_q^K}) \end{aligned}$$

and finally taking into account the formula 2, we get

$$\begin{aligned} & \frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{p_j^k} Var(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K})) \\ = \; & \frac{\sigma^2}{N} - \frac{1}{N^2}\sum_{j=1}^{m}(-p_j\theta^2 + \frac{1}{p_j^{2k-2}}\sum_{i=1}^{p_j^k} \mu^2_{i+\sum_{q=1}^{j-1} P_q^K}) \end{aligned} \tag{6}$$

2) We calculate now the second part of the right hand side of the equality 3 , according to the covariance and $w_{i+\sum_{q=1}^{j-1} P_q^K}$ first properties, it follows that,

$$\begin{aligned}
&\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}[Cov(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K}),\\
&w_{t+\sum_{q=1}^{j-1} P_q^K} \times g(y_{t+\sum_{q=1}^{j-1} P_q^K}))]\\
= &\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}\mu_{i+\sum_{q=1}^{j-1} p_q^K} \times \mu_{t+\sum_{q=1}^{j-1} p_q^K}\\
&\times E(w_{i+\sum_{q=1}^{j-1} P_q^K} \times w_{t+\sum_{q=1}^{j-1} P_q^K})\\
&-\frac{1}{p_j^{2K-2}}\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}\mu_{i+\sum_{q=1}^{j-1} p_q^K} \times \mu_{t+\sum_{q=1}^{j-1} p_q^K}
\end{aligned}$$

By virtue of the 3^{rd} and 4^{th} probability properties of $w_{i+\sum_{q=1}^{j-1} P_q^K}$, we have,

$$\begin{aligned}
&\frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}[Cov(w_{i+\sum_{q=1}^{j-1} P_q^K} \times g(y_{i+\sum_{q=1}^{j-1} P_q^K}),\\
&\qquad w_{t+\sum_{q=1}^{j-1} P_q^K} \times g(y_{t+\sum_{q=1}^{j-1} P_q^K}))] \qquad (7)\\
= &(P_j-1)^{-K+1} \times P_j^{K+1}\\
&\times\frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}\mu_{i+\sum_{q=1}^{j-1} p_q^K} \times \mu_{t+\sum_{q=1}^{j-1} p_q^K}\\
&-P_j^{-2K+2}\\
&\times\frac{1}{N^2}\sum_{j=1}^{m}\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}\mu_{i+\sum_{q=1}^{j-1} p_q^K} \times \mu_{t+\sum_{q=1}^{j-1} p_q^K}
\end{aligned}$$

Substituting 6 and 7 in 3, we write,

$$\begin{aligned}
&Var(T_{RDS})\\
= &Var(T_{RS})\\
&+\frac{1}{N^2}\sum_{j=1}^{m}\frac{(P_j-1)^{-K+1}}{P_j^{-K+2}}\\
&\times\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}(\mu_{i+\sum_{q=1}^{j-1} p_q^K}-\theta) \times (\mu_{t+\sum_{q=1}^{j-1} p_q^K}-\theta)
\end{aligned}$$

Indeed,

$$\sum_{i=1}^{p_j^K}\sum_{\substack{t=1\\ i\neq t}}^{p_j^K}(\mu_{i+\sum_{q=1}^{j-1} p_q^K}-\theta) \times (\mu_{t+\sum_{q=1}^{j-1} p_q^K}-\theta) \leq 0$$

using Lehmann's theorem and Hoeffding's equation, we then obtain the result. ■

5 Conclusion

We developed the use of refined descriptive sampling for more than one input random variable, we have discussed the bias of RDS estimate and concluded that it is unbiased. We gave a mathematical argument together with a proof that the variance of RDS estimate is less than that obtained with random sampling with respect to a class of estimator.

References

[1] G. S. Fishman, Monte-Carlo: Concepts, algorithms and applications (Springer-Verlag, 1997).

[2] M. Ourbih-Tari, A. Aloui & A. Alioui. A software component which generates regular numbers from refined descriptive sampling. Proccedings of the European Simulation Modelling (ESM'2009) conference. Leicester, United Kingdom. Edited by Marwan Al-Akaidi. Softbound, pp 23-25. ISBN 978-90-77381-52-6. 2009.

[3] J. S. Ramberg, and B. W. Schmeiser, An Approximate Method for generating Symmetric Random Variables, Communications of the ACM. 15 (1972) 987- 990.

[4] E. Saliby, 1990. "Descriptive Sampling: A better approach to Monte Carlo simulation". Journal of the Operational Research Society, 41, 12, 1133-1142.

[5] Tari, M. and Dahmani, A. 2005. "Flowshop simulator using different sampling methods." Operational Research: An International Journal, 5, 2, 261-272.

[6] Tari, M. and Dahmani, A. 2005. "The three phase discrete event simulation using some sampling methods" International Journal of Applied Mathematics & Statistics, 3, D05 (Dec) 37-48.

[7] Tari, M. and Dahmani, A. 2006. "Refined descriptive sampling : A better approach to Monte Carlo simulation." Simulation Modelling Practice and Theory, 14, 143-160.

ON PHYSICAL PRINCIPLE OF COMPETITIVE NETWORKS

Dusan Fedorcak,
Ivo Vondrak
{dusan.fedorcak,ivo.vondrak}@vsb.cz
Department of Computer Science,
Faculty of Electrical Engineering and Computer Science,
VSB - Technical University of Ostrava, 17. listopadu 15/2172, 70833 Ostrava-Poruba, Czech Republic

September 30, 2010

KEYWORDS

Competitive Network, Unsupervised Learning, Force-Based Simulation, Physical Model, Particle System, Dynamic Relaxation

ABSTRACT

Competitive learning is a well-known method to process data. Various goals may be achieved using competitive learning such as classification or vector quantization. In this paper, we present a different insight into the principle of competitive learning. We want to highlight the fact that some learning rules used within competitive learning process is very similar to equations of motion. If motion equations are solved by numerical integration the similarity becomes very clear. In this article, we present a physically based model of the competitive network that emulates the standard competitive network through the dynamic relaxation of particle system. The model we built is equally powerful as the standard model and there are some positive features resulting from physically based approach.

INTRODUCTION

Artificial neural networks are phenomena of modern computer science. Many research groups are interested in these systems and artificial neural networks are strong computational devices by any means. Many types of artificial neural networks has been invented (or reinvented) so far, and most of them are inspired by some biological template. The research is also focused on various learning processes and similarities between artificial learning techniques and biological behaviors.

Our recent research was focused on utilizing the spring/particle system for visualization of some competitive networks (Fedorcak 2007). In this article, we will focus on altering the competitive network and turning it into the physically based particle system driven by motion equations. We will describe pseudo-physical model of simple competitive network where the learning rules are applied through the particles, forces etc. The comparison between the physically based model and standard approach is presented and some experimental results are also reported.

COMPETITIVE LEARNING

Competitive learning, as one of the unsupervised learning representatives, is a process of learning the organization of the input dataset through a competition between neurons. Neurons are often defined as reference vectors with the same dimensionality as the input data. The competition is usually driven by some metric, for example the Euclidean metric. The learning process consists of a fixed number of repetitive steps. Every step contains several substeps:

1. Determination of the winner according to the metric
2. Adaptation of the winner using some pseudo-hebbian adaptation rule
3. Optionally, if some sort of topology is defined, the neighborhood of the winner may be adapted as well.

Most of the competitive learning methods use very a similar learning step (Fritzke 1997). Sometimes, the network is able to change its structure dynamically by adding or removing neurons (*growing networks*) (Fritzke 1995); some other attributes such as a local error may be used for learning purposes. Our goal is to define a pseudo-physical model with the similar behavior as the competitive network has. The usage of the Euclidean metric and hebbian adaptation my be found useful and we will start with it. The determination of the winner is usually defined as:

$$\mathbf{w}_{win} = \arg\min_{i \in A} \|\xi - \mathbf{w}_i\| \tag{1}$$

where A is the set of neurons within the network, $\mathbf{w}_{win}$ is the reference vector of the winning neuron and ξ is the input signal passed to the network. The Hebbian adaptation is defined as:

$$\Delta\mathbf{w}_{win} = \eta(\xi - \mathbf{w}_{win}) \tag{2}$$

where η is the learning factor which will affect the speed of adaptation.

PHYSICAL REPRESENTATION OF COMPETITIVE LEARNING

The standard learning process is based on adaptation of neuron's weight vector according to the input signal passed to the network. An input signal is passed to the network in every time step, the winner is found and the adaptation (i.e. change of weight vector) is evaluated. In physically based approach we want to achieve the same effect through the forces affecting the particle system. This approach is widely used in many different applications and it is called *dynamic relaxation* (Brandes 2001)(Lewis 2003). The principle of dynamic relaxation is also used in machine learning theory, the famous Hopfied network is based on such principle (Hopfield 1982).

Our model is simple particle/force system. We assume that the neuron weight vector is represented by the position of the shapeless particle. We also assume that every input signal from the training set may be a source of distance dependent force. The sum of forces from every input signal to every particle (neuron) may have similar effect as the standard adaptation rule (2). The physically based learning process stands for a simulation of particle system through the time. In every time step the sum of all forces affecting the particle is evaluated and the change of its position is the adaptation of the neuron.

There is a significant difference between the standard learning process and the physically based simulation we describe above. The standard learning process is incremental by the mean that there is only one input signal processed in every time step. On the other side, the physically based learning evaluates the influence of all input signals within single time step. Nevertheless, there is a similar approach used widely in neural computing called batch-learning where the adaptation is evaluated after all input signals are precessed. Our physically based learning process is the imitation of such batch-learning method.

Winning Neuron Problem

The common principle of competitive networks is the competition of neurons. The competition is usually based on some kind of metric, typically the Euclidean metric. Unfortunately, there is no simple physical model which has exactly the same behavior as competition of neurons and individual adaptation of the winner. Nevertheless, there is a way how to imitate such behavior by introducing additional forces to the system. The first is an attractive force between the input signal and the particle (neuron). This attractive force has to be distance dependent and it should affect the winning neuron (the closest one) only, thus it needs to be inverse to the distance between the neuron and the input signal.

The attractive force is not sufficient to imitate winning neuron behavior of the standard model. The standard model is so-called order dependent where the winning neuron takes whole adaptation or the most of it, the second neuron take lesser portion of adaptation etc. The attractive force will affect close particles only but two particles within the same distance will be attracted with the same force. We need to penalize the second nearest neuron somehow. The inter-particle repulsive force is the influence we need. Such force will cancel the attractive force (the adaptation) if there is another particle (neuron) in a direction of the input signal.

There are some other differences between the standard model and the physically based model (such as the effect of particle velocity). The explanation of these differences along with the more detailed view of the physically based adaptation is given below.

Physical Representation of Simple Competitive Network

The model we built is based on basic equations of motion. Every neuron within the network is shapeless particle affected by forces generated by input signals or other particles. There is no shape associated with the particle therefore the motion is linear only and no rotational part of motion is needed.

$$\mathbf{F} = m\mathbf{a}, \quad \frac{d\mathbf{v}}{dt} = \mathbf{a}, \quad \frac{d\mathbf{x}}{dt} = \mathbf{v} \tag{3}$$

Every particle in the system is affected by all input signals. Every input signal ξ_i stands for a source of distance dependent force.

$$\mathbf{d}_i = \xi_i - \mathbf{x}, \quad \mathbf{F}_a = \sum_i \frac{g}{\|\mathbf{d}_i\|^k} \hat{\mathbf{d}}_i \tag{4}$$

The introduction of such force brings us a minor problem of creating a singularity in every input signal position. There is a simple solution of this problem; introduction of additional short-range repulsive force between the input signal and the particle.

$$\mathbf{F}_a = \sum_i \left[\frac{g_1}{\|\mathbf{d}_i\|^{k_1}} - \frac{g_2}{\|\mathbf{d}_i\|^{k_2}} \right] \hat{\mathbf{d}}_i \tag{5}$$

Without such modification it is likely to happened that the neuron approaching towards the input signal position will be ejected out of the cluster with very high velocity. The figure below (fig.1) shows the progression of $\mathbf{F}_a$ according to the particle/neuron distance where

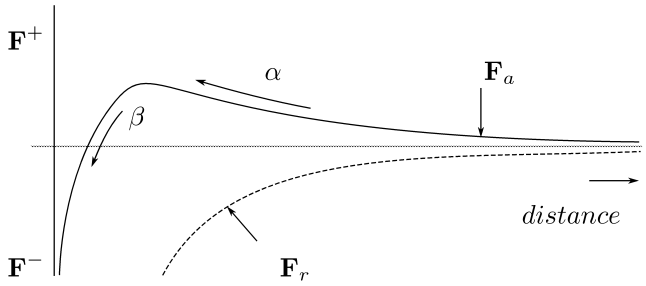


Figure 1: The progression of attractive force $\mathbf{F}_a$ in dependence on distance. Particles are attracted to the input signal (α) but within the short distance the repulsive force (β) will prevail. The inter-particle repulsive force $\mathbf{F}_r$ is also plotted (see below).

$g_1 > g_2$ and $k_1 < k_2$. As we mentioned before, the inter-particle repulsive force has to be added to the system. The definition of this force is similar to the attractive force (4) but it is generated between particles. The inter-particle repulsive force affecting particle j is described as:

$$\mathbf{p}_i = \mathbf{x}_i - \mathbf{x}_j, \quad \mathbf{F}_r = -\sum_{i \neq j} \frac{r}{\|\mathbf{p}_i\|^2} \hat{\mathbf{p}}_i \tag{6}$$

Simulation and Learning Process

The physically based learning is a simulation of particle system through the time. At the beginning, the initial positions of all particles in the system is random within the meaningful interval e.g. within the range of training set. The standard learning process advances in discrete time steps of fixed length and the same approach is used in the physically based approach. The simple Euler method (Ascher 1998) was found satisfactory for our purposes.

$$\Delta\mathbf{v} = \mathbf{a}(t)\Delta t, \quad \Delta\mathbf{x} = \mathbf{v}(t)\Delta t$$

$$\mathbf{x}(t+1) = \mathbf{x}(t) + \Delta\mathbf{x}, \quad \mathbf{v}(t+1) = \mathbf{v}(t) + \Delta\mathbf{v} \tag{7}$$

As the simulation advances the particle system tends to converge to the configuration with minimal kinetic energy e.g. to the configuration where the forces affecting the particles are balanced. We believed that such configuration will be also the configuration where the error of the competitive network is minimal or small enough.

Effect of Velocity

The velocity of a particle may affect the adaptation process significantly. The standard adaptation rule changes the position of neuron directly. The physically based learning does the same thing through force that changes the velocity of the particle and through the velocity the position is changed. Almost the same approach is used widely in neural computing (backprop. etc.) and it is called the momentum (Hecht 1989). With the omission of the mass the velocity practically become momentum and the effect to learning is identical.

The comparison of the standard adaptation and the physically based adaptation is given below. If we have an example configuration of only one neuron which is winning the competition every time and there is only one input signal, the adaptation is defined by recursive equation:

$$\begin{aligned} \mathbf{w}(t+1) &= \mathbf{w}(t) + \eta(\xi - \mathbf{w}(t)) \\ \mathbf{w}(t) &= \mathbf{w}(0)(1-\eta)^t + \sum_{i=0}^{t-1} \eta\xi(1-\eta)^i \\ \mathbf{w}(t) &= (\mathbf{w}(0) - \xi)(1-\eta)^t + \xi \end{aligned} \tag{8}$$

The last equation shows that if we have learning coefficient $0 < \eta < 1$ then in infinite time $t \to \infty$ the neuron weight $\mathbf{w}$ becomes ξ.

If we analyze the same situation in physically based approach, we will get simple harmonic oscillator without dumping:

$$\mathbf{x}(t) = (\mathbf{x}(0) - \xi)cos(2\pi f t) + \xi \tag{9}$$

Due to the conservation of energy, the particle will be oscillating the input signal infinitely. For this reason and also for numeric stability the damping force was introduced to the system. The damping force is velocity dependent and the whole force affecting the particle is described as follows:

$$\mathbf{F} = \mathbf{F}_a + \mathbf{F}_r + \mathbf{F}_d, \quad \mathbf{F}_d = -c\mathbf{v} \tag{10}$$

If we compare the damping force and the momentum factor from the standard model, it is right to say that without any dumping force ($c = 0$) the momentum factor is equal to 1 and whole weight change is transfered to the next learning step. The introducing the dumping force ($c > 0$) is equal to setting momentum factor lower then 1.

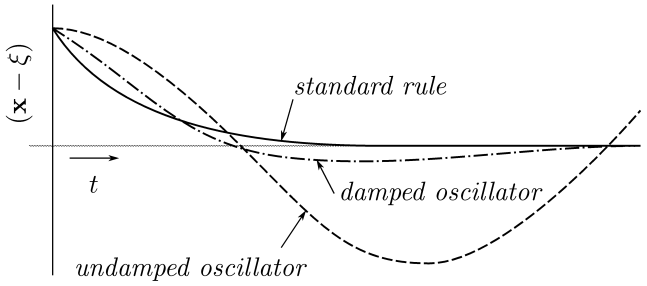


Figure 2: Distance progess through time with various learning rules

EXPERIMENTAL RESULTS

The physically based system was implemented according to the equations above and some experiments were done. The example training set of five hundred 2-dimensional patterns (Fritzke 1997) was used for learning. Simple competitive network based on winner-take-all rule was run on this dataset and the comparison between this standard model and the physically based approach is shown below.For presentation purposes, the trajectory of every neuron (or particle) was recorded through the learning process. These trajectories are displayed as red lines and may help to understand the different behavior of standard model (fig.3) and physically based approach (fig.4).

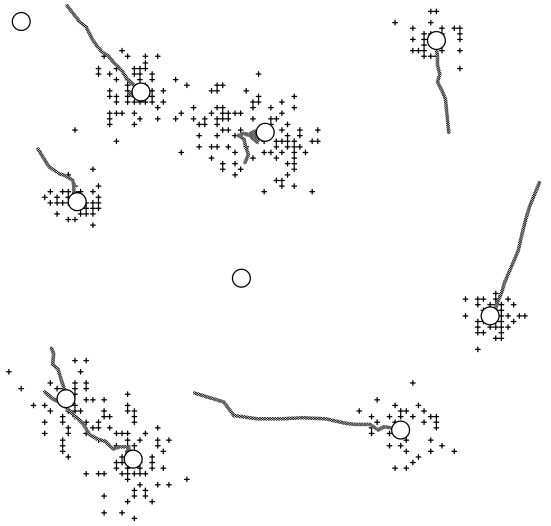


Figure 3: Experimental Results 1 – Standard simple competitive network

The physically based model we implemented found solution successfully and the final configuration is similar to one produced by the standard model. The trajectories shows some oscillations around ideal neuron position caused by momentum/velocity usage. Also, trajectories produced by physically based model are longer i.e. the system need more time (more learning steps) to converge to the solution. However, more experiments show that the physically based model produces better solution.

It is known that there is no guarantee of identical solution when the competitive network is run several times on identical dataset due to the random initial positions of neurons. Sometimes, never winning neurons (dead units) are present or there is a neuron oscillating between two clusters of input signals infinitely. The physically based model is better avoiding such configurations. For example, there are dead units on both of presented solutions but the physical model used nine of ten neurons (the standard model used only eight) and such solution is very repetitive. Almost 95% of experiments ended with the same solution (including the position of the dead unit). Figures below show some bad solution from standard competitive network (fig.5) and different solution from physically based model (fig.6).

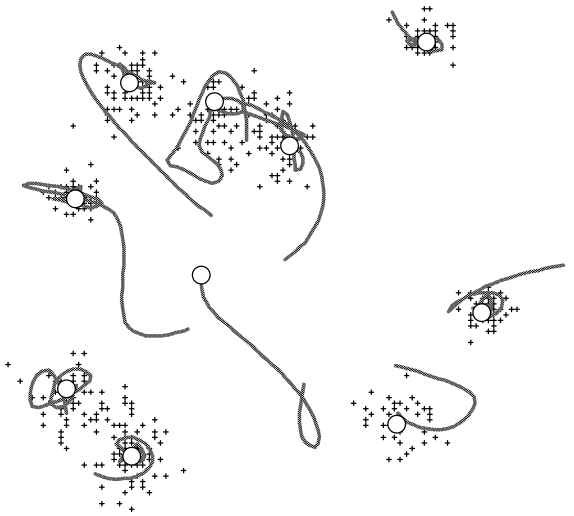


Figure 4: Experimental Results 2 – Physically based model

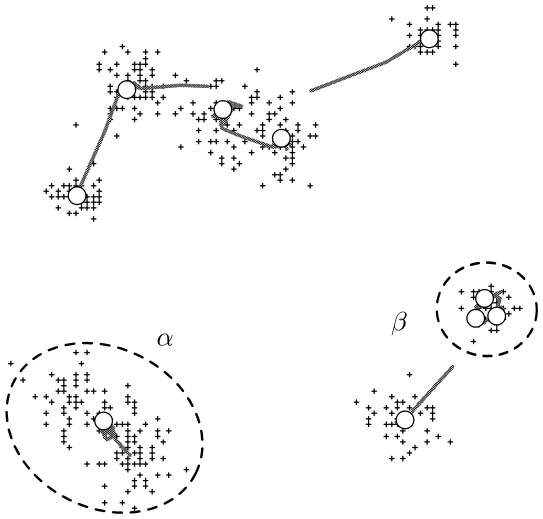


Figure 5: Experimental Results 3 – Standard competitive network; oscillating neuron (α) and unbalanced cluster (β) may be observed.

CONCLUSION

The pseudo-physical self-organized network presented in this article is different approach to self-organization than standard competitive networks. We presented that

Figure 6: Experimental Results 4 – Physically based model; identical solution (fig. 4) was reached from different initial configuration.

it is possible to emulate the behavior of competitive network through simulation of physically based particles within heterogeneous force field generated by input signals and particles themselves. The model we presented has some positive attributes and it is better in handling some problematic situations that the standard competitive model is not able to solve. Also, an important attribute is the fact that in every learning step (simulation step) all particles (neurons) are adapted and the adaptation is not driven by input signals only but also by positions of other neurons. Such approach leads to *more balanced and similar solutions* and elimination of dead units. We are aware that there are more sophisticated competitive networks using various approaches to avoid dead units and other unwanted situations. The aim of this article is to highlight the fact that it is possible to understand the self-organization from different angle and build a successful model on this idea.

Future Work

There is a possibility to transform various self-organized network to physically based model. For example, the physically based model of self-organized map (aka Kohonen map) is very promising and we are already experimenting with this model. It is possible to model self-organizing map as a grid of particles connected with springs emulating the adaptation of neighbors. Also, standard rules of self-organizing map may be enriched by some interesting features such as maximum spring force which will lead to tearing the map apart. In such way, sparse datasets may be mapped effectively and with better accuracy.

ACKNOWLEDGEMENT

This research has been supported by the internal grant agency of VSB-TU of Ostrava - SP/2010214 Modeling, simulation and verification of software processes II

REFERENCES

Fedorcak, D., Vondrak, I, *Force-based Visualization for Competitive Learning Methods.* in conference proceeding ESM 2007. Ghent: EUROSIS. 2007. 331-335. EUROSIS. 978-90-77381-36-6

Brandes Ulrik, *Drawing on Physical Analogies* M. Kaufmann and D. Wagner (Eds.): Drawing Graphs: Methods and Models, LNCS 2025, pp. 71-86, 2001. Springer-Verlag Berlin Heidelberg 2001

Fritzke B., *Some Competitive Learning Methods.* `http://www.neuroinformatik.ruhr-unibochum.de/ini/VDM/research/gsn/JavaPaper/`

Martinetz T. M., Schulten K. J. *A Neural Gas Network Learns Topologies* in T. Kohonen, K. Mkisara, O. Simula, and J.Kangas, editors, Artificial Neural Networks, pages 397-402. North-Holland, Amsterdam, 1991.

Fritzke B., *A Growing Neural Gas Network Learns Topologies* in Advances in Neural Information Processing Systems 7, pages 625-632. MIT Press, Cambridge MA, 1995.

Lewis W. J., *Tension Structures: Form and behaviour*, London, Telford, 2003 pages 60-80.

J. J. Hopfield. *Neural networks and physical systems with emergent collective computational abilities*, in proceedings of the National Academy of Sciences of the USA, vol. 79 no. 8 pp. 2554-2558, April 1982.

Ascher, Uri M.; Petzold, Linda Ruth. *Computer methods for ordinary differential equations and differential-algebraic equations.* 1998. SIAM. ISBN 0898714125

J. A. Kangas, T. Kohonen, and T. Laaksonen. *Variants of self-organizing maps* in EEE Transactions on Neural Networks, 1-99, 1990.

Ahalt S. C., Krishnamurthy A. K., Chen P., Melton D. *Competititve learning algorithms for vector quantization.* Neural Networks, 1990

Sima, J., Neruda R. *Teoreticke otazky neuronovych siti* MATFYZPRESS, Prague, 1996, (in czech)

Hecht, Nielsen, *Neurocomputing*, Addison-Wesley, 1989

A WORKFLOW HYBRID AS A MULTI-MODEL, MULTI-PARADIGM SIMULATION FRAMEWORK

Stuart Rossiter and Keith R.W. Bell
Department of Electronic & Electrical Engineering
University of Strathclyde
204 George Street, Glasgow, G1 1XW, UK
email: srossiter@eee.strath.ac.uk

KEYWORDS
hybrid simulation framework, social simulation, comparative, multi-paradigm, workflow

ABSTRACT

We propose that workflow software can be coupled with existing simulation frameworks (particularly agent-based ones) to provide three broad benefits: an improved modelling process due to the separation of concerns and rich scheduling syntax; interchangeable human and AI agents at minimal development cost; a common conceptual and software base for multi-model, comparative studies of the same system (including shared, distributed data visualisation).
We explain these benefits, providing a proof-of-concept framework implementation and examples from the domain of electricity generation expansion planning.

MOTIVATION

The complexity of social systems (e.g., markets) has led to a multitude of different paradigms and methodological schools for simulating them (Gilbert and Troitzsch 2005). Various software frameworks have sprung up to help structure this process, often with attempts to provide a more visual programming environment (e.g., the use of flowchart-driven behavioural processing in Repast Simphony (North et al. 2007), alongside traditional Java code). However, these typically assume a modelling paradigm (e.g., agent-based or system dynamics), and focus on structuring the model, supporting tools, performance and usability; the overall modelling *process* is left separate. (There are exceptions, such as Mimosa (Müller 2007), which is a suite of tools for a coherent modelling process formalism.)
This all means that studies of social systems are often multi-disciplinary, multi-model approaches where differences in concepts and modelling techniques mean it is very difficult for individuals to understand, compare and coherently extend existing models. (Axelrod (1997) provides a more focused summary on the scope and maturity of social simulation as a discipline.) As an example, consider the domain of **generation expansion models**. These attempt to model how investment decisions are made for electricity generation plant, and how they may fare in the resultant market; the aim being to consider what future configurations of plant might arise given certain exogeneous scenarios and endogeneous system mechanisms. Modern generation expansion models can vary from traditional least-cost optimisation (New Zealand Electricity Commission 2009), to aggregate systems-dynamics-based studies (Ford 1999), to detailed agent-based models with individualised decision-making (Botterud et al. 2007).
This domain is interesting because, despite these differences, the particular nature of electricity as a product—and its associated technical infrastructure—means that participant behaviour actually has some well-defined constraints, in terms of the electricity market protocols and generation connection processes enforced by the transmission network operator. (These constraints can be contrasted with the idiosyncratic, per-organisation processes by which generating companies make their strategic and operational decisions.) We call these **constraining global processes** hereafter.

A WORKFLOW HYBRID FRAMEWORK

Workflow software is normally used to help define, automate and re-engineer business processes. The workflows themselves are defined using some specification language, which typically maps to some graph-based formalism. They run in an engine and, where human input is required for tasks, users interact with the workflow via some front-end (see figure 1). Automated 'users' may also be responsible for some workflow tasks, or for custom code which might do things like allocating a user to a task, or deriving and rendering specific data.
We propose that workflow software can be used in a framework for simulation models which primarily use another paradigm (e.g., agent-based); effectively, the workflows define selected scheduling and related data definitions. We argue that such a hybrid can help address some of the issues in the previous section, despite the fact that, on the surface, it seems very much a 'like-for-like' replacement of the scheduling mechanism already used. The benefits are particularly apparent for multi-model studies of systems which have constraining global processes, since decisions on what processes to enshrine in workflow form are much more apparent and 'uncontroversial'.
The hybrid framework includes a **workflow framework**, and a coupled **main model framework**. Such hybridisation fits

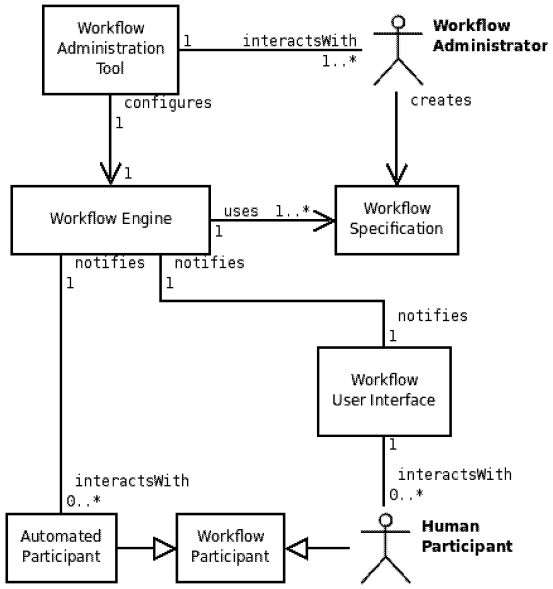


Figure 1: Generic workflow software architecture (as a UML class diagram, with actor icons used for classes representing human participants)

best conceptually for *agent-based* main model frameworks, since there is a natural fit between workflow users and agents in the agent-based model (ABM), and typically a clear separation of scheduling from agent behavioural logic (cf. system dynamics as a set of coupled difference equations). Workflows also tie their actions to roles, which aligns nicely with the use of object-oriented inheritance in agent development. However, we believe it is still feasible for other paradigms (see later). For reference, the basic ABM architecture is shown in figure 2, with the assumption that no further background explanation is needed.

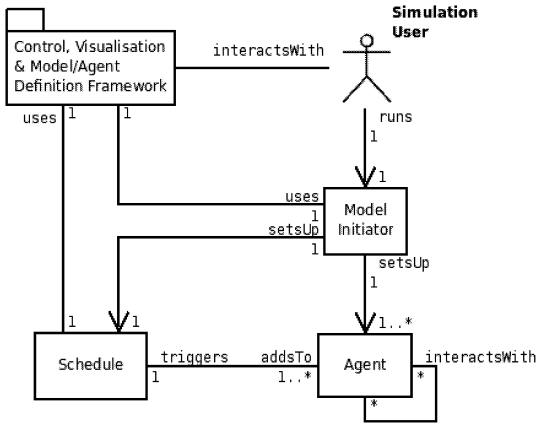


Figure 2: Generic agent-based modelling (ABM) architecture (same notation as figure 1)

We define three areas of potential benefit in the sections below.

Improved Modelling Process

Using workflows provides a natural, visual development style which enforces early, up-front consideration of the overall control flow (in terms of real-world processes), attendant data structures, and the set of agent roles involved. (Most workflow software uses a platform-neutral XML data representation, which also helps move the modeller away from programming language specific design.) This therefore enshrines some good model design practices, including the separation of this 'programming-in-the-large' from the 'programming-in-the-small' of the agent behaviour; a software improvement technique with a long history (DeRemer and Kron 1975). Although similar separations and visual environments may exist in ABM frameworks, their use is much more optional than in this approach. In addition, workflows provide a rich set of control primitives which may be better suited for clearer and more powerful scheduling.

In terms of model extension, reuse and collaboration, this separation also helps allow different agent implementations to be switched in and out, since there is no temptation to embed control flow logic in the agents themselves, and the data interfaces for each agent action are clearly and centrally defined. We have also found in practice that having to focus on shared data definitions for multiple models is very effective in clarifying the essential concepts and meanings that the data embeds.

Finally, agents often tend to require information on the actions of other agents to update their own internal models. The decoupling above can be enhanced by defining what we call **informational events** in the workflow. That is, 'real' workflow tasks (carried out by agents) can be followed by 'infrastructural' tasks which publish events providing data on the action that just occurred and its outcomes. Agents can subscribe to events of interest, thus decoupling such informational interactions from agent logic and further centralising agent interaction capabilities. (A modeller can then choose to implement their own information filtering logic to reflect things like interaction topologies, without having this fixed by interaction logic encoded elsewhere.) See the *Info Events Server* component in figure 3.

Interchanging Human and AI Agents

It is very difficult to validate models of human decision-making, primarily because the modeller cannot easily experiment with the real-world system (Windrum et al. 2007). This means that empirical simulation experiments can be very useful, where real humans may take the place of, or compete against, computational (AI) agents (Tesfatsion 2002; p.57). Human agents can also be used for computational steering, directing the simulation towards interesting areas of behaviour. If a modeller wants to be able to mix or interchange human and AI agents, most of the complication is in the data definition, rendering, thread control and user interface for the human agent. However, workflow software is designed precisely to handle these aspects, so we can get these benefits

at minimal development cost, without some custom-coded client-server simulation—such as PSERC's PowerWeb simulation of power exchange auction markets (Zimmerman et al. 1999).
There are some inherent complexities in merging workflow-based human actions (which normally run in real-time) with a simulation-time-based ABM framework (see later).

A Common Conceptual and Software Base for Multi-Model Research

If we want to use *shared* workflows to add some structure and coherency to multi-model approaches, the question becomes what should be 'workflow-ised' without overly restricting the modelling freedom required. Do shared workflow definitions even make sense where we may be modelling the same real-world system at different aggregation levels, and potentially focusing on different aspects?
We propose that it *does* make sense for for real-world systems which have constraining global processes, as discussed earlier. The use of workflows brings the following conceptual and development benefits:

1. It explicitly models baselines for the identified constraining processes, where the level at which the processes are defined establishes up-front what type of freedom the set of models is intended to have. It also specifies the aggregation level at which some form of comparison is likely to be required.

   Models which use these workflows are ensured as consistent at this conceptual level. Each may model further, disaggregated detail *outside* of the workflow definitions. (For example, the workflow might represent a daily market as a 'black box', with details of participants passed in, and per-participant revenues output; particular models would use their own sub-model to determine what those output revenues were.)

   In practice, we also found that the need to tie-in with the workflows makes it much clearer to the modeller when their particular model is moving away from the conceptual underpinnings, since they find themself having to code around 'restrictions' in the design. This forces a deeper consideration of where the conceptual discontinuity lies, which is very useful in effectively comparing and discussing models.

2. It provides a consistent terminology and set of causal assumptions on the system (including assumptions such as what order elements of these processes have to occur in).

3. By introducing the possibility for human agents, it directs the modeller to produce meta-data with maximal cross-model, long-term benefit. To see why this is true, consider that human agents typically prefer human-oriented graphical or statistical aggregate representations of data. In addition, most decisions will often refer back to a small number of 'global' data items which tend to reflect the shared environment within which the agents operate (such as, for generation expansion models, underlying plant costs or power flows on the electricity transmission network under certain load or outage scenarios).

   Therefore, part of our framework includes a standard way to define these global fields, their statistics or visualisations, and human/AI agent access to them. (Agents can still filter this data to model aspects such as perceptual range outside of the workflows.) Workflow software typically provides a generic user-interface (e.g., a visual representation of an XML data hierarchy), with software 'hooks' to allow for customised visualisation as required. Such customisations can reuse the visualisation capabilities of the coupled main model framework, though this requires some distribution of these capabilities so that visualisations can be recreated for human agents participating in the model via networked clients (see figure 3).

   Importantly, these data visualisations also tend to be useful views on the dynamics of the model *for the simulation researcher*, so the modeller is not significantly wasting coding effort if human agents do not end up being extensively used. Because this meta-data is based on well-thought-out underlying data (that has been agreed as a consistent base for a set of possibly very different models), major future changes should not be needed, and the code provides clean separation of shareable meta-data from agent decision logic.

4. It ensures careful consideration of the effects of differing aggregation levels, which is often the main distinction between modelling paradigms (e.g., Bonabeau (2002) compares agent-based and system dynamics representations of the same system).

   Workflows are typically defined at the lowest aggregation level (e.g., individual traders in a market), with the coupled main model having to aggregate and disaggregate as it requires. This makes it much easier to do comparative models that compare the effects of individual variation at differing definitions of 'individual'.

PROOF-OF-CONCEPT IMPLEMENTATION

HAWSER (Hybrid Agent-Workflow Simulation Engine for Research) has been developed as a proof-of-concept implementation of the framework. (Some specific aspects, such as distributed visualisation, are not yet implemented.) HAWSER couples two existing, open source frameworks: MASON (Luke et al. 2005) for agent-based modelling, and YAWL (van der Aalst et al. 2004) for workflows. All code, including both open source frameworks, is written in Java.
The high-level architecture is shown in figure 3, which previous discussions refer to. Due to space restrictions, we do not describe this further, restricting ourselves to an overview of the key technical challenges. We intend to publish the

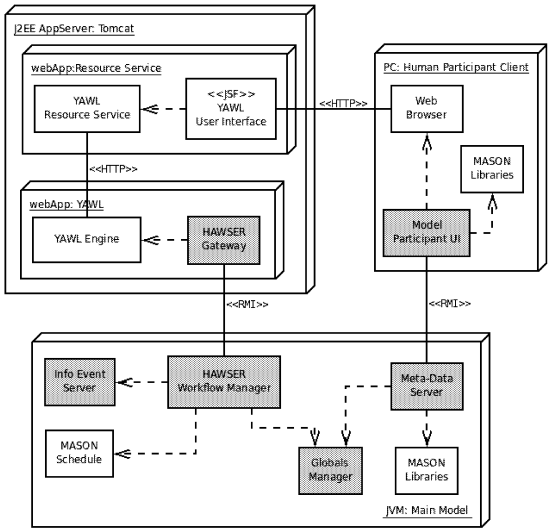


Figure 3: High-level deployed architecture of the HAWSER framework (as a UML deployment diagram). Shaded components are those added to the existing ABM and workflow frameworks

framework, together with detailed documentation and demo models, as an open source project in the near future.

Technical Challenges

1. The main ABM runs in a separate process (Java virtual machine) to the YAWL system, and thus inter-process communication is required. We use Java RMI for simplicity. This also allows the workflow software to run on a separate physical machine if required.

   YAWL provides the ability to define a third-party ('observer') gateway which receives notification of all new workflow tasks and has direct access to the main YAWL engine. We use this feature to implement a HAWSER Gateway component, which communicates with a Workflow Manager component within the main model's process. The latter provides the bridge for workflow actions to be passed on to the relevant agents, and is used to initiate workflows.

2. The MASON discrete-event schedule defines the global simulation clock. Workflow tasks, which now represent agent actions, should therefore take some elapsed simulated time to complete; that is, the action should not be completed until a simulation timestep which represents the simulated completion time.

   Agents declare the simulated time taken, and special Deferred Action Handler objects are added to the schedule to trigger task completion at the appropriate timestep.

3. Workflows with parallel tasks result in multiple, parallel threads which need to be controlled so that they are fed *sequentially*, and in a *repeatable order* to agents. (It is a fundamental simulation requirement that models are repeatable, in that the same model with the same input data, including a random seed for stochastic elements, results in exactly the same output.) This is handled via meta-knowledge of what tasks are going to be, or have been, triggered by workflow actions. All new threads are held by the Gateway, and the meta-data used to release tasks in a deterministic order.

   In addition, the main ABM model has to run with only a single thread active at any one time. This requires some elaborate control logic in the Gateway and Workflow Manager to determine when main model threads should wait or continue (especially since workflows may be run from within workflows).

4. To avoid excessive data traffic to and from workflows, it often makes sense to visualise global data externally to the workflow software, but presented in a common unified interface to human agent participants. The framework is designed to support this, which also allows AI agents to be presented directly with the more powerful object-based representations (cf. some 'flattened' XML equivalent in workflow data).

5. Agents may wish to delay their processing (within some time window) to see what others agents do. (What they are actually aware of is dependent on the specifics of the model.) This complicates matters for human agents, who need to be able to declare that they are waiting for some simulated time. Enabling such time limits and human re-processing loops requires dynamic alteration of special template workflow definitions. YAWL does not currently support this, but work to add such a capability is planned.

Generation expansion models have been realised using the framework and the related multi-model methodology. The global data consists of a representation of the transmission network, together with visualisations useful to the researcher and human agents.

REFLECTIONS & FUTURE WORK

We reflect on two general aspects of this work.

Theory & Novelty

We are promoting the benefits of a workflow formalism for agent scheduling, and we believe that there are significant methodological benefits for the right types of real-world system. In the workflow literature, and the related field of queueing models in operational research, workflow *has* been coupled with simulation (and agents), but the interest has been in the workflows *themselves*, and the business processes they represent—such as in: the simulation *of* workflows to test proposed business process changes (Rozinat et al. 2008); ABM alternatives to queueing models to represent dynamic

business processes (Tan et al. 2007); and workflows representing inter-agent interaction frameworks (Zhuge 2003). This is rooted in the history of workflow software as a tool for business process optimisation and re-engineering; we are 'repurposing' workflows for 'normal' simulation modelling. From a theoretical point of view, we should bear in mind that *all* simulations with discretised time can be represented by some form of discrete-event based model (Zeigler et al. 2000), and that this DEVS formalism can be extended to explicitly represent agent-based models (Müller 2009). The HAWSER implementation aligns with the common ABM practice of using randomised iteration to represent simultaneous events. Müller (2009) points out that this is a potentially undesirable formalism, and looks at various DEVS extensions to provide a better alternative. It may therefore be of interest to further consider workflow-ABM coupling in this more theoretical light, particularly as workflow formalisms are specifically designed to model concurrent processes.

Extension to Non-ABM Models

We stated earlier that, in theory, the approach could be used for other simulation paradigms, notably system dynamics (the DEVS background discussed above supports this). However, the details need to be considered and proven: workflow data has to link to and from the stocks and flows of the system dynamics paradigm. Ninios et al. (1995) discuss some of the 'paradigm clash' difficulties in making such a switch.

REFERENCES

van der Aalst W.; Aldred L.; Dumas M.; and ter Hofstede A., 2004. *Design and implementation of the YAWL system*. In *Proceedings of the 16th International Conference on Advanced Information Systems Engineering (CAISE 2004)*. 281–305.

Axelrod R., 1997. *Advancing the Art of Simulation in the Social Sciences*. In R. Conte; R. Hegselmann; and P. Terna (Eds.), *Simulating Social Phenomena*, Springer, *Lecture Notes in Economics and Mathematical Systems*, vol. 456. 21–40.

Bonabeau E., 2002. *Agent-based modeling: Methods and techniques for simulating human systems*. *Proceedings of the National Academy of Sciences*, 99, 7280–7287.

Botterud A.; Mahalik M.; Veselka T.; Ryu H.; and Sohn K., 2007. *Multi-Agent Simulation of Generation Expansion in Electricity Markets*. In *IEEE Power Engineering Society 2007 General Meeting*. IEEE, 1–8.

DeRemer F. and Kron H., 1975. *Programming-in-the large versus programming-in-the-small*. In *Proceedings of the international conference on Reliable software*. 114–121.

Ford A., 1999. *Cycles in competitive electricity markets: a simulation study of the western United States*. *Energy Policy*, 27, 637–658.

Gilbert N. and Troitzsch K., 2005. *Simulation for the Social Scientist*. Open University Press, 2nd ed.

Luke S.; Cioffi-Revilla C.; Panait L.; Sullivan K.; and Balan G., 2005. *MASON: A Multiagent Simulation Environment*. *Simulation*, 81, no. 7, 517–527.

Müller J., 2007. *Mimosa: using ontologies for modeling and simulation*. In *Proceedings of the 8th Asia-Pacific complex systems conference (Complex'07)*.

Müller J., 2009. *Towards a Formal Semantics of Event-Based Multi-agent Simulations*. In *Multi-agent Based Simulation IX*. Springer, no. 5269 in LNCS, 110–126.

New Zealand Electricity Commission, 2009. *2009 Grid Planning Assumptions*. Tech. rep., New Zealand Electricity Commission.

Ninios P.; Vlahos K.; and Bunn D., 1995. *Industry Simulation: System modelling with an object oriented / DEVS technology*. *European Journal of Operational Research*, 81, 521–534.

North M.; Tatara E.; Collier N.; and Ozik J., 2007. *Visual agent-based model development with Repast Simphony*. In *Proceedings of the Agent 2007 Conference on Complex Interaction and Social Emergence*.

Rozinat A.; Wynn M.; van der Aalst W.; ter Hofstede A.; and Fidge C., 2008. *Workflow Simulation for Operational Decision Support Using Design, Historic and State Information*. In *Proceedings of the 6th International Conference on Business Process Management (BPM 2008)*. Springer-Verlag, 196–211.

Tan W.; Li S.; Tang A.; and Shen W., 2007. *A Workflow Simulation Framework Based on Multi-agent Cooperation*. In *Proceedings of the 2007 11th International Conference on Computer Supported Cooperative Work in Design*.

Tesfatsion L., 2002. *Agent-based computational economics: growing economies from the bottom up*. *Artificial Life*, 8, 55–82.

Windrum P.; Fagiolo G.; and Moneta A., 2007. *Empirical Validation of Agent-Based Models: Alternatives and Prospects*. *Journal of Artificial Societies & Social Simulation*, 10, no. 2, 8.

Zeigler B.; Gon Kim T.; and Praehofer H., 2000. *Theory of modeling and simulation : integrating discrete event and continuous complex dynamic systems*. Academic Press, 2nd ed.

Zhuge H., 2003. *Workflow- and agent-based cognitive flow management for distributed team cooperation*. *Information & Management*, 40, no. 5, 419–429.

Zimmerman R.; Thomas R.; Gan D.; and Murillo-Sánchez C., 1999. *A Web-based platform for experimental investigation of electric power auctions*. *Decision Support Systems*, 24, 193–205.

SIMULATION MODELS

A SECURITY SIMULATION MODEL FOR LARGE SCALE DISTRIBUTED SYSTEMS

Ciprian Dobre, Florina Constantin, Florin Pop and Valentin Cristea
Department of Computer Science
University POLITEHNICA of Bucharest
Spl. Independentei, 313, Bucharest
Romania
E-mails: florina.constantin@cti.pub.ro, {ciprian.dobre, florin.pop, valentin.cristea}@cs.pub.ro

KEYWORDS
Security, modeling and simulation, evaluation, large scale distributed systems.

ABSTRACT

Today there is a growing interest for large scale distributed systems, both from academia and the industrial environment. If until recently the research in this area mainly focused on developing operational infrastructures, currently many applications have some additional needs. Among these, security represents a hot research topic. In this paper we present a simulation model suitable for evaluating methods and techniques designed to increase security in large distributed systems. The model has the characteristics needed to develop a wide range of security scenarios, being able to assess from solutions to secure data transfers to various mechanisms to assess the access management in a distributed system. The model was implemented as an extension of the MONARC simulator for distributed systems. We present experimental results demonstrating its capabilities to correctly model security solutions for large scale distributed systems, and to pinpoint likely security problems in the simulated environments.

1. INTRODUCTION

Modeling and simulation were seen for a long time as viable solutions to develop new algorithms and technologies and to enable the enhancement of large-scale distributed systems, where analytical validations are prohibited by the scale of the encountered problems. The use of discrete-event simulators in the design and development of large scale distributed systems is appealing due to their efficiency and scalability.
Together with the extension of the application domains, new requirements have emerged for large scale distributed systems; among these requirements, security is needed by more and more modern distributed applications, not only by the critical ones. Most times the resources of such systems are located in different geographically dispersed administrative domains. The evaluation of such solutions is usually done by implementing them in real-world environments. Such an approach, however, implies costs. Also, it is hard to make general remarks on the validity of a particular solution based on the observations made in a particular study case.
In this paper we present a security model that allows the analysis of security-dependent experiments, where possible problems can occur in any simulated component. The use of modeling and simulation is appealing because it allows a greater flexibility in evaluating security solutions for distributed systems.
The model was implemented as an extension of the MONARC simulator. This extension allows the user to correctly describe security solutions currently used in many real-world distributed environments (such as GSI, PKI, SSL, cryptographic solutions, etc.). In addition, the implementation includes already-available simulated security attacks. It allows the addition of detection mechanisms for such attacks, by providing simulation mechanisms for message encryption or authentication and authorization. The modularity and extensibility also allows the user to easily add new capabilities or components for custom experiments.
The rest of the paper is structured as follows. Section 2 gives a description of the work related to the work presented in this paper. In Section 3 we present the security model. Section 4 presents implementation details of the extension added to the MONARC simulator. In Section 5 we present experimental results that demonstrate the capabilities of the security model. Finally, in Section 6 we give conclusions and present future work.

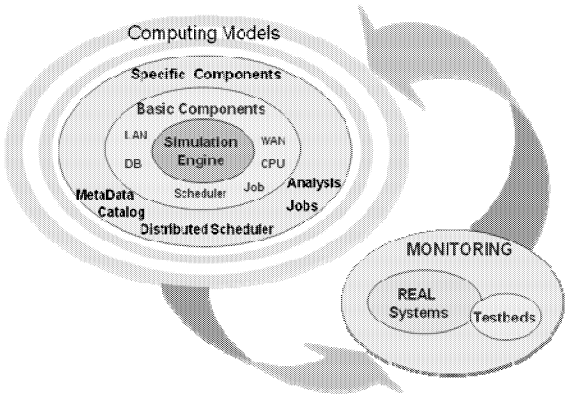
2. RELATED WORK

Currently there are various simulators designed to evaluate solutions for distributed systems (SimGrid, Grids, OptorSim, etc.). In their vast majority, they were all implemented for the modeling of particular problems (such as scheduling, data management, etc.). Because of their narrow areas of usage, many of them do not support the modeling of security functionalities.
The only existing simulator that offers this capability is *G3S* (*Grid Security Services Simulator*), a simulator that can be integrated with GridSim to combine technical investigations to find more efficient allocation of resources with the testing of the security services (Naqvi and Riguidel, 2005). The simulator is currently no longer available and/or supported.
G3S was developed to support various authentication mechanisms, including X.509 certificates and Kerberos tickets. For authorization G3S uses the Roles Based Access Control (RBAC). It also supports the Bell-LaPadula model for ensuring privacy, and the Watermarking technique for ensuring the integrity of data transmitted between the grid's

resources (Naqvi and Riguidel, 2005). It also simulates the privacy feature.
G3S offers various types of patterns of attack, enabling designers to verify if their design may prevent security threats and survive them. In addition, G3S includes a mechanism for spreading security threats notifications. For example, if a node tryies to exceed / violate the privileges defined then an alert about the existence of a malicious node is transmitted to all major nodes (Naqvi and Riguidel, 2005).
In this we also present a simulation model that includes such capabilities for modeling security features, from patterns of attack to intrusion detection, authentication or privacy enforcement solutions. The proposed model also considers a wide-range of security solutions in the general case of large scale distributed systems. The simulation model provided by MONARC is more generic that others, as demonstrated in (Dobre and Cristea, 2007). It is able to describe various actual distributed system technologies, and provides the mechanisms to describe concurrent network traffic, to evaluate different strategies in data replication, and to analyze job scheduling procedures. MONARC offers ample customization possibilities, thus enabling us to integrate our model while preserving the interface. Also, because of this feature, our model can incorporate custom security solutions designed by the user for particular scenarios.

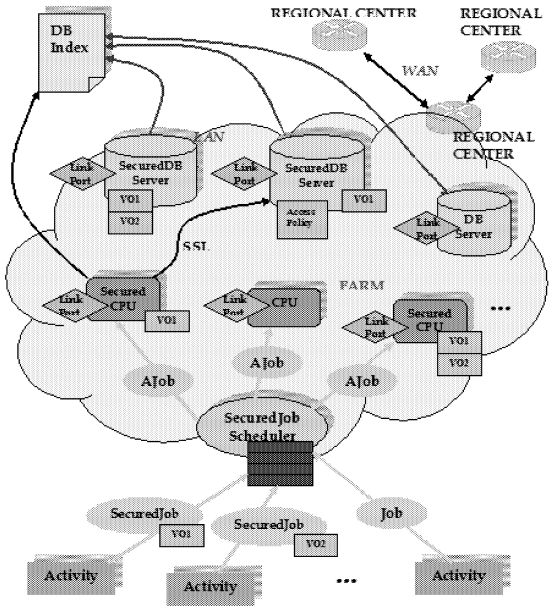
3. A SECURITY MODEL FOR LARGE SCALE DISTRIBUTED SYSTEMS

The proposed model considers the general case of security, as a mean to ensure that systems remain safe and reliable to the errors, threats or malicious changes. The model considers solutions for *data privacy, data integrity* and *system availability*. To ensure such objectives, we consider components designed to protect the services, data and offered information from threats such as *interruption, interception, change* or *forgery*.
The starting point in designing the security model consisted in the specification of security requirements, namely security policy. A security policy describes which actions are allowed and which are prohibited. Entities to which these actions apply include users, services, information, machinery, etc.
Once the security policy is established, the necessary security arrangements for its implementation may be considered. The most important security mechanisms considered are (Johnston, 2004) *confidentiality (*the model includes mechanisms designed to ensure that an authenticated entity can access only the information that has been authorized to*)*, *authentication (*the model includes mechanisms to identify entities involved in a communication or collaboration*)*, a*uthorization (*the model guarantees that once the entity has been authenticated, its options will be restricted / limited to those operations that it is authorized to perform*)*, and a*udit (*the models includes the mechanisms to guarantee the non-repudiation of origin and content of a message*)*.
For the particular case of Grid systems, an additional important concept, also considered by the security model, is the one of Virtual Organization (VO). In a VO different organizations (commercial companies, universities, governmental institutions or laboratories) collaborate to share resources and work together to solve common problems. Each company within a VO is managed independently and has its own security solutions such as Kerberos or PKI infrastructure (Public Key Infrastructure).



Figures 1: The architecture of the MONARC simulator (Dobre, *et al*, 2008)

The model extends the regional center model provided by the MONARC simulator (Dobre and Stratan, 2004). The simulator was chosen based on its capability to allow the modeling of a wide range of distributed systems architectures (Dobre and Cristea, 2007). MONARC provides a realistic simulation environment for modeling large distributed computing systems. The simulator contains the necessary mechanisms for the modeling of competing traffic, for the evaluation of various data replication strategies or the scheduling of task execution.



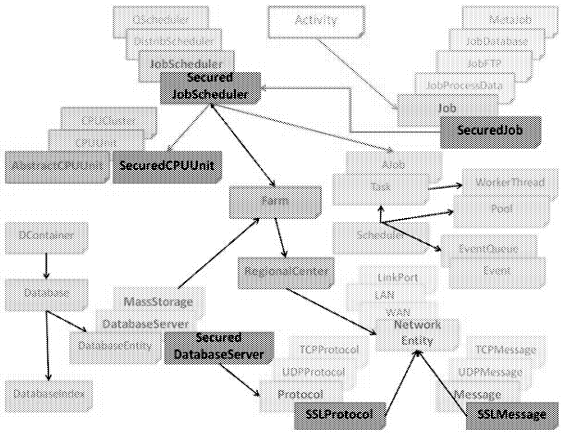
Figures 2: The security model

One of the advantages of MONARC is the ease of expansion and this approach is facilitated by a layered structure (Fig. 1). The first two layers contain the core of simulator (or

"simulation engine") and the basic components of any distributed system (processing units, tasks, databases, networks, the planning, etc.). On top of these are the components designed specifically for certain models of distributed systems. These individual components include different tasks, schedulers with specific algorithms or databases that support data replication. Such simulation components can be easily extended by the user according to requirements of various simulation experiments. Users can also extend the architecture by introducing new components.
The MONARC's model includes several components (Fig. 2). A first set of components is used for describing the physical resources of the distributed system under simulation. The largest one is the regional center, which contains a farm of processing nodes (CPU units), database servers and mass storage units, as well as one or more local and wide area networks. Another set of components model the behaviour of the applications and their interaction with users. Such components are the "Users" or "Activity" objects which are used to generate data processing jobs based on different scenarios. The job is another basic component, simulated with the aid of an active object, and scheduled for execution on a CPU unit by a "Job Scheduler" object.
The security model extends the regional center model. It includes extensions to all existing components of a distributed system (processing units, database server, jobs, job schedulers). We added a new secured job that carry authentication tokens or certificates, and is able to request data based on specific rights. The user can specify the use of X.509 certificate, together with a PKI infrastructure for example, or can easily add new means of authentication.
The model also includes the possibility to define VOs, based on specific security policies shared between regional centers. The model includes the mechanisms to evaluate various authentication solutions. Such authentication mechanisms are applied to the scheduler, processing unit, and even when jobs request data from the database servers.
For example, the job scheduler includes restrictions to where to execute specific jobs, based on the VO to which they belong. The processing units are capable to verify if a particular job is allowed to be executed. The access control verification can be implemented based on various schemas (RBAC, MAC, DAC, etc).
Also, we added delegation, a concept important especially in case of Grid experiments. Thus, a job is capable to delegate his rights (in the form of a certificate for example) to all the jobs that he further instantiates. The database server supports various security policies and includes mechanisms to verify authentication of incoming connections, and to support secured transport protocols. It also includes various mechanisms to certify the validity of data access requests. Similar to the processing unit, the database server maintains a list of VOs to which it belongs and the access policies for the data shared by the server with the other resources within the VO. For example, when a job accesses data on the server the model specifies a series of mechanisms to verify the identity of the requester (by default the model uses validation based on a PKI infrastructure and X.509 certificates) and if the job presents sufficient access rights. For compatibility with other experiments, a regional center can include both secured and non-secured database servers. The same applies for the processing units.
Within the networking simulation model, the model adds the possibility to include secured data transport protocols. For example, we included the SSL protocol to offer the possibility to encrypt the messages being exchanged between entities in a simulation experiment. We added the possibility to implement various handshake mechanisms (for protocols supporting authentication capabilities). The user can easily add and evaluate new protocols and mechanisms. The model includes mechanisms for data encryption, keys and certificate management, etc. In addition, it includes mechanisms for traffic filtering by specifying exclusion rules based on various metrics (ports, addresses, protocols, etc) and corresponding actions (reject for example).

4. IMPLEMENTATION DETAILS

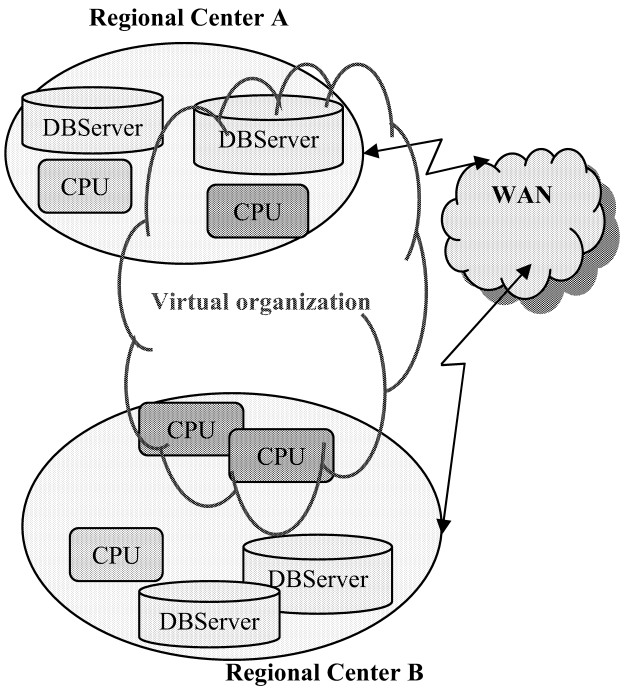
Figure 3 presents the main classes added to the MONARC simulator. The implementation allows the possibility of *defining virtual organizations* in which users can share resources from various regional centers, such as computing power or databases, in a manner as safe as possible. Also, it allows *defining security polices* for each of the components belonging to a virtual organization, policies through which the proprietary organization is able to impose restrictions on the access to its shared components. These restrictions may include restrictions on the execution of jobs on workstations or access restrictions to the data on certain database servers.
Based on the security module of Globus Toolkit (The Globus security Team, 2005) and the GSI (The Globus security Team, 2010), we added within the security model the possibility to specify *authentication* mechanisms using *X.509 certificates*. The X.509 certificate is used for authentication to various Grid entities. Users belonging to a VO present such a certificate that 'vows' for their identity when submitting a job for execution or when communicating with other system entities.



Figures 3: The classes added in MONARC

The security policies in VO shape the access *authorization* mechanism to the shared resources within them. For instance, if a shared workstation belonging to a VO has an associated security policy that imposes a limit on the amount of memory that a job submitted can use then any job that

violates this restriction is ignored. Authentication and authorization ensure the system from unauthorized access to its resources.
The security implementation also enables the *protection of message content* sent throughout the network against attacks such as interception (eavesdropping), and thus keeping its confidentiality, by encrypting its content. It also ensures *secure data transfers* by proposing the use of various protocols that allow the authentication of the parties involved in the communication (such as the SSL protocol). This ensures both the integrity of messages transmitted, and their protection against attacks such as man in the middle.
The implementation also includes an exclusion rule based *traffic filtering* of all components of a virtual organization. This mechanism can be used to prevent attacks such as DoS. In case of many connections coming from the same address, for example, the filtering policy can specify that that particular address is banned for a certain period of time (or permanently).



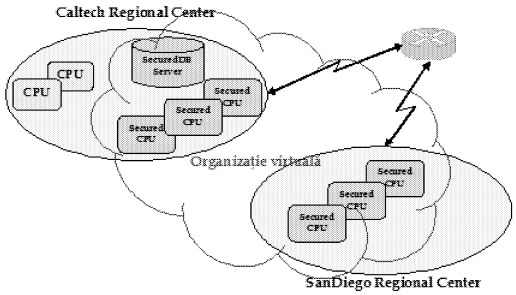
Figures 4: A Virtual Organization example

In the implementation we extended all basic components: the processing unit, the job, the database and the job scheduler by introducing an authentication mechanism, access control mechanisms, or the possibility to schedule the jobs according to the restrictions imposed by the virtual organization in which they are executed. For the jobs being executed the processing units include various mechanisms for specifying their execution rights. Also, the data can specify different protection rights and mechanisms.
A component that models the specific behavior of a SSL protocol was added in an effort to simulate the authentication of the components involved in the message transfer and to ensure the confidentiality of the message transmitted through the network by using cryptography.
In addition we added a component to simulate traffic filtering based not only on static rules defined by users in the configuration file, but also dynamic ones, added during the simulation.
The simulation model had the important goal to preserve the original extensibility capacity offered by the simulator. Thus, the user can use the proposed security technologies simulation model for evaluating the characteristics of a new cryptographic protocol for example, or for testing the performance of a new tool for non-repudiation, or to ensure the integrity of users accessing applications running in a distributed environment. The user has the possibility to not only extend the job scheduling classes, but also the components that implement the security features (such as the processing unit, protocols and entities involved in the transfer of data or databases). He can even add additional functionalities.
As previously stated, one objective was to enable the definition of VOs that would allow resource sharing and collaboration between various different organizations (in this case between different regional centers). Thus a virtual organization can gather many resources in the system (e.g., processing units, servers, databases, etc.). In turn shared resources within a virtual organization may belong to different regional centers (Fig. 4).
The security policy implementation is based on several assumptions. For example, a VO defines at least one associated certificate. A job must also present the certificate that validates its identity and the name of the virtual organization in which it has to be processed in order to be executed. A user can delegate its certificate attesting its identity to all of the tasks he submits in the system. Every component of a VO can set restrictions on access to it. Any operation of the system must comply with imposed restrictions on access or use. The operations between two entities in the system requires the authentication of at least one of them, and by default any data transfer between entities requires the authentication of the components involved in the transfer. And finally, any component of the system reserves the right to filter out messages from unwanted sources.
The certificates associated with a virtual organization are designed to model the certification authorities (CA). These certificates must be X.509 certificates. Each system user or submitted jobs are associated with certificates signed by a trusted authority of the VO. In order to trust the certificate of an entity a mechanism must be defined that validates the trust in the CA that signed the certificate. In the implementation of the security model a factory object was defined to hold the multitude of certification authorities of the virtual organizations. Thus to verify and validate a certificate of another entity it is sufficient to interrogate the existing certificates from the factory. This will also contain all the trusted authorities of the defined VOs.

5. EXPERIMENTAL RESULTS

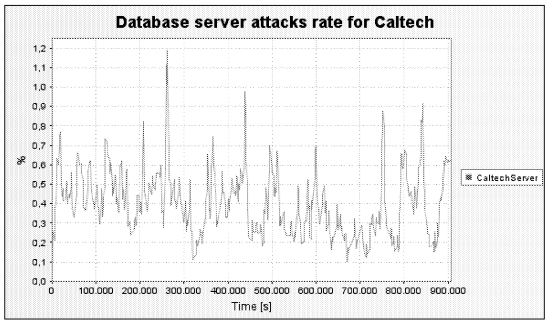
The evaluation of the proposed simulation model consisted of a series of simulation experiments. They were designed to test the capability of the model to correctly model security threats, as well as security solutions designed as

countermeasures (such as security policies, or various components for detecting security problems and react). The experiments also evaluated if the model is easily extendable and if it supports various scenarios. We were interested in possible performance degradation caused by its use in various scenarios.
One such simulation experiment (Fig. 5) evaluates the possibility of simulating an access policy enforcement mechanisms acting for all requests made to a database server. We were particularly interested if the simulation model allows the interpretation of security breaches in such a scenario. In particular, the experiment consists of two regional centers sharing several workstations and one database server.



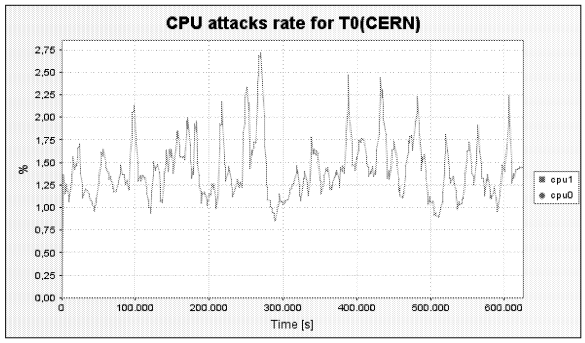
Figures 5: Simulation scenario

For this experiment we defined two custom jobs working with the shared database. One job creates and writes data to the database. The other connects and requests the data matching a specific pattern. We added a security policy on the database server, similar to a UNIX file system. For reading data the job (or the VO generating the job) must have read rights, for writing data it must have a corresponding right, and for removing data the job must have both read and write rights.



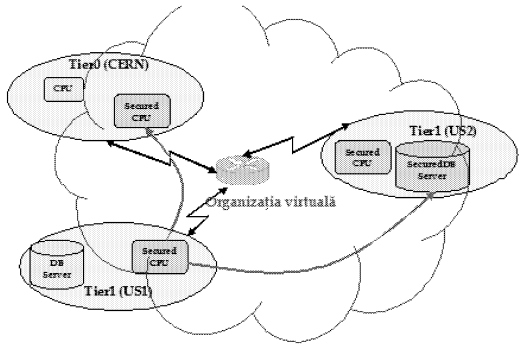
Figures 6: The percent of attacks recognized on the database side from the total number of generated requests

By extending the security model, we were able to concurrently simulate both ordinary jobs, as well as ones that tried different operations on the database without having sufficient rights. We logged and compared how many attacks were randomly generated (reads without the read right, etc.) versus how many attacks did the database server successfully recognized (Fig. 6).
Another scenario consisted of three regional centers sharing together, in the same VO, several workstations (Fig. 7). The experiments evaluated the authentication mechanism used for when jobs are submitted. We defined two types of jobs, with and without a valid certificate used for authentication.
Again, the addition of the authentication mechanism to the simulation model was done easily because of its extensibility. We evaluated and compared the number of generated non-valid authentication attempts versus the number of successfully recognized attempts to authenticate with such certificates (Fig. 7).



Figures 7: The attack rate

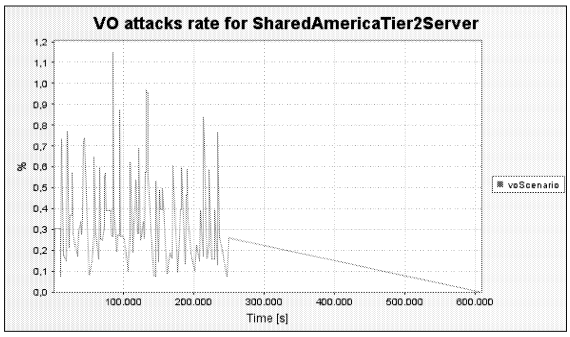
Next we evaluated the possibility to add a data filtering mechanism to the experiment. We defined a filtering rule for both a workstation within the virtual organization, as well as for the database server. In the scenario the virtual organization shares the resources belonging to three regional centers. The filtering mechanism is responsible with verifying all data received from one of these regional centers (Fig. 8).



Figures 8: The configuration used for the filtering experiment

For this experiment we added an extra database server within the regional center T1-US2. In case of this database server and the workstation shared by T0-CERN we defined filtering rules for all messages received from the network belonging to T1-US1. The experiment considered both data traffic between workstations belonging to different regional centers, as well as traffic generated towards the database server (reading and writing data into the database).
In the experiment the jobs in T0-CERN are waiting data or send requests to the database server. The jobs running in T1-US2 send data to T0-CERN, and the ones running in T1-US1 send messages to T0-CERN and requests to the database server in T1-US2. The data messages and database requests sent from T1-US1 are filtered because they come from addresses outside the filtering rule and are reported as

security problems. Figure 9 presents the attack rate on the database server in T1-US2.



Figures 9: The attack rate over the database server at VO level

In all these cases not only the security solutions designed and included in the proposed security model correctly handled possible attacks, but also the performance of the distributed simulated environment (throughput in the network or processing capability of the simulated processing units) was not affected beyond rendering the environment to be used anymore.

6. CONCLUSIONS

Large scale distributed systems are currently progressing from operational infrastructures to environments providing many "modern" capabilities. Security in large scale distributed systems represents an important research subject in this area. There are many solutions for enforcing security policies, or establish well-defined administrative domains between distributed organizations, secure communication taking place between distributed resources, etc. Validation of such security solutions is generally accomplished using real-world implementations.

Simulation is an attractive alternative to evaluating such solutions. Unfortunately, even though there are several simulators designed for distributed systems, with few exceptions they do not present solutions that can be used for the evaluation of security methods and techniques.

In this paper we proposed a simulation model suitable for evaluating methods and techniques designed to increase security in large distributed systems. As presented, this model has the characteristics needed to develop a wide range of security scenarios, being able to assess from solutions to secure data transfers to various mechanisms to assess the access management in a distributed system.

We presented implementation details of an extension of the MONARC simulator for distributed systems. The proposed components and mechanisms allow the evaluation of a wide range of security protocols and solutions, in the context of various distributed architectures. The implementations allow the evaluation of secure communication protocols, of mechanisms for authentication, of VOs, etc.

We also presented experimental results demonstrating the capability to correctly model security solutions for large scale distributed systems, and the capability of the model to pinpoint likely security problems in the simulated environments.

In the future we plan to extend the simulation model with the support for other authentication mechanisms (such as Kerberos tickets for example), include additional patterns of attack, and experiment with more security scenarios to further evaluate the generality of the model.

.

ACKNOWLEDGMENTS

The research presented in this paper is supported by national project "DEPSYS – Models and Techniques for ensuring reliability, safety, availability and security of Large Scale Distributes Systems", Project "CNCSIS-IDEI" ID: 1710, and by national project "TRANSYS – Models and Techniques for Traffic Optimizing in Urban Environments", Project "CNCSIS-PD" ID: 238.

REFERENCES

Dobre, C, and C. Stratan. 2004. "MONARC Simulation Framework", in *Proc. of the 3rd Edition of RoEduNet International Conference*, Timisoara, Romania.

Dobre, C, and V. Cristea. 2007. "A Simulation Model for Large Scale Distributed Systems", in *Proc. of the 4th International Conference on Innovations in Information Technology*, Dubai, United Arab Emirates.

Dobre, C., F. Pop, and V. Cristea. 2008. "A Simulation Framework for Dependable Distributed Systems", *First International Workshop on Simulation and Modelling in Emergent Computational Systems* (SMECS-2008), Portland, USA.

Johnston, S. 2004. "*Modeling security concerns in service-oriented architectures*", Accessed on 28.06.2010, from: http://www.ibm.com/developerworks/rational/library/4860.html , published 2004.

Naqvi, S., and M. Riguidel. 2005. "*Grid Security Services Simulator (G3S) – A Simulation Tool for the Design and Analysis of Grid Security Solutions*", Ecole Nationale Supérieur des Télécommunications, France.

The Globus security Team, 2010. "*Overview of the Grid Security Infrastructure*", Accessed on 18.06.2010, from http://www.globus.org/security/overview.html.

The Globus security Team. 2005. "*Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective*", Accessed on 17.06.2010, from http://www.globus.org/toolkit/docs/4.0/security/GT4-GSI-Overview.pdf, published 2005.

BIOGRAPHY

CIPRIAN DOBRE PhD, is lecturer with the Computer Science and Engineering Department of the University Politehnica of Bucharest. The main fields of expertise are Grid Computing, Monitoring and Control of Distributed Systems, Modeling and Simulation, Advanced Networking Architectures, Parallel and Distributed Algorithms. Ciprian Dobre is a member of the RoGRID (Romanian GRID) consortium and is involved in a number of national projects (CNCSIS, GridMOSI, MedioGRID, PEGAF) and international projects (MonALISA, MONARC, VINCI, VNSim, EGEE, SEE-GRID, EU-NCIT). His research activities were awarded with the Innovations in Networking Award for Experimental Applications in 2008 by the Corporation for Education Network Initiatives (CENIC).

RUNNING AGENT-BASED MODELS ON A DISCRETE-EVENT SIMULATOR

Bhakti S. S. Onggo
Department of Management Science
Lancaster University Management School
Lancaster LA1 4YX, United Kingdom
E-mail: s.onggo@lancaster.ac.uk

KEYWORDS
Agent-based simulation, agent-based model, discrete-event simulation, formal specification

ABSTRACT

One of the main obstacles hindering the use of large-scale agent-based simulation in practice is its scalability. The ability to run realistic and complex agent-based models is desirable to provide empirically and practically useful results. One of the suggested solutions is to run the agent-based models on top of a scalable parallel discrete-event simulation engine. This proposal raises a question whether an equivalent discrete-event model can be built for any agent-based model. This paper proves that an equivalent discrete-event model can be found for any agent-based model that conforms to a given specification. I show that a translator can be built to convert the agent-based model into an equivalent discrete-event model automatically and transparently. The advantage of this approach is that modellers do not need to change their modelling paradigm and at the same time a highly scalable parallel discrete-event simulator can be used to run the model.

INTRODUCTION

An agent-based model (ABM) 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. The ABM, just like other types of model, is used to represent a real world system to help us understand the system and make decisions. An ABM is commonly implemented as a piece of computer code and run using a simulator. The computer implementation of an ABM is referred to as the agent-based simulation (ABS). ABS has been applied in the physical sciences as well as the social sciences (Macal and North, 2007). There is a common issue which confronts many researchers in this field, namely, the scalability of ABS (Hybinette et al. 2006; Tesfatsion 2006). It is known that the scalability of ABS depends on many factors, notably the execution platform and the complexity of its ABM. The overall complexity of an ABM depends on the problem size (number of agents), the behaviour complexity of each agent, and the communication complexity between agents. Popov et al. (2003) have shown that the simulation of an ABM with a large number of agents on a cluster of PCs is possible. Recently, Perumalla and Aaby (2007) ran ABS with a large number of agents on GPUs. However, problem size is only one aspect of the ABM complexity. It is also important to see whether more realistic agent-based models with complex behaviour and complex communication network can be scaled-up to provide empirically and practically useful results.

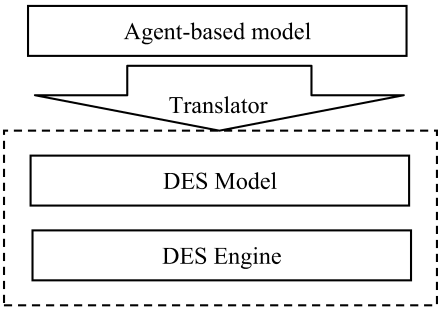
One of the proposed solutions is the use of the infrequent time advancement mechanism. In ABS, agents need to be synchronized (updated) regularly. The main objective of the infrequent time advancement approach is to minimize the number of updates and the number of updated agents during the simulation run. Lee et al. (2001) discussed a number of time advancement mechanisms that could be used in ABS. They are: fixed time advancement, variable time advancement, optimistic time advancement, complete resynchronization and partial resynchronization. Zaft and Zeigler (2002) provided an empirical analysis of the benefit of the variable time advancement in their artificial society simulation called XeriScape. This approach requires modellers to change their modelling paradigm from an agent-based modelling paradigm to a more event-based modelling paradigm. In simulation modelling, we select a certain portion of the real world system to be simulated for specific objectives. The process of capturing the essential elements of the system is referred to as conceptual modelling and the resulting model is referred to as a conceptual model (Pidd 2004, Chapter 3). Robinson (2008) supports a principle that a conceptual model should be independent of its computer implementation. Therefore it is possible that in the conceptual modelling we use an agent-based model and in the implementation we use a discrete-event simulator. This paper addresses the idea of running agent-based models on a discrete-event simulation (DES) engine. The idea of running an agent-based model on top of a discrete-event simulator has been proposed by a number of writers (Hybinette et al. 2006; Macal and North 2007). Many implementation requires modellers to change their modelling paradigm from agent-based to discrete-event. This paper proves that it is possible to transform an ABM that conforms to a given specification into a DES model which can be run on a DES engine. The translation process is transparent to the modellers. The main advantage of this approach is that modellers do not need to change their modelling paradigm (that is, ABM), and at the same time, a scalable parallel DES engine can be used to improve the overall simulation performance.

The remainder of this paper is organized as follows. Section 2 provides an overview of conceptual modelling in simulation. Section 3 discusses the proposed approach on

how to run an ABM on top of a DES engine without changing the modelling paradigm. Section 4 addresses the performance issues related to the proposed approach. Finally, I present the conclusion and highlight some avenues for future work in section 5.

PROPOSED APPROACH

Figure 1 shows how an ABM can be run on top of a DES engine. The agent-based modelling paradigm is used at the conceptual model level. At the implementation level, the translator will translate the ABM into a discrete-event model that is ready for execution using a DES engine. In an ideal case, the translation process should be transparent to the modellers. Hence, modellers should not change the way they model real world systems. This proposal raises a question whether an equivalent discrete-event model can be built for any agent-based model. Among the multi-agent systems (MAS) community, researchers have been proposing the use of DES in the design and validation of a multi-agent system. Uhrmacher and Schattenberg (1998) developed JAMES, a discrete-event simulator for agent modelling. Riley and Riley (2003) developed a similar system called SPADES. These works show that it is possible to simulate a MAS model using a discrete-event simulator. Since the basic structure of an agent in MAS and ABS are not significantly different, it should be possible to develop an equivalent discrete-event model at least for a subset of agent-based models. This paper differentiates between the use of simulation in MAS and ABS. In MAS, simulation is used mainly in the design of artificial agents. In ABS, agents are the main components of the simulation model. In addition, ABS focuses more on the representation of human behaviour, social interaction and the emergent behaviour. Macal and North (2007) provides a good tutorial on the ABS and how it is different from MAS.



Figures 1: ABM on a DES Engine

Formal Proof

I use the formal specification of a discrete-event model and an agent-based model to show that an equivalent discrete-event model can be built for any agent-based model that conforms to the specification given later in this section. I use Zeigler's formal specification of a discrete-event model because it is applicable to wide range of discrete-event models (Zeigler 1976). For the agent-based model, I use the formal specification described in Wooldridge (2002) and expand it whenever necessary. This specification is chosen because it is very generic and should cover wide-range of agent-based models.

DEFINITION 1. A discrete-event model is defined as a tuple of seven components $<X, \Sigma, Y, \delta int, \delta ext, \lambda, ta>$ where:

- X is the set of events in the system.
- Σ is the state of the system.
- Y is the set of output variables.
- δint: $\Sigma \rightarrow \Sigma$ is the internal function that changes the current system state ($\sigma \in \Sigma$) at time t to a new state $\sigma' \in \Sigma$ at time $t+ta(\sigma)$ provided there is no event $x \in X$ which occurs between time t and $t+ta(\sigma)$
- δext: $\Sigma \times T \times X \rightarrow \Sigma$ is the external function that changes the current system state ($\sigma \in \Sigma$) at time t to a new state $\sigma' \in \Sigma$ because of the occurrence of an event $x \in X$ at time $t+\Delta t$ (where $0 \leq \Delta t \leq ta(\sigma)$)
- ta: $\Sigma \rightarrow R^{+}_{0,\infty}$ is the time when the current system state ($\sigma \in \Sigma$) is scheduled to change (see δint).
- Finally, λ: $\Sigma \rightarrow Y$ is the output function that gives a set of output values (Y) given the current system state (Σ).

A discrete-event model that conforms to definition 1 must specify a set of events and the system state. Given an initial state $\sigma_0 \in \Sigma$, the simulation starts by advancing its current time t to either $t+ta(\sigma_0)$ or an earlier time depending on whether or not an event $x \in X$ occurs at time earlier than $t+ta(\sigma_0)$. In the later case, the current time t is advanced to the time when the event occurs. Next, the state of the system may change based on either δint (the first case) or δext (the later case). The change in the system state may change the simulation output. This process is repeated until a stopping condition is met.

DEFINITION 2. An agent-based model is defined as a tuple $<A, E>$ where A is a set of agents ($A = \bigcup a^k$ and $1 \leq k \leq N_{agents}$,) and E is the environment.

The agent-based model in definition 2 is very generic; an agent-based model is formed by two components: a set of agents (definition 3) and its environment (definition 4). Before the two components are defined, let us define the following terms:

- s^k_j is the j^{th} set of state variables that is seen by agent k where $1 \leq k \leq N_{agents}$
- $S^k = \bigcup s^k_j$, i.e., all sets of state variables seen by agent k where $0 \leq j \leq N^k_{states}$
- $S = \bigcup S^k$, i.e., all sets of state variables seen by all agents
- α^k_j is the j^{th} action done by agent k in response to a set of state variables s^k_j
- $\Lambda^k = \bigcup \alpha^k_j$, i.e., all actions done by agent k
- $\Lambda = \bigcup \Lambda^k$, i.e., all actions done by all agents
- The i^{th} run of agent a^k (i.e., r^k_i) is the i^{th} sequence of interleaved s^k_0, α^k_0, s^k_1, α^k_1, ….

- $R^k = \bigcup r_i^k$, i.e., the set of runs of agent k, where $1 \le i \le N_{runs}^k$
- R_S^k is R^k that ends with an s_j^k
- R_Λ^k is R^k that ends with an α_j^k
- $R_S = \bigcup R_S^k$, $R_\Lambda = \bigcup R_\Lambda^k$

DEFINITION 3. An agent k is defined as a function $a^k : R_S^k \rightarrow \Lambda^k$.

DEFINITION 4. An environment E is defined as a tuple <Σ, τ > where Σ is the system state and $\tau : R_\Lambda \rightarrow \Sigma$ is a state transformer function that changes the system state based on $r_i^k \in R_\Lambda$.

Definition 3 shows that an agent (a^k) performs an action ($\alpha_j^k \in \Lambda^k$) based on past states and its past actions ($r_i^k \in R_S^k$). Definition 4 shows that the first component of an environment is the system state (Σ). An action from an agent will influence the state transformer function (τ) in the environment that is responsible for the change in the system state. Note that in a non-deterministic agent-based system, a given run $r_i^k \in R_\Lambda$ may change the current system state into more than one possible states, but only one state will be chosen (randomly or based on a pre-determined rule). In other words, for the same action from an agent, the environment may act differently.

LEMMA 1. In a discrete-event model, a set of agents A can be implemented using X and δext.
Proof. Based on definition 3, each agent is specified as a function (a^k) that selects one action ($\alpha_j^k \in \Lambda^k$) based on past states and actions ($r_i^k \in R_S$). The action will influence the state transformer function τ that will decide new values for the state variables. This can be implemented by defining an event $x \in X$ for each action that can be performed by an agent and an external function δext that implements the effect of each action (i.e., the change in the state variables). Therefore a set of agents A can be implemented using a set of events X and an external function δext. □

LEMMA 2. In a discrete-event model, the environment E = <Σ, τ > can be implemented using <Σ, δext, δint, ta>.
Proof. Let us expand the system state in the discrete-event model to include past states and actions (R_S and R_Λ) in addition to a set of state variables (S). In other words, Σ = {S, R_S, R_Λ}. R_S and R_Λ will enable δext and δint to implement τ. Based on how the system state changes, the environment E in an agent-based system can be categorized as either static or dynamic. A static environment is an environment in which its state will only change due to actions performed by agents. To implement this, we can use a time advancement function $ta(s) = \infty$ for any $s \in S$. In this case, the changes in the system state will be triggered by an external function δext. In contrast, the system state in a dynamic environment is constantly changing, even if there is no action performed by any agent. This can be implemented using a time advancement function with $0 < ta(s) < \infty$. In this case, the change from state s at time t to a new state at time $t+ta(s)$ is controlled by an internal function δint (if there is no event occurs between t and $t+ta(s)$) or δext (if otherwise). In other words, $\tau : R_\Lambda \rightarrow \Sigma$ can be implemented using δext, δint, and ta (note that the expanded Σ includes R_S and R_Λ). Therefore, the environment E can be implemented using <Σ, δext, δint, ta>. □

THEOREM 1. An agent based model <A, E> can be implemented using a discrete-event model <X, Σ, Y, δint, δext, λ, ta>.
Proof. It can be derived from lemma 1 and 2. Intuitively, an agent-based simulation will not be useful if it does not produce any output. Hence, if Y is the set of output variables, then we need an output function $\lambda : \Sigma \rightarrow Y$ that maps the current system state onto a set of output variables. □

Lemma 1 and lemma 2 show how the two components of an agent-based model, i.e., a set of agents A and their environment E, can be built from five of the seven components of a discrete-event model. Theorem 1 shows that it is possible to find an equivalent discrete-event model for any agent-based model that conforms to the specifications given in definition 2.

Translation Process

The use of formal specification in the proof implies that the process of generating a discrete-event model from a given agent-based model can be done automatically. Figure 2 shows an ABM and its equivalent discrete-event model. Based on definition 2, an ABM needs to specify its initial state, the stopping condition, the output function, the state transformer function, and a set of agents. The bold lines in the discrete-event model are taken directly from the agent-based model. The remaining lines are the same for any given ABM. Therefore, the translation process can be done automatically and it is transparent to the modellers.

The detailed explanation of the pseudo code for the discrete-event model in figure 2 is as follows. Line 1 initializes the simulation clock. In line 2, the set of state variables is set to s_0. This initial set of state variables' values is then used as the first element in the sequence R_S. Finally, R_Λ is set to an empty list. Note that σ is an instance of Σ. All actions are scheduled in line 3. If the ABS uses a fixed-increment time-advance mechanism then Δt_x is constant for every event $x \in X$. Lines 4 to 9 show the typical main iteration in discrete-event simulation. The simulation output is produced in line 8. Procedure Execute in line 11 accepts an event and the system state as its parameters. Based on the event and R_S, an action is chosen and is executed (lines 12 and 13). Note that definition 3 implies that an agent can only perform an action at any given time. The system state is updated in lines 14 to 16. First, the action is added to the end of R_Λ. Next, the state transformer function is executed to update the current set of state variables. Finally, R_S is updated by adding the new set of state variables' values to the end of the sequence. In line 17, we schedule the same event $x \in X$ to occur at time $t+\Delta t_x$.

```
Agent-based model
Initial state: s0
System state: Σ = {S, RS, RA}
General methods:
•  Stopping condition: IsComplete()
•  Output: y ← λ(Σ)
•  State transformer: S ← τ(RA)
Agents: A set of α ← a^k(RS)

Discrete-event model
 1. t ← 0
 2. σ.S ← s0; σ.RS ← <s0>; σ.RA ← <>
 3. ∀x ∈ X • Schedule(x, t+Δtx)
 4. while (~IsComplete()) {
 5.   x ← GetNextEvent()
 6.   t ← GetEventTime(x)
 7.   Execute(x, σ)
 8.   y ← λ(σ)
 9. }
10.
11. Execute(x∈X, σ∈Σ) {
12.   a ← GetAgent(x)
13.   α ← a(σ.RS)
14.   σ.RA ← σ.RS ∪ <α>
15.   σ.S ← τ(σ.RA)
16.   σ.RS ← σ.RA ∪ <σ.S>
17.   Schedule(x, t+Δtx)
18. }
```

Figures 2: An ABM and its equivalent discrete-event model

In practice, how the elements of an ABM (such as, a set of actions and the state transformer function) are specified depends on the chosen agent-based simulation library. For example, figure 3 shows how an ABM called SugarScape is specified using Repast/J agent modelling toolkit (North et al. 2006). SugarScape is an artificial society model that simulates the behaviour of agents (people) located on a landscape of sugar (to represent a generalized resource). Agents move around to find a location with the most sugar and eat the sugar. Epstein and Axtell (1996) provides a more detailed explanation on the SugarScape model. Figure 3 shows that the model (SugarModel) is formed by a list of agents (line 3) and an environment (space in line 5). The initial state is defined in line 8. The agents (SugarAgent) are created in lines 10 to 14. An action called step is defined for each agent. At every simulation timestep all agents will perform their actions by executing their step methods (line 25). After the actions are completed, the system state will be updated by executing the state transformer function updateSugar (line 27). This iteration is repeated from time 0 until a stopping condition is satisfied (line 31).

```
 1. public class SugarModel extends
 2. SimModelImpl {
 3.   private ArrayList agentList =
 4.     new ArrayList();
 5.   private SugarSpace space;
 6.   ...
 7.   private void buildModel() {
 8.     space = new
 9.       SugarSpace("sugarspace.pgm");
10.     for (int i = 0; i < numAgents; i++) {
11.       SugarAgent agent = new
12.         SugarAgent(space, this);
13.       agentList.add(agent);
14.     }
15.     ...
16.   }
17.   private void buildSchedule() {
18.     ...
19.     class SugarRunner extends BasicAction{
20.       public void execute() {
21.         ...
22.         for (int i = 0;
23.         i < agentList.size(); i++) {
24.           ...
25.           agent.step();
26.         }
27.         space.updateSugar();
28.         ...
29.       }
30.     }
31.     schedule.scheduleActionBeginning(0,
32.       new SugarRunner());
33.   }
34.   ...
35. }
```

Figures 3: SugarScape in Repast/J

PERFORMANCE ISSUES

The first issue relates to the fact that many ABMs inherently adopt the fixed-increment time-advance mechanism. In this situation, all being equal, the performance achieved from running the ABM on top of a DES engine may not be different from running it using a time stepped based ABS engine. However, in some cases, we can find agent-based models where some of the agents perform an action for certain $r_i^k \in R_S$ only. Let p be the probability that this condition is spotted in the system. This means that every Δt some agents will decide whether or not they will execute an action. In other words, every Δt all agents perform a Bernoulli trial where the probability of success is the probability that an action is executed, i.e., p. It is known that the number of Bernoulli trials until the first success follows the geometric probability distribution with an expected value of $1/p$. Let N_{agents} be the number of agents, T be the number of timesteps in a simulation run, B be the computation time for generating the Bernoulli distribution and G be the computation time for generating the geometric distribution ($B < G$). Let us assume that all actions require the same amount of time to execute, i.e., A. On one processor, the execution time of the fixed timestep time advancement that is commonly used in ABS is $W = N_{agents}\, T\, B + p\, N_{agents}\, T\, A$. The first term is the time to execute all Bernoulli trials and the second term is the time to execute all actions (note that only $p \times N_{agents}$ number of agents perform an action in every timestep). Similarly, the execution time of running an ABM on top of a DES engine is $W' = p\, N_{agents}\, T\, G + p\, N_{agents}\, T\, A$. The theoretical performance improvement (Π) is:

$$\Pi = \frac{W}{W'} \frac{N_{agents}TB + pN_{agents}TA}{N_{agents}(Tp)G + pN_{agents}TA} = \frac{B + pA}{pG + pA}$$

This analysis shows that the performance can be improved as long as $p < \frac{B}{G}$. The improvement may not be significant if the behaviour complexity of most agents is high. Hence the total time to execute the actions ($p \times A$) is high. The translation process can make use of this by automatically converting any Bernoulli distribution used in the agent-based

model into a geometric distribution in the discrete-event model. I have not implemented this. However, the empirical result should be similar to what has been reported in Zaft and Zeigler (2002). Although performance improvement is possible even on a single processor, it should be noted that the main objective of converting a Bernoulli distribution into a geometric distribution is to allow the parallel DES engine to exploit more parallelism from the model.

There are two other related performance issues that need to be addressed. The first issue is related to the environment, i.e., how information (state variables) in the environment is distributed. To avoid the potential bottleneck, the state variables may need to be distributed across processors. Ideally, the translator should handle this transparently. The second performance issue is related to the movement of agents. It is possible for agents to move from one spatial location in the environment to another. A number of solutions to these two problems have been proposed (Lees et al. 2004a, 2004b; Popov et al 2003). However, the scalability of these solutions on a large number of processors remains a challenge.

CONCLUSIONS AND FUTURE WORK

This paper shows how the idea of running an agent-based model on top of a discrete-event simulation engine can be realized. The main contribution of this paper is the analytical proof that an equivalent discrete-event model can be found for any agent-based model that conforms to the specification given in this paper. It implies that a translator can be built to convert an agent-based model into an equivalent discrete-event model automatically and transparently. The advantage of this approach is that modellers do not need to change their modelling paradigm. In cases where agents do not perform actions in every timestep, the simulation performance can be improved even on a single processor. Most importantly, it improves the inherent parallelism in the model which enables a highly scalable parallel discrete-event simulation (PDES) to exploit the parallelism. There have been a number of highly scalable PDES engines reported in the literature, in particular, the library used in this paper, μsik has been proven to be scalable on up to 10^4 processors on an IBM Blue Gene supercomputer (Perumalla 2007). Therefore, this approach has the potential to tackle the scalability issue in agent-based simulation. The issues on how to implement the environment and the migration of agents will affect the scalability and have to be addressed in our future work.

REFERENCES

Epstein, J.M. and R. Axtell, 1996. *Growing Artificial Societies: Social Science from the Bottom Up*. Brookings Institution Press, Washington, D.C.

Hybinette, M., E. Kraemer, Y. Xiong, G. Matthews and J. Ahmed. 2006. "SASSY: A Design for a Scalable Agent-Based Simulation System Using a Distributed Discrete Event Infrastructure". In *Proc. of Winter Simulation Conf.*, 926-933.

Lee, S., A. Pritchett and D. Goldsman. 2001. "Hybrid Agent-Based Simulation for Analyzing the National Airspace System". In *Proc. of Winter Simulation Conf.*, 1029-1036.

Lees, M., B. Logan, T. Oguara and G. Theodoropoulos. 2004. "Modelling Environments for Distributed Simulation". In *Proceedings of the 1st International Workshop on Environments for Multi-Agent Systems*, 150-167.

Lees, M., B. Logan, R. Minson, T. Oguara and G. Theodoropoulos. 2004. "HLA_Agent: Distributed Simulation of Agent-Based Systems with HLA". In *Proceedings of the 4th International Conference Computational Science*, 881-888.

Macal, C.M. and M.J. North. 2007. "Agent-Based Modeling and Simulation: Desktop ABMS". In *Proceedings of Winter Simulation Conference*, 95-106.

North, M.J., N.T. Collier and J.R. Vos. 2006. "Experience Creating Three Implementations of Repast Agent Modeling Toolkit". *ACM Transactions on Modeling and Computer Simulation*, 16, No. 1, 1-25.

Perumalla, K. 2007. "Scaling Time Warp-based Discrete-Event Execution to 10^4 Processors on a Blue Gene Supercomputer". In *Proceedings of the 4th International Conference on Computing Frontiers*, 69-76.

Perumalla, K. and B.G. Aaby. 2007. "Data Parallel Execution Challenges and Runtime Performance of Agent Simulations on GPUs", Oak Ridge National Laboratory (November).

Popov, K., V. Vlassov, M. Rafea, F. Holmgren, P. Brand and S. Haridi S. 2003. "Parallel Agent-Based Simulation on a Cluster of Workstations". In *Proceedings the 9th Euro-Par Conference*, 470-480.

Pidd, M. 2004. *Computer Simulation in Management Science*, 5th edition. John Willey and sons, Chichester, England.

Riley, P. and G. Riley. 2003. "SPADES: A Distributed Agent Simulation Environment with Software-in-the-loop Execution". In *Proc. of Winter Simulation Conf.*, 817-825

Robinson, S. 2008. "Conceptual Modelling for Simulation Part I: Definition and Requirements". *Journal of the Operational Research Society*, 59, No. 3, 278-290

Tesfatsion, L. 2006. "Agent-based Computational Economics: A Constructive Approach to Economic Theory". In *Handbook of Computational Economics: Agent-based Computational Economics*, L. Tesfatsion and K.L. Judd (Eds). North-Holland, Amsterdam, The Netherlands, 831-880.

Uhrmacher, A.M. and B. Schattenberg. 1998. "Agents in Discrete-Event Simulation". In *Proceedings of the 10th European Simulation Symposium*.

Wooldridge, M. 2002. *An Introduction to Multi Agent Systems*. John Wiley and sons, Chichester, England.

Zaft, G. and B.P. Zeigler, "Discrete Event Simulation of Social Sciences: The XeriScape Artificial Society". In *Proc. of the 6th World Multiconf. on Systemics, Cybernetics and Informatics*.

Zeigler, B.P. 1976. *Theory of Modeling and Simulation*. John Wiley, New York.

AUTHOR BIOGRAPHY

BHAKTI S. S. ONGGO is a lecturer at the Lancaster University Management School. He received his MSc in Management Science from the Lancaster University and completed his PhD in Computer Science from the National University of Singapore. His research interests are in the areas of simulation methodology (conceptual model representations, modeling paradigms such as discrete-event and system dynamics), simulation technology (parallel and distributed simulation) and simulation applications. His current research includes the application of parallel simulation in policy analysis and simulation conceptual model representation. His email address is s.onggo@lancaster.ac.uk.

Ant-Algorithm for the automatic optimization of material flow modelling for complex simulation models

Dr. Christoph Laroque
Sebastian Krimmer
Robin Delius
Heinz Nixdorf Institute, University of Paderborn
Fuerstenallee 11
D-33102 Paderborn, Germany
E-mail: {christoph.laroque, skrimmer, robin.delius}@hni.uni-paderborn.de

KEYWORDS
Ant-algorithm, automatic modelling, material flow simulation, complex simulation models

ABSTRACT

This article describes an automatic approach for a specific modelling support within material flow simulation studies. Based on the adoption of classical algorithms from the artificial intelligence, especially ant colony algorithms, parameters at forks as well as joins within material flow models are adjusted automatically to their optimal values. The designed method is implemented within the material flow simulation tool d³FACT insight, developed at the Heinz Nixdorf Institute of the University of Paderborn, Germany. First attempts show good results during application, since even initial worst-case parameter settings are automatically arranged to a nearly „optimal" solution.

MOTIVATION

The current situation in many areas of the industrial manufacturing is characterized by abbreviated product lifecycle, customer-oriented production and an increased diversity of variants. In order to manufacture cost- and also time-efficiently, new products are almost completely designed, shaped and optimized with the support of computer techniques. The advantages of this methodology accrue inter alia in diminished development expenses and production periods. The progressive digitalization goes beyond plain product modelling and refers to the planning, implementing and the control of all relevant processes of manufacturing and logistics (VDI 2008). An established way of planning, stabilization and improvement of manufacturing processes is the material flow simulation. Typical problems being examined in this domain are for example planning hedges, lot size planning and especially the development and adaption of control rules in an existing system (Law and Kelton 2000). Within this development especially with complex simulation models, the simulation expert faces the question, which parameter he should use for the particular control rules on a certain material flow branching or on merging, in order to improve for instance the throughput of the overall system. This is where this work is present; the user is supported in the modelling phase by using an automated procedure to find the best possible setting of model parameters. It is based on a meta-heuristic in the field of artificial intelligence (Dorigo et al. 1991).

MATERIAL FLOW SIMULATION WITH D³FACT INSIGHT

At the Heinz Nixdorf Institute (University of Paderborn), there is a material flow simulator being developed for years, which should create new areas of application of process simulation and support the user better by the implementation of simulation studies (Laroque 2007; Dangelmaier and Laroque 2008). The performance of a modelled system, achieved by modelling and subsequent simulation, often decides about the variety of a particular design variant and therefore about the results on the reality. Particularly with regard to the modelling of each system it raises the question of how even experienced simulation experts can systematically improve the resulting solutions manually. Especially, if stochastic influences in the model arise, in the real application the problem of the number of required simulation runs is complicating this task. With the evaluation and refinement of each configuration is a more or less large computational and also time-consuming effort combined, due to the stochastic effects there are several simulation runs required for each scenario (Law and Kelton 2000). An optimization of the model parameters of a given configuration is taken place in an operational application mainly by iterative model modification. With regard to the objective target of this work raises the central question: Is it possible to use methods of artificial intelligence from the field of operations research (OR) to reach an automated optimization of a concrete material flow model with respect to a maximization objective?

NATURE AS PARAGON

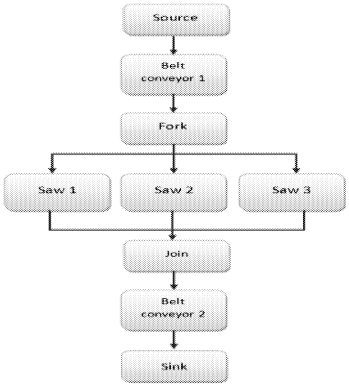
Many of the methods used today in the area of operations research, such as heuristic solution procedures, have been inspired by the nature. As one example, it can be observed, that: Ants searching for food create paths between their nest and the appropriate food source. Astonishing is the observation, that these self-organized ant paths always seem to find the direct link. Biologists were able to confirm in a so-called Double-Bridge experiment that ants always find the shortest way on their search for food (Goss et al. 1989). Ants use two fundamental properties: Ants mark their way with pheromones, which attract other ants. The decision for

the best route is made by probability sampling. The higher the concentration of pheromones on the path, the more likely the path is to be chosen. The Italian Marco Dorigo was the first to use the food search of ants to solve an optimization problem (Dorigo 1991).

IDEA: THE ANTI-ANT ALGORITHM

As part of this work, an ant algorithm is used to identify the most efficient throughput within an existing material flow model by adapting the different control parameters on material flow branches and conflations in order to maximize the total material flow. A simulative evaluation of the found configuration(s) is supposed to validate respectively evaluate the selection of the ant algorithm in a subsequent step. The ant colony optimization algorithm in this application answers the question, which path an ant has to follow in order to use the fastest way to the target (simply by finding the time-weightened, shortest path). As the standard ant colony optimization algorithm does not take throughput limitations of each path into account, it is not applicable in its original form. In this paper the original algorithm was modified, so it can now answer the question, how many ants (resp. workpieces in our case) can get to the target in a particular time interval (which corresponds to the maximum throughput of the model). The developed anti-ant algorithm enhances the pheromone concentration on the paths the ant did not walk on. This is exactly the opposite of the original ant colony optimization approach. To develop the anti-ant algorithm, a

Figure 1: Simple test model for anti-ant algorithm



simple model was chosen (see figure 2). The allocation strategy, which is used by the fork to allocate workpieces onto the saws, must be manipulated by the anti-ant algorithm in a way to maximize the throughput of this model. The anti-ant algorithm is called every time when a saw has finished a workpiece. In the other runs, it is ensured that a "weighted distribution", based on the performance of the fork, can represent material flow and results from the algorithm. For example, different machine capacities or services can be considered, as the subsequent analysis shows.

IMPLEMENTATION

In the chosen example of the model, logical links lead from the fork to each saw. The higher the pheromone concentration on the edge of the material flow model is, the more likely a workpiece will be moved along this path. To each saw belongs an attribute "final processing", which corresponds to the last required time for machining a workpiece. When a workpiece is finished by one of the saws, the developed anti-ant algorithm will be invoked (see pseudo code). Based on the processing period of a workpiece, the pheromone concentrations are adapted on all other routes, except the path which has been passed by the workpiece. The pheromone concentration being added on a path is calculated by the relation between the processing period of the workpiece and the last processing period of the saw to which the path leads. Programme expression 1 shows the fundamental strategy of the anti-ant algorithm in pseudo code. Attention should be paid to this simple implementation of the anti-ant-algorithm, because it does not involve features such as the evaporation of pheromones over time. The point of time at which saw i gets a workpiece from the fork is stored in the array pt. After saw i has finished the workpiece, it invokes the anti-ant algorithm. As can be seen in expression 1, the processing time (The time duration for finishing the last workpiece) is stored in the array pt. The array fork stores double values, which are used to calculate the allocation of workpieces to the saws in the model (e.g. fork(f) stores the strength of the pheromone trail to saw f). The first time the anti-ant algorithm is called, no processing times are stored for each saw that is why they are estimated in line 4, by setting those to the processing time of saw i. The loop in line 2 calculates then for all other saws (f ≠ i) new probabilities, by adding the ratio of processing times of saw i and f to the double value of f. At last the fork allocation is updated in line 7.

```
anti_ant(i){
1. pt(i) = now()-pst(i)
2. for(f = 0; f < fork.length; f++){
3.     if (pt(f)==0){
4.       pt(f)=pt(i)
5.     }
6.     if (f != i){
7.       fork(f)=fork(f)+ (pt(i)/pt(f))
8.     }
9. }
10. updateforkallocation(fork)
}
```

In the example model are 3 saws, therefore the arrays pt and fork have the length n=3 each. One workpiece is forwarded from the fork to saw i with the following probability:

$$P(i) = \frac{fork(i)}{\Sigma_{f=1}^{n} fork(f)} \quad (1)$$

If the fork tries to transport one workpiece to a busy saw i, the fork is blocked, although the fork could forward the workpiece to another free saw.

TEST SCENARIO

In the first simulation model the processing times of each model component can be seen in table. Fork and join don't consume time. One workpiece can reach the sink after 13 time units. Because all saws work at the same speed, the

optimal distribution is P(i) = 1/n. So, in one third of all cases, the fork transports a workpiece to saw i, if an optimal allocation strategy has been set. The model has been simulated three times with a simulation length of 10.000 time units. In the first run the fork transported only pieces to saw 1, which is a conceivable bad allocation.

Table 1: Processing Times of Model Components

Component	Processing time
Source	1
Belt conveyor 1, 2	1
Saw 1, 2, 3	10

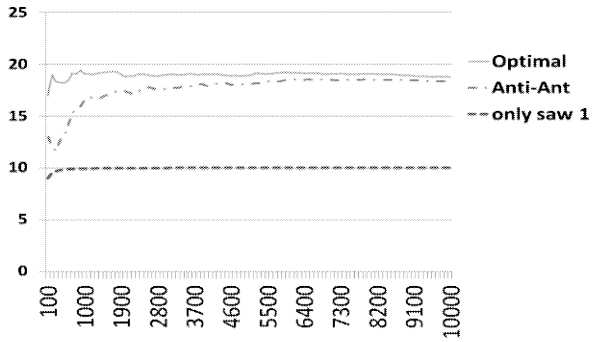


Figure 2: Evaluation of test scenario 1

The question arises of how it is possible to create an optimal setting with the developed anti-ant algorithm, but also whether it is possible to create an evolutionary development towards an "optimal" setting of the components based on bad starting values. Based on the bad allocation of the first simulation run, the allocation has been improved during the second run with the help of the anti-ant algorithm. The third run used the optimal allocation to show how efficient the ant-ant algorithm approaches to the maximum throughput of this model. The resulting chart (see figure 3) shows the number of finished workpieces per 100 time units for each run.

In the second simulation model the processing time of each saw has been altered. As before, the model within the actual simulation experiment has been simulated three times with a simulation length of 10.000 time units. The first run used a bad allocation as well by transporting all workpieces to saw 1. Based on this bad allocation, the anti-ant algorithm improved the allocation strategy of the fork in the second run. The third run used the optimal allocation, which can still be easily calculated for this model. Whereas the optimal allocation for each saw i is calculated by the ratio of 'workpieces per time unit of saw i (wze(i))' and 'the sum of workpieces per time unit of all saws (wze(f))':

$$P(i) = \frac{wze(i)}{\Sigma_{f=1}^{n} wze(f)} \qquad (2)$$

The obtained results show that the anti-ant algorithm works in its basic construction and that the classical restrictions in material flow models are already included in this simple implementation. Gradually approaches the algorithm automatically to an optimal setting. By this way an arbitrarily distribution of a model can be optimized quickly and also depending on other model parameters. The implementation of the algorithm, or the required blocks in the material flow model is deliberately kept flexible. Related sections can be defined easily by the operator and be used for the calculation of the final processing period. Therefore, not just single machines, but also the use of many material flow chains can be defined and optimized.

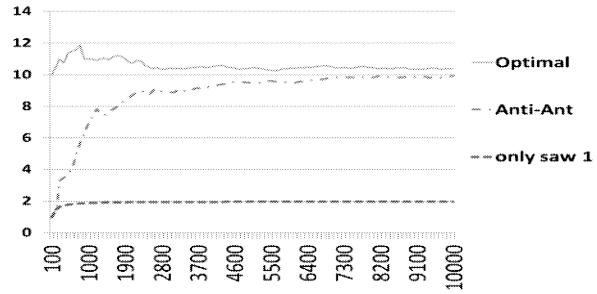


Figure 3: Evaluation of test scenario 2

Additionally, it is possible to vary, due to the structure of the used material flow simulator, the amount of saws or their processing period in a model about runtime and to calculate the algorithm automatically with the adaption to the new, optimal turnout distribution.

CONCLUSION AND OUTLOOK

This article describes the design of a procedure based on the group of ant algorithms in order to support simulation experts in creating simulation models for material flow simulations. Based on the assumption that the maximization of the throughput is a desirable aim of planning, first implementations of the approach and the evaluation presented show a good performance. Thereby, the initial setting of model parameters can be selected rather beneficial or disadvantageous at the start of improvement. The parameter between the selected material flow routes is improved gradually by the application of the presented algorithm. Further experiments have to show how the process of identifying these "optimal" parameters by extensions of the actual algorithm can be accelerated. Beyond, the concept must be validated in further studies with clearly more complex models.

REFERENCES

Dangelmaier W. and C. Laroque: „Immersive 3D-Ablaufsimulation von richtungsoffenen Materialflussmodellen zur integrierten Planung und Absicherung von Fertigungssystemen". In Leobener Logistik Cases - Management komplexer Materialflüsse mittels Simulation. DUV Verlage, 2007.

Dorigo, M.; V. Maniezzo; A. Colorni. 1991. "Ant System: An outocatalytic optimizing process". working paper No. 91-016 Revised. Italy, Politecnico di Milano.

Goss, S.; S. Aron; J.-L. Deneubourg; J.M. Pasteels. 1989. "Self-organized shortcuts in the Argentine Ant". Naturwissenschaften. 76:579–581.

Laroque, C. 2007. „Ein mehrbenutzerfähiges Werkzeug zur Modellierung und richtungsoffenen Simulation von wahlweise funktions- und objektorientiert gegliederten Fertigungssystemen“. HNI Verlagsschriftenreihe. Paderborn.

Law, A. M. and W. D. Kelton. 2000. Simulation Modeling and Analysis. McGraw-Hill.

VDI: Digitale Fabrik Grundlagen VDI-Richtlinie 4499. Blatt 1. 2008. VDI-RICHTLINIEN.

ADDING NEW DEPENDENCIES TO ACTA FRAMEWORK

Mourad kaddes, Majed Abdouli and Rafik Bouaziz
MIRACL Lab.
University of Sfax
Tunisia
E-mail: Mourad.kaddes@gmail.com
Majed.abdouli@gmail.com
Raf.bouaziz@fsegs.rnu.tn

KEYWORDS
ACTA, Commit Dependency, Abort Dependency, Pre-Commit, Conditional Dependency.

ABSTRACT

Due to the diversity of extended transaction models, their relative complexity and for some of them their lack of formalization, the characterization and the comparison of these models become delicate. Moreover these models capture only one subset of interaction which can be found in the spectrum of the possible interactions. In front of this established fact, the framework ACTA was introduced. Our contribution in this field is two fold: (i) we extend ACTA by adding many dependencies for capturing a new interaction between transactions in real time and mobile environment and (ii) we derive a new transaction model by combining the specifications of existing models.

INTRODUCTION

Although powerful, the transaction model adopted in traditional data base systems is found lacking in functionality and performance to satisfy the needs of collaborative, long-lived, real-time activity and distributed applications. Hence, many extensions to the traditional model were developed based on relaxing ACID properties referred to herein as extended transaction model. The first extended model is the nested transaction proposed by (Moss 1985). Mostly, the others models are based on the idea of nesting transactions. All extended transaction models described in literature, split/join transactions, Saga, kangaroo transactions, nested split transaction… satisfy a part of users' applications and satisfy a particulars needs. This diversity of models, its complexity and its lacks of formalism encouraged Chrysanthis et al. (Chrysanthis and Ramamritham 1994) to define the framework ACTA which made it possible not only to formalize the various existing models of transactions but to define new models of transactions. Since ACTA is just a base many extension are proposed in literature. Schwarz et al in (Schwarz et al. 1998a, Schwarz et al. 1998b, Schwarz et al. 1998c), have proposed an extension of ACTA and introduce the notion of transaction closure that is defined as generalisation of nested transaction. They classified the dependency between transactions in two categories: Termination dependency and Execution dependency. They also introduce the N-ary termination dependency that permits to express dependency between more than two transactions. As the same way in (Abdouli 2006), the authors introduce some new dependencies to express the dependency between transactions in imprecise computations. We propose to extend ACTA.
This paper is organized as follow: In section 2, we present a framework ACTA and their different extensions in litterature. Thereafter, in section 3, we study the influence of pre-commit event. In section 4, we introduce the concept of preferable dependency. Finally, in section 5, we conclude the paper.

FOUNDATIONS

ACTA is not a new model of transactions, but rather a formalism which allows the formal description of properties of the wide transactions. Precisely, using ACTA, one can specify and reason about the effect of transaction on objects and the interactions between the transactions in a particular model.
ACTA is composed of five blocks that we will re-examine along this paper: history, dependencies between transactions, the view of transaction, the conflict set of transaction and delegation.
The effects on the objects deal with the concepts of visibility of conflict and delegation of effects, while the effects on the transactions deal with dependencies between the transactions.

Preliminaries

A transaction must be in one of the following states:

active (t_i): $\Leftrightarrow b_{ti} \in H_{ct} \wedge c_{ti} \notin H_{ct} \wedge a_{ti} \notin H_{ct}$

committed (t_i): $\Leftrightarrow c_{ti} \in H_{ct}$

aborted (t_i): $\Leftrightarrow a_{ti} \in H_{ct}$

Advanced transactions are structured, i.e., they consist of sets of transactions which are interrelated. The following predicates describe the relationship between a transaction and its creator:

Root:={ t_j has no parent}

Parent(t_i, t_j):= {t_i is parent of t_j}

Superior(t_i,t_j):={ Parent(t_i, t_j) $\vee$ ($\exists$ t_K: Superior(t_i,t_K) $\wedge$ Superior(t_K,t_j))}

Child(t_i,t_j):= {t_i is child of t_j}

Descendent (t_i, t_j) :={ Child(t_i, t_j) $\vee$ ($\exists$ t_K: descendent(t_i,t_K) $\wedge$ descendent(t_K,t_j))}

The concept of History is one of fundamental notion which represents the concurrent execution of a set of transactions T, contains all the events invoked by the transaction in T and indicates the (partial) order in which these events occur. The complete history H contains only terminated transactions, the current (incomplete) history is termed as H_{ct}.

Effects Of The Transactions Ones On The Others

The dependencies provide a practical manner to simplify and to reason on the behavior of the concurrent transactions. In fact the dependencies describe the effects of the transactions on other transaction and represent constraints on possible stories. By examining the possible effects of the transactions acting ones on the 'others; it is possible to determine the dependencies which can be developed between them.

We review the fundamental dependencies described in (Chrysanthis and Ramamritham 1994) and extended by (Schwarz et al 1998a)(Schwarz et al 1998 b) (Schwarz et al 1998 c)(Abdouli 2006). Let t_i and t_j be transactions and H be a finite history:

The two first and important dependencies presented in (Chrysanthis and Ramamritham 1994) are Commit dependency and Abort dependency defined following:

Commit Dependency (t_i CD t_j). If t_i and t_j commit, then the commit of tj must precede the commit of t_i. $c_{ti} \in H \Rightarrow (c_{tj} \in H(c_{tj} \rightarrow c_{ti}))$

Abort Dependency (t_i AD t_j). If t_j aborts, t_i has to abort, too: $a_{tj} \in H \Rightarrow a_{ti} \in H$

Schwarz et al (Schwarz et al. 1998a) distinguish between different upwards and downwards abort dependencies, in general structure, e.g. open nested transaction when the child transaction can leave the scope of parent transaction.

Influences Of Abortions On Superiors

Four upwards abort dependencies are defined: vital, weak-vital, weak-non-vital, and non-vital.

Vital transaction: A transaction t_j is vital for another transaction t_i iff t_i is (transitively) abort as well as commit dependent on t_j

Contingency transaction: transaction t_c is a contingency transaction of transaction t_j iff t_c is semantically equivalent to t_j, t_c is force-begin-on-abort depend on t_j, the common parent t_i of t_c and t_j is commit dependent on t_c, and t_i is abort dependent on t_c or there exists a contingency transaction t_m for t_c

A transaction is termed as an alternative transaction to t_j iff t_c is (transitively) a contingency of t_j

Weak-vital transaction: a transaction t_j is weak-vital for another transaction t_i iff t_i is transitively) commit dependent on t_j, begin-before-commit depended on the contingency transaction t_c of t_j in case t_j aborts, an there is an alternative t_m of t_j which is vital to terminate the execution of t_j's alternatives

Weak-non-Vital transaction: A transaction t_j is weak-non-vital for another transaction t_i iff ti is (transitively) commit-on-termination dependent on t_j

Non-vital transaction: Transaction t_j is non-vital for another transaction t_i iff the abortion of t_j has no effects on the termination of t_i

Effects Of Abortions On Child Transaction

Three dependencies are defined to express the influences of abort of transaction on their superior: dependent, weak-dependent and independent.

Dependent transaction: A transaction t_j is dependent on another transaction t_i iff t_j is (transitively) abort dependent on t_i

Weak-Dependent transaction: Transaction t_i is weak-dependent on another transaction t_i iff t_j is (transitively) weak-abort dependent on t_i

Independent transaction: Transaction t_j is independent of another transaction t_i iff the abortion of t_i has no effects on the termination of t_j

Many others transaction described in literature are described in following:

Begin Dependency (t_iBDt_j). Transaction t_i can only be initiated if t_j is already initiated.

Weak-Abort Dependency (t_i WD t_j). If t_i commits and t_j aborts, the commit of t_i precedes the abortion of t_j.

Serial Dependency (t_i SD t_j). Transaction t_i cannot begin executing until t_j either commits or aborts.

Parallel Strict Overlapping dependency: two different transactions t_i and t_j are executed parallel strict overlapping if and only if the begin of t_i preceds the begin of t_j, the begin of t_j precedeces the termination of t_i, and the termination of t_i precedes the termination of t_j

Parallel including dependency: two diffrent transaction t_i and t_j are excuted parallel including if and only if the begin of t_i precedes the begin of t_j but the termination of t_j precedes the termination of ti

Parallel dependency: Two transactions are executed parrallel if and only if they are executed parrel strict overlapping or parrallel including

The last four dependencies are named execution dependency and can be combined with many others dependencies.

Begin-on-Abort Dependency (t_i BAD t_j). Transaction t_i cannot begin its execution until t_j aborts

Begin-on-Commit Dependency (t_i BCD t_j). Transaction t_i cannot beging executing until t_j commits

Additionally to the dependencies stated above, the following dependencies are introduced by (Schwarz et al. 1998a, Abdouli 2006)

Force-Begin-on-Abort Dependency (t_i FBDA t_j). if t_j aborts, t_i has to begin.

Commit-on-Termination Dependency (t_i CTD t_j) Transaction t_i cannot commit until t_j either commits or aborts

Begin-Before-Abort Dependency (t_i BBAD t_j). Transaction t_i cannot abort until transaction t_j starts its execution.

Begin-Before-Commit Dependency (t_i BBCD t_j). Transaction t_i cannot commit until transaction t_j starts its execution

Exclusive dependency (t_i ED t_j) if transaction t_j commits then t_i must aborts

Exclusive Exc(t_i, t_j) Two transactions t_i et t_j are excluded mutually iff each transaction develops a relation of exclusive dependency on the other

Weak-Commit dependency (t_i WCD t_j) Transaction t_i cannot commit until transaction t_j commits (t_j is a vital transaction)

Loan dependency (t_i LD t_j) Transaction t_i cannot begins execution until t_j is in uncertainty phase

Access dependency (t_i ACCD t_j) Transaction t_i cannot lends its data until the transaction t_j terminates (abort or commit)

Authorization dependency (t_i AU ε) A transaction t_i develops a relation of Authorization-dependent on a set of transaction ε, hence transaction t_i can lend its data only to transaction of the type ε

Until now we presented only the binary dependencies but it is proven that these dependences are not enough to express the various relations between the transactions; thus Schwarz et al.(Schwarz et al. 1998 c) define the various ternary dependency and "N-ary" dependencies. In (Schwarz et al. 1998 b) the authors studied the impact of transaction compensation on the dependences.

Effects Of Transaction On Objects

A transaction invokes an operation on an object and modifies its state and its statute which characterize it.

The transaction' effects on objects are characterized by the set of effects which are visible for it, the whole of the conflict operations which it carries out and the whole of the effects that it delegates to other transactions. ACTA allows the capture of these effects by the introduction of two sets: ViewSet and AccessSet and by the concept of delegation.

ViewSet And AccessSet

ACTA allows finer control over the visibility of objects by associating two entities, namely ViewSet and ConflictSet with every transaction. We mean by Visibility the ability of one transaction to see the effects of another transaction on objects while they are executing.

Defintion: the view set contains all the objects potentially accessible to the transaction.

Definition: the conflictSet of a transaction contains those operation in the current history with respect to which the effects of conflicts have to be determined when t invokes an operation.

Definition: AccessSet contains all objetcs already accessed by a transaction.

Delegation

Traditionnally, the committing or aborting of an operation is part of responsability of the invoker. However, in general, the invoker and the one committing the operation may be different. We say that a transaction delegate his responsability to another transaction.

INFLUENCE OF EVENT PRE-COMMIT ON TRANSACTION DEPENDENCY

Pre-commit is significant event beside begin, commit and abort events. It permits to transaction to commit temporarily. We try to explain its importance and its impacts in many transaction models with the following examples of applications.

Applications in mobile networks have particular requirements; so many new transactions models, new algorithms of scheduling and new commit protocol are proposed. Indeed, the commit in these models passes in the majority of these models by two steps: a local commit (pre-commit) followed by global commit (Madria 1998) (Pitoura and Bhargava 1995) (Madria et al. 2002). In the first step, after transaction pre-commits, all result produced by a transaction can be viewed by all transactions executed in mobile unit and fixed host. In the second step, after transaction commits, their results are viewed by all transactions. Hence, we allow a more availability of data and more concurrency between transactions in such applications precisely when disconnections are frequent.

Recent research in real-time transaction systems and database has focused on the idea of utilizing partial result so that transactions meet their deadlines. In others words, transactions reports estimate or approximate results when they cannot complete within their time quotas. Transactions are composed by two types of sub-transactions: optional and required. When required sub-transaction(s) are executed, transactions pre-commit. Optional sub-transactions and their sub-transactions can be cancelled during execution if time does not permit. Optional sub-transactions strive to improve the result being provided to users. Consequently, pre-committed transactions will not abort.

Due to induce of pre-commit event, transactions relationship and behaviour between transactions are modified. So, dependencies described in the preceding section can't express the new relationship between transactions. To fill these deficiencies, we propose many new dependencies for more expressiveness of these relationships.

In former work if two transaction t_i and t_j must commit we can define only one dependency "t_i CD t_j". This means that transaction t_i can't commit until t_j commit. So with such dependency, we can't express the relationships between pre-commit transactions. So we define these three new dependencies for more precision of execution of transaction:

Pre-Commit dependency (t_i PCD t_j). Transaction t_i can't pre-commit until transaction t_j pre-commit. $pc_{ti} \in H \Rightarrow (pc_{tj} \rightarrow pc_{ti})$

Strict-Pre-Commit dependency (t_i SPCD t_j). Transaction t_i can't pre-commit until transaction tj commit. $pc_{ti} \in H \Rightarrow (c_{tj} \rightarrow pc_{ti})$

Weak-Commit dependency (t_i WCD t_j). Transaction t_i can commit if t_j has already pre-commit. $pc_{ti} \in H \Rightarrow (pc_{tj} \rightarrow c_{ti})$

For more precision we can combine the pre-commit dependency or weak-commit dependency with commit dependency.

In same way we can introduce the pre-commit on the exclusion dependency. In the literature exclusion dependency (t_i ED t_j) express that t_i must abort if t_j commit. In our case we can exclude the transaction t_i when t_j pre-commit. So we not forced to wait the commit of t_j.

Pre-Exclusive dependency (t_i PED t_j) if transaction t_j pre-commit then t_i must abort. $pc_{tj} \in H \Rightarrow (b_{ti} \in H \Rightarrow a_{ti} \in H)$

Pre-Exclusive Pre-Exc(t_i, t_j) Two transactions t_i et t_j are excluded mutually iff each transaction develops a relation of pre-exclusive dependency on the other. Pre-Exc(t_i, t_j) : $\Leftrightarrow$ (t_i PED t_j) $\wedge$ (t_j PED t_i)

In the same manner we can define the following dependency:

Weak Serial Dependency (t_i WSD t_j). Transaction t_i cannot begin executing until t_j either pre-commits or aborts: $b_{tj} \in H \Rightarrow ((pc_{tj} \rightarrow b_{ti}) \vee (a_{tj} \rightarrow b_{ti}))$

Begin-on-Pre-Commit Dependency (t_i BPCD t_j). Transaction t_i cannot beging executing until t_j pre-commits: $b_{ti} \in H \Rightarrow (pc_{tj} \rightarrow b_{ti})$

Weak-Commit-on-Termination Dependency (t_i WCTD t_j) Transaction t_i cannot commit until t_j either pre-commits or aborts: $c_{ti} \in H \Rightarrow (pc_{tj} \rightarrow c_{ti}) \vee (a_{tj} \rightarrow c_{ti})$

Pre-Commit-on-Termination Dependency (t_i PCTD t_j) Transaction t_i cannot pre-commit until t_j either commits or aborts: $pc_{ti} \in H \Rightarrow (c_{tj} \rightarrow pc_{ti}) \vee (a_{tj} \rightarrow pc_{ti})$

Weak Pre-Commit-on-Termination Dependency (t_i WPCTD t_j) Transaction t_i cannot pre-commit until t_j either pre-commits or aborts: $pc_{ti} \in H \Rightarrow (c_{tj} \rightarrow pc_{ti}) \vee (a_{tj} \rightarrow pc_{ti})$

Begin-Before-Pre-Commit Dependency (t_i BBPCD t_j). Transaction t_i cannot pre-commit until transaction t_j starts its execution: $pc_{ti} \in H \Rightarrow (b_{tj} \rightarrow pc_{ti})$

Weak Access dependency (t_i ACCD t_j) Transaction t_i cannot lend its data until the transaction t_j pre-commits or aborts.

PREFERABLE AND CONDITIONAL DEPENDENCY

In recent application such as web service, the models transactions used are dynamic i.e. transactions and sub-transactions invoke dynamically different sub-transactions. Subsequently the structure of transaction is not known from the begging. In others words, different executions of the same Web service (transactions) may call different sub web services depended on conditions and parameters (Le Gruenwald and Obermeier 2006).

For that, we define a new alternative-begin dependency which makes it possible to choose which transaction will start according to the condition C.

Alternative-begin dependency: Alter-begin(t_i, t_j, C) t_i and t_j are said alternative, if C is satisfied then t_i begin else t_j begin. $b_{tj} \in H \Rightarrow b_{tj} \notin H$.

Alter-begin dependency is different from the Alter dependency, indeed in Alter-begin we choose from the beginning the transaction that will begin, the second transaction will never start even the first one abort. In alter dependency, if the first transaction aborts than the second transaction begins.

In dynamic systems, such web servers and sensor networks with non uniform access patterns, the workload of RTDBs cannot be precisely predicted and, hence, the RTDBs can become overloaded. As a result, uncontrolled deadline misses may occur during the transient overloads.

To provide reliable service quality and guarantees a set of requirements on the performance of the database imprecise computation techniques have been introduced to allow flexibility in operation and graceful degradation during transient overloads. (Amirijoo et al. 2006, Haubert et al. 2004). (Dogdu 1997, Abdouli 2006) propose new models of transactions such as CAT or adapt existing models of transactions to support imprecise computation and define new protocols of concurrency and scheduling of the transactions.

This will imply modifications on the behavior of the transactions and their interrelationships. None of this work was interested by the types of dependences between these transactions in this new model. For all this, we propose to introduce a concept of condition in dependencies. To show its impact in dependencies, we give some example and we add new dependencies. Finally we derive a new transaction defining by combining the specification of CAT and (m-k) firm transaction.

Definition: a conditional dependency is noted as follow: t_i(dependency [C]) t_j where the condition C is optional part. When condition is mentioned and satisfied the dependency must be respected.

If the condition is omitted then the dependency must be respected all the time.

We review some of dependency described earlier:

Begin dependency: (t_i BD t_j). This dependence means that the transaction t_i can start only if the transaction t_j already started. This dependence can be relaxed to make possible the starting t_i before the starting of t_j when the condition C is not satisfied. (t_i BD[/ C] t_j). If the transaction is omitted ti cannot start only if the tj has alredy strarted.

Commit Dependency: (t_i CD t_j) in real time environment, if c_{tj} not be occure before the deadline of t_i then t_i must abort, so t_j must abort also. This will have as a consequence a loss or wasting of the resources especially when the system is overloaded. From where in a case of overload it is preferable that this dependence is ignored or forgotten if possible to respect the temporal constraints which are more important. This requirement cannot be expressed with a simple dependency for this we define this preferable dependency that permit to expresse such requirement. (t_i CD [/C] t_j).

Abort dependency (t_i AD t_j): this dependency means that if the transaction t_j abort then the transaction t_i must abort. The abort of the two transactions generated a wasting of time and use of resource. In many case, we can tolerate ti to validate in spite of the abort of t_j since one does not have enough time to start again they transactions and the system is overloaded. So we define a transaction that permit to relax the abort dependency when is possible (t_i AD[/C] t_j).

In (Kuo 1997), authors define m-k firm transaction model. A transaction is composed from k sub-

transaction. The first "m" transactions are vital. The number of vital sub-transactions is defined from the beginning by the database manager according to some conditions of environment. The rest (k-m) sub-transactions are non vital, i.e., the abort of some of these transactions will not abort the global transaction. On the contrary others models, in m-k firm transaction model the optional sub-transactions don't begging when system is overload. For express this relation we define a new dependency named preferable dependency.

Preferable dependency: (t_i, pref [/C] t_j) transaction t_j is non vital transaction and t_j begin only if C is true. When the condition is not satisfied then the sub-transaction t_j can't begin. When a C is not mentioned t_j is considered as a non vital transaction.

(t_i, pref [/C] t_j) $\Leftrightarrow$ parent(t_i, t_j) $\wedge$ $b_{tj} \in H \Rightarrow$ C is true $\wedge$ $b_{ti} \in H \wedge$ non-vital(t_i, t_j)

In the same way we will be able to apply these conditions to the following dependencies: Serial dependency; Begin-on-abort dependency; Begin-on-commit dependency; Begin-on-pre-commit dependency; Force-begin-on-Abort dependency; Commit-on-termination dependency; Begin-before Abort dependency; Begin-before commit dependency.

Example:

IN Chain-structured Adaptive (CAT) model the transaction is structured as transaction tree. Chain-structured Adaptive transaction model contains two types of sub-transactions: required sub-transactions and optional sub-transactions. A CAT gives a partial result after the execution of its required part but it cannot commit it before the execution of its optional part. Sub-transaction between two commits points constitutes a subset. A CAT pre-commits after the execution of the first subset and thus it can stop at this point if the expiry is imminent. It is said that a CAT finished correctly so at least it's MES (required sub-transactions subset) is executed and committed. The dependency between root and MES is vital dependency. The sibling sub-transactions develop a serial dependency between them.

The lack of CAT is that optional transactions are aborted after its beginning execution. It is better to define from the beginning sub-transactions that will not admitted in system when it is sure that will no terminate in some condition.

We can define a new transactions model by apply the concept of m-k firm to CAT. Optional subsets develop a preferable dependency with the root i.e. we associate a condition with each optional subset and if condition is not satisfied the subset of transaction will not authorize to begin. E.g. the second subset is not authorised to begin if the uses ratio is more than 55%, the third part is not authorised to begin if the miss ration is 60% and uses ration is more than 80%. Then we not begin this optional subset if uses ratio more than 55%, in others word the probability of success if very low (null), and we will waste resources if this subset of transaction will be executed and will not be committed.

Hence, the MES subset develops vital dependencies with the root transaction. Others siblings' sub-transactions develop a preferable dependency with root transaction.

CONCLUSION

In this paper, we have presented dependencies and its uses for describing transaction models. In particular, we are interested to presented an overview of framework ACTA and its extension in literature. We have described all its components and we have insisted on study transactions dependency. We have proposed many new dependencies by the introduction of the pre-commit event on the dependencies. We have defined the concept of conditional dependency and proposed some new dependencies. In final, we have given an example of use of such dependency to define a new model of transaction.

REFERENCES

Abdouli, M. 2006 "Study of extended transaction model adaptation to real time DBMS". PhD Thesis, Le Havre University , 2006 (in French)

Amirijoo, M., Hansson, J., Son, S. H. 2006. "Specification and Management of QoS in Real-Time Databases Supporting Imprecise Computations". Transactions on computers, vol. 55, no. 3, March.

Chrysanthis, P.K., Ramamritham K. 1994. "Synthesis of Extended transaction Models Using ACTA" ACM Trans, Database Syst., Vol. 19, n°.3, pages 450- 490.

Dogdu E. 1996. "Scheduling Adaptive Transactions in Real-Time Databases", j-LECT-NOTES-COMP-SCI, vol 1134, p130.

Haubert, J., Sadeg, B., Amanton, L. 2004. "(m-k) firm real-time distributed transactions". Proc. of the 16th WIP Euromicro Conference on Real-Time Systems (ECRTS).

Kuo, H-C. 1997. "A Rule-Based Cooperative Transaction Model and Event Processing in Real-Time Active Database Systems". PhD thesis

Le Gruenwald, S., Obermeier, S. 2006. "An Atomic Web-Service Transaction Protocol for Mobile Environments" EDBT-Workshop Privacy Iinformation Management (PIM2006), Munich, Germany.

Madria, S. K. 1998. "A Prewrite Transaction Model" in the proceedings of 3rdInternational Baltic Workshop on Database and Information Systems, Riga, Latvia, April.

Madria, S. K., Mohania M., Bhowmick S. S., Bhargava B. 2002. "Mobile data and transaction management" Information Sciences 141 279-309.

Moss, J. E. 1985. "Nested Transactions: An Approach to Reliable Distributed Computing". Massachusetts Institute of Technology ed., Cambridge, MA, USA.

Pitoura, E., Bhargava, B.K. 1995. "Maintaining Consistency of Data in Mobile Distributed Environments" International Conference on Distributed Computing Systems, p404-413.

Ramamirthan, K. 1997. "Towards the Specification and Analysis of Transactions in Real-Time Active Databases" RTBD'97, Burlington Vermont, USA, p 327-348.

Schwarz, K., Turker, C., Saake, G. 1998a. "Analyzing and formalizing dependencies in generalized transaction structures". In proc. of Int. workshop on Issues and Applications of database technology, 1998, Germany.

Schwarz, K., Turker, C., Saake, G.1998b "Transitive dependencies in transaction closures". Database Engineering and Applications Symposium, July 8-10, 1998 Cardiff, Wales, UK.

Schwarz, K., Turker, C., Saake, G. 1998c "Extending transaction closures by N-ary Termination dependencies" Symposium on Advances in Databases and Information System (Adibis'98), Semptember 8-11, 1998, Poland.

MODELLING LANGUAGES AND TOOLS

TOWARDS A COMPONENT BASED CONCEPTUAL MODELING LANGUAGE FOR DISCRETE EVENT SIMULATION

Deniz Cetinkaya
Alexander Verbraeck
Mamadou Seck
Systems Engineering Group, Faculty of Technology, Policy and Management
Delft University of Technology
Jaffalaan, 5, 2628BX, Delft, THE NETHERLANDS
email: d.cetinkaya@tudelft.nl, a.verbraeck@tudelft.nl, m.d.seck@tudelft.nl

KEYWORDS
Conceptual modeling, conceptual modeling language, discrete event simulation, hierarchical modeling

ABSTRACT

Recent studies state the importance of conceptual modeling in simulation life cycles. Proper development of a conceptual model is critical for expressing the context, elements, relationships, limitations and purpose of the simulation study. Surprisingly there are many simulation projects that have no explicit conceptual model, a poorly or only partially developed conceptual model, or incomplete documentation of the simulation conceptual model. The reason for the deficiency in conceptual modeling stage is that there does not exist a well defined simulation conceptual modeling method. In this paper, a brief overview of the conceptual modeling techniques used in simulation field is provided and the need for a unified simulation conceptual modeling method is stated. Then, a conceptual modeling approach for discrete event simulation is proposed and compared to other modeling techniques.

INTRODUCTION

In general terms, each simulation study has a problem definition, conceptualization (conceptual modeling), model building (simulation model construction), and experimentation stages. Conceptual modeling is probably the most difficult aspect of a simulation study and recent studies state the importance of conceptual modeling in simulation life cycles (Pace 2000, Yilmaz and Oren 2006, Robinson 2006; 2008). During the simulation conceptual modeling stage, a modeler makes an abstraction of the system and prepares the conceptual model for the simulation study. A simulation conceptual model is a simplified representation of the real system without reference to the implementation details. It generally describes the elements, relationships, boundaries and objectives of a simulation study.

Conceptual modeling not only requires that the modeler develop an appropriate model, but that all parties involved in a simulation study understand and agree to that model. As such, it is important that the conceptual model is represented and communicated in a manner that is understandable to all. A range of modeling methods have been used for representing simulation conceptual models, such as event graphs, activity diagrams, IDEF diagrams, process flow diagrams, Petri nets, etc. (Robinson 2006). Many of the techniques present an abstract way of thinking which is not natural and so it is difficult to properly model the real system in the required level of detail. For example, a flow diagram provide an overview of the system and do not have much detail. Petri nets are well defined and they represent a directed graph of nodes and arcs. However, there is not an elegant way of representing hierarchies graphically.

Moreover, conceptual models are often not reused explicitly in the further steps of the simulation process, as formal model transformation methods are not available to guarantee model continuity (Olive 2007). This means that, based on exactly the same conceptual model, different simulation modelers will most likely create different simulation models. This puts an excessively high share of simulation project success responsibility in the hands of the code writer. This situation would have been mitigated if stakeholders were involved in the design of the conceptual models, and if the latter were reused explicitly in the further stages of the process.

Therefore, we can conclude that there is a big semantic gap between the conceptual modeling stage and the simulation model construction stage. Therefore, we would like to pay attention to the deficiency in conceptual modeling stage and the lack of a commonly accepted standardized conceptual modeling method and language in Modeling and Simulation (M&S). In short, the existing modeling methodologies require some development in the state of the art of conceptual modeling and simulation model construction stages.

In this paper, firstly a brief overview of the conceptual modeling techniques used in simulation field is provided. Then, two useful modeling approaches, namely hierarchical modeling and component based modeling are dis-

cussed. After that, a conceptual modeling approach for discrete event simulation is proposed. Finally, conclusions are drawn and future work is outlined.

CONCEPTUAL MODELING METHODS USED IN M&S

Simulation conceptual modeling generally benefits from general purpose diagramming techniques, which are not adequate for meeting the needs of simulation projects. Despite the fact that conceptual modeling is an important step in a simulation study, there is not a common simulation conceptual modeling language. Thus, in many cases conceptualization deeply depends on the skill and experience of individual modelers. This section provides a brief overview of the conceptual modeling methods used in M&S.

In order to provide a better understanding, after giving a brief introduction we will give a sample model of a single server queue for each method. Simulation of a single server queuing system is a common example of discrete event simulation such as an information desk at an airport or a hotel, a pharmacy, a barber shop, or a ticket office. This example was chosen because of its simplicity enables an easier comparison of the methods.

For example, consider a service facility with a single server for which we would like to estimate the average delay in the queue for arriving customers. We define the following state variables: status of the server (idle or busy), number of customers waiting to be served (if any), the arrival time of each customer waiting in the queue. We define three types of events: arrival, service and departure. Delay in the queue means the length of time from the arrival of a customer at the information desk queue until the instant he/she begins to be served.

Event Graphs

Event graphs provide a representation for discrete event simulation (Schruben 1983). An event graph partitions the model into events and relationships between events. The events are represented by vertices (nodes) in the graph and relationships between events are represented as directed edges (arcs) between event vertices. Figure 1 shows an event graph for the sample problem.

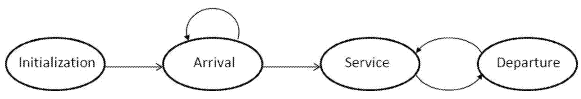


Figure 1: An event graph for a single server queue (Seila et al. 2003)

Activity Cycle Diagrams

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. UML activity diagrams can be used to describe step-by-step workflows of components in a system. An activity diagram mostly consists of, activities (rounded rectangles), decisions(diamonds) and flows(arrows). Bars represent the start (split) or end (join) of concurrent activities. A black circle represents the start (initial state) of the workflow and an encircled black circle represents the end (final state). Flows(arrows) run from the start towards the end and represent the order in which activities happen. Activity diagrams can be regarded as a form of flowchart. Figure 2 shows an activity cycle diagram for the sample problem.

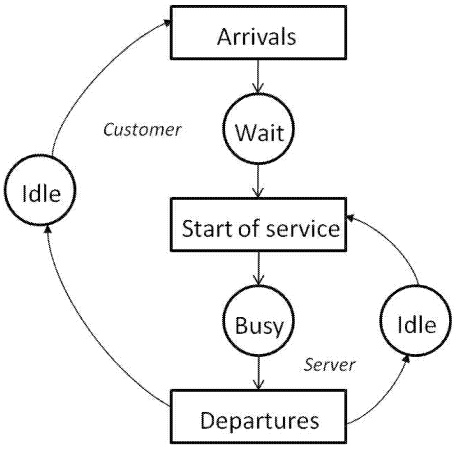


Figure 2: An activity diagram for a single server queue (Seila et al. 2003)

IDEF Diagrams

IDEF (Integration DEFinition) is a family of modeling languages in the field of systems and software engineering. They cover a wide range of modeling methods, yet the most-widely recognized and used one is IDEF0. IDEF0 (Integration Definition for Function Modeling) is a function modeling methodology for describing organizations or systems. An IDEF0 model consists of functions, data and objects. Functions are represented by boxes. Data or objects that interrelate those functions are represented by arrows). Figure 3 shows a simple IDEF0 diagram for the sample problem.

Petri Nets

Petri nets are bipartite graphs and provide a mathematically rigorous modeling framework. They serve as a ready simulation model, as well as a conceptual model.

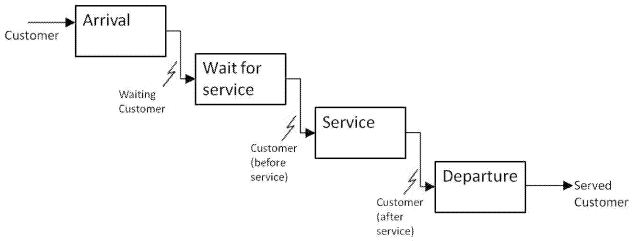


Figure 3: An IDEF0 diagram for a single server queue

However, their analysis is intractable for large models. Petri nets consist of places, transitions, and directed arcs. Arcs run from a place to a transition or a transition to a place, never between places or between transitions.The places from which an arc runs to a transition are called the input places of the transition; the places to which arcs run from a transition are called the output places of the transition. Places may contain a number of tokens. A transition of a Petri net model is fired whenever there is a token at the start of all its input arcs. Figure 4 shows a petri net model for the sample problem.

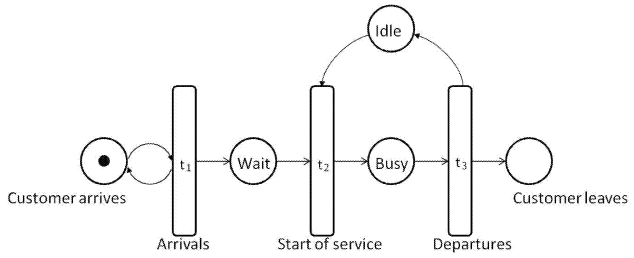


Figure 4: A petri net model for a single server queue

There are various types of Petri nets, such as timed Petri nets, stochastic Petri nets, and colored Petri nets. The use of stochastic Petri nets has become particularly important in the modeling of discrete event systems. Timed Petri nets are the particular types of Petri nets that associate the time and time delays.

HIERARCHICAL COMPONENT BASED MODELING

As modelers build more complex and complicated models for large systems, it becomes hard to design, develop, manage and maintain the simulation models. The monolithic approach for developing models becomes too cumbersome in large simulation projects. Besides, when each simulation model is designed from scratch, the lack of reuse makes simulation a time consuming and expensive task (Oses et al. 2004).

Applying different software engineering approaches into the simulation field can help managing larger models, such as thinking at various levels of abstraction or component based development. In this section two modeling approaches, which have been applied in the simulation field and provided valuable contributions, are discussed. These are hierarchical modeling and component based modeling.

Hierarchical modeling (also known as multi-level modeling) provides a way to represent a system in a hierarchical structure to deal with large scale or complex models in a thorough manner (Simon 1962). Hierarchical modeling allows modeling with more manageable sub-parts at different levels of detail. The ability to move among the different levels of a model hierarchy greatly increases the manageability and understandability of large models (Daum and Sargent 1999). Hierarchical modeling can provide for a more natural way of modeling and help to focus on different degrees of detail when using a model.

Hierarchical models are generally developed in two different ways, that are top-down and bottom-up strategies. In both cases, hierarchical models mostly represent a tree-like structure. In the top-down approach, a system is broken down into subsystems and this is called as decomposition. During the top-down modeling process, modelers specify the main parts and relationships of the system without inner details first and then they fill in the lower levels. In the bottom-up approach, subsystems are coupled together to form a larger system and this is called as composition. During the bottom-up modeling process, modelers first think of the lowest level, i.e. smallest parts or building blocks of the system and then they use these previously constructed building blocks to compose larger models and systems. Simulation models can be developed by employing either a top-down decomposition approach or a bottom-up composition approach.

In the component based approach software systems are built by assembling components already developed and prepared for integration. Component based modeling and simulation is an interesting research area that many researchers studied in the last decade (Buss 2000, Himmelspach and Uhrmacher 2004, Sarjoughian and Elamvazhuthi 2009, Verbraeck and Valentin 2008). Component based simulation relies on having pre-built, validated simulation model components that can be coupled to form a composed model that represents a system. A simulation model component is expected to be a self-contained, interoperable, reusable and replaceable unit, providing useful services or functionality to its environment through properly defined interfaces (Verbraeck and Dahanayake 2002). Component based approach promises to have many benefits over a monolithic approach such as reuse of interoperable components and rapid development (Verbraeck and Valentin 2008).

The development process for component based systems consists of two major stages: component development and component composition (Oses et al. 2004). These

stages are usually carried out by different parties, like domain experts and software engineers. When a component library is available, a developer can build a system in a bottom-up fashion, by combining components into larger components, where an assembly of the highest level components is considered to be the system. In component based approaches, overall software quality increases due to components are thoroughly tested first and reviewed during reuse (Sommerville 2007).
The component based approach has originally a bottom-up way of assembling components, which means that it can be applied together with the hierarchical modeling approach. Simulation model components can be assembled in many ways into a hierarchy. New components can be built from scratch in each layer or reused if they already exist in pre-defined and verified component libraries, so it is not necessary to always create larger components from smaller components.
Applying a unified hierarchical component based modeling approach looks like an encouraging way in the simulation field. During the the conceptual modeling stage, by applying a top-down hierarchical modeling approach, a modeler can first partition the system into the relevant subsystems and define the relationships between them, without delving yet into their inner details. For example, to represent an airport system, one would identify such subsystems as gates, security check points, information desk, check-in desks and so forth. At the simulation model construction stage, by applying a bottom-up component based approach, basic available primitives and building blocks can be composed to provide the desired functionality of the identified subsystems and the simulation model.
However, there is a big semantic gap between the conceptual modeling and the simulation model construction stages. For example, to be able to reuse the existing components, one should know that what is already available. This means that, there must be a way to express how the components relate to the subtrees in the conceptual model. Besides, good classification and documentation is essential for the successful reuse of simulation model components. We believe that in order to bridge this gap, we need a common simulation conceptual modeling language and a model transformation method between the conceptual model and the simulation model. A higher level representation on top of the rigid simulation model implementation is expected to make the simulation model development process faster.

A CONCEPTUAL MODELING APPROACH FOR DISCRETE EVENT SIMULATION

In this section, a hierarchical component based conceptual modeling approach is suggested. The proposed conceptual modeling method will basically define a system with its components, relations, and objectives based on the following definition of a system. A system is a set of interrelated components working together toward some common objective or purpose (Kossiakoff and Sweet 2003, Blanchard and Fabrycky 2006).
We define two types of nodes, that are components and entities. Each component can have four different types of variables: input variables, output variables, local variables and parameters. Components can have various properties such as descriptors and rules. Descriptors define the meta information such as name, version, author, bugs, keywords, etc. They can be used in cataloging and searching components. Rules are constraints that can be defined about components. They can be used to express the boundaries of the system. Besides, we define a component type for each component which is used for classification purposes. We only allow type inheritance in our method and use component type information to classify the components. Type inheritance only provides a limited support for component structure.
Every component has an objective, which is defined by its behavior. At the conceptual modeling level, we provide a way to define the pseudo algorithm for the behavior of a component. This will be used to support model transformation and not obligatory.
Entities are specialized components, having both variables and properties. The only difference between a component and an entity is that entities do not have an internally defined behavior. For example, entity components can be used to represent system resources. A graphical representation for components and entities is shown in Figure 5.

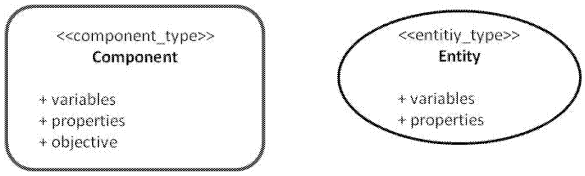


Figure 5: Visual representation for components and entities

In order to define the state of the system, we use the definition of Law and Kelton (1991): the state of a system is the collection of variables necessary to describe a system at a particular time. Hence, we use the local variables to refer to the state and state change is possible when the local variables are updated. Output of a component is available when the output variables are updated.
Relations define how components and entities relate to each other. Six basic relations are suggested in our method and a textual representation for them is listed in Table 1.
Due to hierarchical modeling is applied, composition and decomposition capability is especially handled. Besides, composition and aggregation are differentiated clearly. The hierarchies are represented with 'HAS A'

Relation
Fixed Composition CompA **HAS** CompB
Temporary Composition CompA **GOES** CompB
Logical link CompA **ISLINKED** CompB
Physical link CompA **ISJOINED** CompB
'Send-To' relation CompA **SENDS** EntityC **TO** CompB
'Send-To-Via' relation CompA **SENDS** EntityC **TO** CompB **VIA** CompD

Table 1: Textual representation of basic relations

relation and called as fixed composition. Since a unified approach is performed, composition refers to both composition and decomposition capability. Aggregation refers to a temporary whole-part relationship during the execution of the simulation model. This type of relation is called as temporary composition and represented as with 'GOES TO' relation.
Logical links and physical links are distinguished as well. Association relations or any other logical relationships can be expressed with logical links. 'Send-To' and 'Send-To-Via' relations are provided for transferring data between components. When necessary and appropriate cardinality information can be defined for the relations, such as: 1..*, * or 0..*, n, 0..1, 1, ...etc. A graphical representation for the suggested relations is illustrated in Table 2.
In order to define the objective of the components we suggest four main behavioral modeling primitives, which are if condition, while loop, switch case and assignment. Then we define an expression as a combination of these primitives. A possible textual notation is given below:

- $IF < condition > THEN < expression > ELSE < expression >$
- $SWITCH\{CASE < case >< expression >\}$

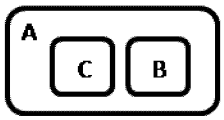
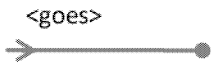
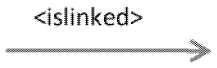
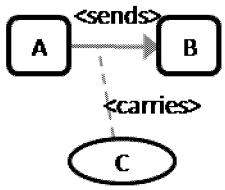
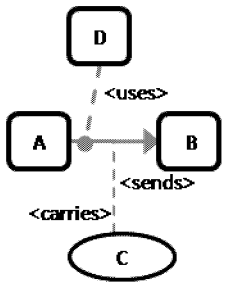
Relation	Representation
Fixed Composition	
Temporary Composition	
Logical link	
Physical link	
'Send-To' relation	
'Send-To-Via' relation	

Table 2: Basic relations of the proposed conceptual modeling method

- $WHILE < condition > DO < expression >$
- $Assignment :< variable >= value$
- $Expression :$ $\{IF\_Cond, While\_Loop, Switch, Assignment\}$

A sample component diagram of a single server queue is demonstrated in Figure 6. The model defines the following steps:

- *Customer* arrives(GOES) to the *Waiting Queue* of the *Service Desk*
- *Waiting Queue* ISJOINED to the *Service Process*
- *Waiting Queue* is a queue component, thus when available *Customer* is sent to *Service Process*
- When *Service Process* is finished, *Customer* leaves
- *Waiting Queue* calculates the delay time for each *Customer*

- *Service Process* calculates the service time for each *Customer*

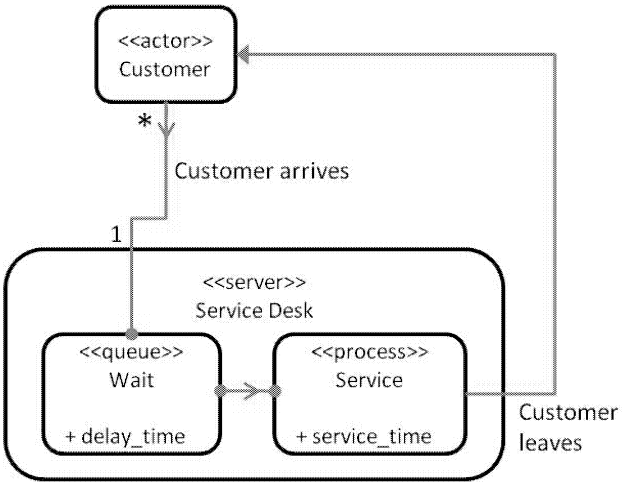


Figure 6: A sample conceptual model with the proposed method

COMPARISON WITH THE EXISTING TECHNIQUES

In order to compare the proposed method in its current status with the other techniques, a number of requirements for an effective conceptual modeling language is represented below:

1. It should represent the system structure (elements and relations) clearly.
2. It should represent the purpose of the simulation study (objectives and boundaries).
3. It should be abstract from technical or organizational details (Ribbert et al. 2004).
4. It should support classification and inheritance.
5. It should support hierarchical modeling to develop manageable and understandable models.
6. It should be formal enough to avoid misinterpretations. Besides, it should be theoretically possible to map the conceptual model to a formal specification to support model transformations (Ribbert et al. 2004).
7. It should be easy to learn and use (Ribbert et al. 2004).

Most of the conceptual modeling languages and modeling techniques provide the first four requirements. However, the main problem in simulation conceptual modeling is hierarchical modeling and model composability (Kasputis and Ng 2000). Recent studies state that model composability is troublesome in simulation and the existing methodologies require additional effort to facilitate it (Röhl and Uhrmacher 2006, Yilmaz and Oren 2006). Indeed, combining multi-level abstraction and composition with inheritance and aggregation is not easy, neither in theory nor in practice. Figure 7 shows the three dimensions of hierarchical simulation conceptual modeling. Object oriented modeling methods provide few mechanisms to describe components. Simply adopting the object oriented concepts is not adequate for expressing the hierarchies in simulation models. Thus, when a modeler wants to add different layers into his/her models, object oriented conceptual modeling techniques become insufficient.

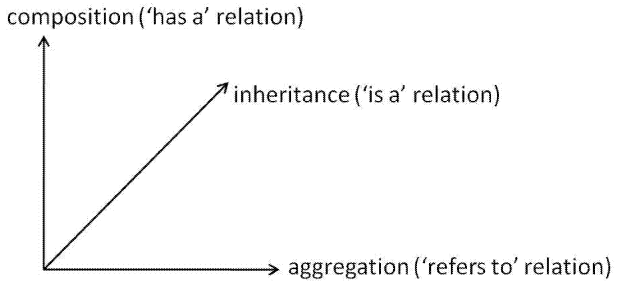


Figure 7: Three dimensions in hierarchical simulation conceptual modeling

Although many modeling techniques have well defined syntax or semantics, a clear metamodel and a rigorous formalism are lacking in many cases. Besides, they do not provide formal model transformation methods to guarantee model continuity. Only Petri net models serve as a ready simulation model. However, they are far from being practical and easy to use. In many cases, simulation modelers need to be experienced and trained.
The proposed method represents the system structure with components and relations. Components have objectives and rules that define the purpose of the simulation study. Hence, it satisfies requirement 1 and 2. The method satisfies requirement 3 partially, since it allows defining pseudo algorithms for objectives. It satisfies requirement 4 partially as well, due to it only allows type inheritance. Hierarchical modeling is supported via the component structure and we claim that the method is easy to use. Requirement 6 is a future work at the moment, a metamodel and a model transformation method will be defined for the proposed method. After that, a more detailed comparison will be performed.

CONCLUSION AND FUTURE WORK

Although, an effective and consistent conceptual model is critical for expressing the purpose of the simulation study, many simulation projects have no deliberate conceptual modeling stage. Moreover, formal model transformation methods are not available to help the simu-

lation model developers while moving from the conceptual model to simulation model. As a result, simulation models generally do not have a higher level representation on top of the rigid simulation model implementation and so they are not understandable to others. We think that the deficiency in simulation conceptual modeling is caused by the lack of a well-defined conceptual modeling method and language in M&S. However, this subject has not been adequately studied yet in simulation field. This work aims at improving the conceptual modeling stage and increasing the reuse of simulation model components in modeling and simulation. We suggest a conceptual modeling approach for discrete event simulation and lead to new insights about conceptual modeling. Reuse of simulation model components will help the modelers to construct their simulation models faster, better and more reliable. Additionally, a common conceptual modeling method will provide a better understanding for conceptual models.
As a future work, we will define the suggested simulation conceptual modeling method formally and propose a metamodel for the simulation conceptual modeling language. After that, a unified modeling and simulation methodology that ensures model continuity will be proposed by the use of the gained insights about conceptual modeling.

AUTHOR BIOGRAPHIES

DENIZ CETINKAYA is a Ph.D. student at Delft University of Technology. She is in the Systems Engineering Group of the Faculty of Technology, Policy and Management. She received her M.Sc. in Computer Engineering from the Middle East Technical University, Turkey in 2005. She received her B.Sc. with honors in Computer Engineering from the Hacettepe University, Turkey in 2002. Her research focuses on component based modeling and simulation. Her e-mail address is <d.cetinkaya@tudelft.nl>.

ALEXANDER VERBRAECK is a full professor in Systems and Simulation in the Systems Engineering Group of the Faculty of Technology, Policy and Management of Delft University of Technology, and a part-time full professor in supply chain management at the R.H. Smith School of Business of the University of Maryland. He is a specialist in discrete event simulation for real-time control of complex transportation systems and for modeling business systems. His e-mail address is <a.verbraeck@tudelft.nl>.

MAMADOU D. SECK is an assistant professor in the Systems Engineering Group of the Faculty of Technology, Policy, and Management of Delft University of Technology. He received his Ph.D. degree from the Paul Cezanne University of Marseille and his M.Sc. and M.Eng. degrees from Polytech Marseille, France. His research interests include modeling and simulation formalisms, dynamic data driven simulation, human behavior representation and social simulation, and agent directed simulation. His e-mail address is <m.d.seck@tudelft.nl>.

REFERENCES

Blanchard B.S. and Fabrycky W.J., 2006. *Systems engineering and analysis.* Pearson Prentice Hall, NJ, 4^{th} ed.

Buss A., 2000. *Component-based simulation modeling.* In J. Joines; R. Barton; K. Kang; and P. Fishwick (Eds.), *Proceedings of the 32^{nd} Winter Simulation Conference (WSC '00).* IEEE Computer Society, San Diego, CA, USA. ISBN 0-7803-6582-8, 964–971.

Daum T. and Sargent R.G., 1999. *Scaling, hierarchical modeling, and reuse in an object-oriented modeling and simulation system.* In *Proceedings of the 31^{st} Winter Simulation Conference (WSC '99).* ACM, Phoenix, Arizona, USA. ISBN 0-7803-5780-9, 1470–1477. doi:http://doi.acm.org/10.1145/324898.325304.

Himmelspach J. and Uhrmacher A., 2004. *A component-based simulation layer for JAMES.* In *Proceedings of the 18^{th} Workshop on Parallel and Distributed Simulation (PADS '04).* IEEE, Piscataway NJ, 115–122.

Kasputis S. and Ng H.C., 2000. *Composable simulations.* In J.A. Joines; R.R. Barton; K. Kang; and P.A. Fishwick (Eds.), *Proceedings of the 32^{nd} Winter Simulation Conference (WSC '00).*

Kossiakoff A. and Sweet W.N., 2003. *Systems enginnering: principles and practice.* Wiley Series.

Law A.M. and Kelton W.D., 1991. *Simulation modeling and analysis.* McGraw-Hili, Inc., 2^{nd} ed.

Olive A., 2007. *Conceptual modeling of information systems.* Springer.

Oses N.; Pidd M.; and Brooks R.J., 2004. *Critical issues in the development of component-based discrete simulation. Simulation Modelling Practice and Theory*, Volume 12, no. 7-8, 495–514.

Pace D.K., 2000. *Ideas about simulation conceptual model development. Johns Hopkins APL Technical Digest*, Volume 21, no. 3, 327–336.

Ribbert M.; Niehaves B.; Dreiling A.; and Holten R., 2004. *An Epistemological foundation of Conceptual Modeling.* In *In Proceedings of the 12^{th} European Conference on Information Systems.* Turku, Finland, 1557–1568.

Robinson S., 2006. *Conceptual modeling for simulation: issues and research requirements.* In L.F. Perrone; B. Lawson; J. Liu; and F.P. Wieland (Eds.), *Proceedings of the 38th Winter Simulation Conference (WSC '06).* WSC, Monterey, California, USA. ISBN 1-4244-0501-7, 792–800.

Robinson S., 2008. *Conceptual modelling for simulation Part I: definition and requirements. Journal of the Operational Research Society*, 59, 278–290.

Röhl M. and Uhrmacher A.M., 2006. *Composing simulations from XML-specified model components.* In L.F. Perrone; F.P. Wieland; J. Liu; B.G. Lawson; D.M. Nicol; and R.M. Fujimoto (Eds.), *Proceedings of the 38th Winter Simulation Conference (WSC '06).* IEEE, 1083–1090.

Sarjoughian H.S. and Elamvazhuthi V., 2009. *CoSMoS: a visual environment for component-based modeling, experimental design, and simulation.* In O. Dalle; G.A. Wainer; L.F. Perrone; and G. Stea (Eds.), *Proceedings of the 2nd International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools '09).* ICST, Rome, Italy.

Schruben L., 1983. *Simulation modeling with event graphs. Communications of the ACM*, 26, no. 11, 957–963.

Seila A.; Ceric V.; and Tadikamalla P., 2003. *Applied Simulation Modeling.* Brooks/Cole Publishing, Thomson Learning Inc.

Simon H.A., 1962. *The architecture of complexity.* In *Proceedings of the American Philosophical Society.* 467482.

Sommerville I., 2007. *Software Engineering.* Addison-Wesley, 8th ed.

Verbraeck A. and Dahanayake A.N.W., 2002. *Building blocks for effective telematics application development and evaluation.* Delft University of Technology.

Verbraeck A. and Valentin E., 2008. *Design guidelines for simulation building blocks.* In S.J. Mason; R.R. Hill; L. Mönch; O. Rose; T. Jefferson; and J.W. Fowler (Eds.), *Proceedings of the 40th Winter Simulation Conference (WSC '08).* WSC, InterContinental Hotel, Miami, Florida, USA. ISBN 978-1-4244-2708-6, 923–932.

Yilmaz L. and Oren T.I., 2006. *Prospective issues in simulation model composability: basic concepts to advance theory, methodology, and technology. The MSIAC's M&S Journal Online*, Volume 2, 1–7.

INTERACTIVE SIMULATION IN MODELICA: A PROPOSAL

Alfonso Urquia, Carla Martin-Villalba and Sebastian Dormido
Departamento de Informática y Automática, UNED
Juan del Rosal 16, 28040, Madrid, Spain
Email: {aurquia,carla,sdormido}@dia.uned.es

KEYWORDS
Modelica, Interactive simulation, Virtual laboratories, Object-oriented modeling, Hybrid models

ABSTRACT

The distinctive characteristic of interactive simulations is that external objects are allowed to change at event instants the value of certain model quantities, named interactive quantities. A formal description of these events, named interactive events, is introduced. Criteria to decide whether a set of parameters and time-dependent variables of the physical model can be selected as interactive quantities are proposed. Different procedures to describe the interactive events in Modelica are discussed. Finally, an extension to the Modelica language, intended to facilitate the description of interactive events, is proposed.

INTRODUCTION

Modelica (Modelica Association 2010) is a freely available, object-oriented modelling language that supports the physical modelling paradigm (Åström et al. 1998). Models are mathematically described by differential and algebraic equations (DAE), algorithms and discrete equations. Modelica supports a declarative (i.e., non-causal) description of the model. Therefore, the use of Modelica reduces considerably the modelling effort and permits better reuse of the models. A number of free and commercial Modelica libraries in different domains are available (Modelica Association 2010).

The distinctive characteristic of interactive simulations is that external objects are allowed to change at event instants the value of certain model quantities, named *interactive quantities*. These events are named *interactive events*. The time instants when these changes are triggered are determined by the external objects. An arbitrary finite number of interactive events can be triggered during the simulation run. Depending on the application, the external objects can be people (e.g., in virtual-labs), hardware (e.g., in hardware-in-the-loop simulations), another model simulations (e.g., in distributed real-time simulation), etc. A bi-directional flow of information between the interactive model and the external objects is established during the simulation. The model sends to the external objects the actual value of selected model quantities. The external objects send to the model the information required to execute the *interactive events*.

Modelica has not been specifically designed to facilitate interactive simulation. However, the language provides features that allow to describe interactive models. Calls to C and Fortran functions can be encapsulated within Modelica functions, which facilitates the communication between the model and external objects, using interface programs written in C and Fortran. Modelica provides the *when* clause and the *reinit* function to describe instantaneous changes in the value of the state variables. The *when* clause and the *pre* function can be used to describe discrete-time variables. Also, Modelica allows the user to select the model state variables.

A systematic methodology for transforming any Modelica model into a description suitable for interactive simulation was proposed in (Martin-Villalba 2007), and was successfully applied to the development of virtual-labs for control education and industrial applications (Martin-Villalba 2007, Martin-Villalba et al. 2008; 2010). The original model of the system is called the *physical model* and its reformulation for interactive simulation is called the *interactive model*. Essentially, the methodology consists of modifying the physical model so that all the interactive quantities are formulated as state variables in the interactive model.

This methodology is revisited in this paper. Firstly, a formal description of the interactive events is introduced. Secondly, criteria to decide whether a set of parameters and time-dependent variables of the physical model can be selected as interactive quantities are proposed. Thirdly, a two-tank model is used to illustrate the selection of the interactive quantities, the definition of the interactive events and the description of the interactive model. Next, different procedures to describe the interactive events in Modelica are discussed. Finally, an extension to the Modelica language is proposed and illustrated using the two-tank model. The discussed models and methods have been developed using Dymola.

PHYSICAL MODEL QUANTITIES

The physical model quantities can be classified into time-dependent variables and parameters.

- *Time-dependent variables* are continuous-time and discrete-time quantities calculated from the model.
- *Parameters* are time-independent quantities. They are assigned initial values and these values remain constant during the simulation run.

The following notation is introduced. The physical model parameters are represented by $\mathbf{p}$ and the time-dependent variables by $\mathbf{x}$. The complete set of physical model quantities is $\mathbf{v} = \{\mathbf{p}, \mathbf{x}\}$.

INTERACTIVE MODEL

Parameters of the physical model can be selected as interactive quantities. These parameters are defined in the interactive model as time-dependent variables, whose values change at interactive event actions and remain constant between consecutive changes. Interactive parameters commonly represent system properties and boundary conditions to the interactive model.

Time-dependent variables of the physical model can also be selected as interactive quantities. In this case, the interactive model needs to combine the dynamic behaviour described in the physical model and the interactive events, in which the interactive quantity value can change abruptly.

INTERACTIVE EVENTS

The definition of an interactive event, A, is composed of *condition* and *action*, i.e. $A = \{c, \mathbf{v}^*\}$.

- The condition, c, is a Boolean variable. The interactive event is triggered when c changes from *false* to *true*.
- The action consists in changing the values of certain model quantities to new ones, which are provided by the external objects or have been pre-defined in the interactive model. $\mathbf{v}^* = \{\mathbf{p}^*, \mathbf{x}^*\}$ represents the interactive quantities modified in the action, where $\mathbf{p}^* \subseteq \mathbf{p}$ and $\mathbf{x}^* \subseteq \mathbf{x}$. $\mathbf{p}^*$ are named *interactive parameters* and $\mathbf{x}^*$ *interactive variables*.

An interactive model can define several interactive events, each one with its own condition and action. The set of interactive events of an interactive model can be represented as $\mathbf{A} = \{A_1, A_2, \ldots\}$, where $A_i = \{c_i, \mathbf{v}_\mathbf{i}^*\}$. *The interactive model must guarantee that only one interactive event is triggered at a time*, i.e., that two conditions do not become true at the same time.

The selection of the interactive quantities modified in an interactive event depends on the particular application. However, some restrictions are discussed below.

Statement 1 *The physical model can contain equations relating parameters, which are used to calculate some parameters from others. In this case, only the value of certain parameters can be set independently. The parameters $\mathbf{p}^* \subseteq \mathbf{p}$ can be selected as interactive quantities if and only if the value of each parameter in $\mathbf{p}^*$ can be set independently of the value of the other parameters in $\mathbf{p}^*$.*

Statement 2 *A time-dependent variable $x \subseteq \mathbf{x}$ can be an interactive quantity if and only if there is at least one selection of the state variables $\mathbf{e}$ that includes this variable (i.e, $x \subseteq \mathbf{e}$). $\mathbf{e} \subseteq \mathbf{x}$ represents a possible selection of state variables of the physical model. If the variable x can not be selected as state variable of the physical model, then this variable can not be an interactive quantity.*

Statement 3 *The time-dependent variables $\mathbf{x}^* \subseteq \mathbf{x}$ can be interactive quantities modified in the same interactive action if and only if there is at least a selection of the state variables, $\mathbf{e}$, that includes to all the variables in $\mathbf{x}^*$, i.e., $\mathbf{x}^* \subseteq \mathbf{e}$.*

RELEVANCE OF THE STATE SELECTION

In general, different choices of the state variables are possible in the physical model. As the state variable values that are not explicitly modified in the interactive event action remain unchanged at the event instant, the result of the interactive actions depends on the state variable selection.

The effect of changes in the value of the interactive quantities depends on the state variable selection. Therefore, the definition of an interactive event needs to include under what state variable selection the interactive action have to be performed. The set of interactive events of an interactive model can be represented $\mathbf{A} = \{A_1, A_2, \ldots\}$, where $A_i = \{c_i, \mathbf{v}_\mathbf{i}^*, \mathbf{e_i}\}$ is an interactive event, c_i is the trigger condition, $\mathbf{v}_\mathbf{i}^* = \{\mathbf{p}_\mathbf{i}^*, \mathbf{x}_\mathbf{i}^*\}$ are the interactive quantities modified in this interactive event and $\mathbf{e_i}$ is the state variable selection that have to be used for solving the re-start problem of this interactive event. According to Statement 3, it has to be satisfied: $\mathbf{x}_\mathbf{i}^* \subseteq \mathbf{e_i}$.

TWO-TANK MODEL

The two-tank model described in this section is used to illustrate the selection of the interactive quantities, the definition of the interactive events and the description of the interactive model.

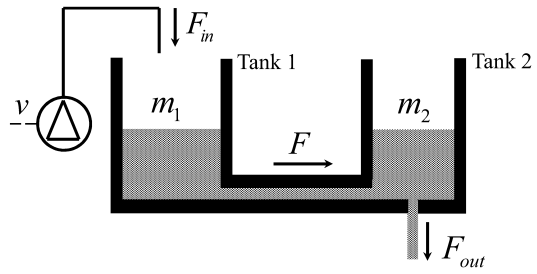


Figure 1: Two-tank system.

Table 1: Two-tank model quantities.

Symbol	Quantity
a	Outlet hole section
F, F_{in}, F_{out}	Mass flows
g	Acceleration due to gravity
h_i	Liquid level in tank i
k_v	Pump parameter
k_0	Proportional factor
m_i, m	Liquid mass in tank i, total mass
p_i	Pressure at bottom of tank i
r_i	Radius of tank i cross section
S_i	Cross section of tank i
v	Voltage applied to the pump
ρ	Liquid density

Physical model

A schematic representation of the two-tank system is shown in Figure 1. The physical model is Eqs. (1)–(11). The meaning of the model quantities is explained in Table 1. A pump introduces liquid into tank 1. The input flow is proportional to the sum of the voltage applied to the pump and a function of time. Liquid exits tank 2 through a hole placed at the tank bottom. The pressure at the bottom of tank 1 is equal to the pressure at the bottom of tank 2. The model quantities are classified into parameters and time-dependent variables: $\mathbf{p} = \{a, g, k_v, k_0, r_1, r_2, S_1, S_2, \rho, v\}$ and $\mathbf{x} = \{F, F_{out}, F_{in}, m_1, m_2, m, p_1, p_2, h_2\}$. The numbers e and π are constants.

$$\frac{dm_1}{dt} = F_{in} - F \quad (1)$$

$$\frac{dm_2}{dt} = F - F_{out} \quad (2)$$

$$m = m_1 + m_2 \quad (3)$$

$$F_{in} = k_v(v + e^{-time}) \quad (4)$$

$$F_{out} = k_0 a\sqrt{p}_2 \quad (5)$$

$$p_1 = \frac{m_1 g}{S_1} \quad (6)$$

$$p_2 = \frac{m_2 g}{S_2} \quad (7)$$

$$p_1 = p_2 \quad (8)$$

$$h_2 = \frac{m_2}{\rho S_2} \quad (9)$$

$$S_1 = \pi r_1^2 \quad (10)$$

$$S_2 = \pi r_2^2 \quad (11)$$

Two variables appear derivated in the model: m_1 and m_2. However, Eqs. (6)–(8) define the relationship between these variables, so that one variable can be calculated from the other. As a result, the model has only one state variable. Possible selections of the state variable are the following: $\mathbf{e_1}$=$\{m_1\}$, $\mathbf{e_2}$=$\{m_2\}$, $\mathbf{e_3}$=$\{m\}$, $\mathbf{e_4}$=$\{p_1\}$, $\mathbf{e_5}$=$\{p_2\}$, $\mathbf{e_6}$=$\{h_2\}$, $\mathbf{e_7}$=$\{F\}$ and $\mathbf{e_8}$=$\{F_{out}\}$.

Interactive events

The restrictions to select interactive parameters are described in Statement 1. Eqs. (10) and (11) relate the parameters S_1 and r_1, and S_2 and r_2, respectively. The value of S_i can not be set independently of the value of r_i, and vice versa. As a result, S_i and r_i can not be simultaneously interactive quantities of an interactive event.

F_{in} can not be a state variable. Consequently, it can not be an interactive quantity (cf. Statement 2).

Statement 3 allows to decide whether a particular set of time-dependent quantities can be simultaneously changed in an interactive event. As the physical model has one state variable, only one time-dependent quantity can be changed in each interactive action. Also, Statement 3 relates the selections of interactive quantities and state variables: $\mathbf{x}^* \subseteq \mathbf{e}$. In this case, if one variable in $\{F_{out}, m_1, m_2, m, p_1, p_2, h_2\}$ is selected as interactive quantity, then it has to be selected as state variable.

The definition of the interactive events depends on the particular application. For instance, the set of interactive events of the two-tank interactive model could be $\mathbf{A} = \{A_1, A_2, A_3, A_4\}$, where:

$$A_1 = \{c_1, \mathbf{v_1^*} = \{S_1\}, \mathbf{e_1} = \{m_1\}\} \quad (12)$$

$$A_2 = \{c_2, \mathbf{v_2^*} = \{S_1\}, \mathbf{e_2} = \{p_1\}\} \quad (13)$$

$$A_3 = \{c_3, \mathbf{v_3^*} = \{\rho\}, \mathbf{e_3} = \{h_2\}\} \quad (14)$$

$$A_4 = \{c_4, \mathbf{v_4^*} = \{v, r_1, m\}, \mathbf{e_4} = \{m\}\} \quad (15)$$

The cross-section of tank 1 (S_1) is changed in the interactive events A_1 and A_2. In the first case, m_1 remains constant, and the liquid level and pressure change abruptly. In the second case, the same interactive change in S_1 produces an abrupt change in m_1, remaining the liquid level and pressure unchanged. The liquid density is changed in A_3, being the liquid level unchanged in the event. Finally, three quantities are changed when the interactive event A_4 is triggered: the pump voltage (v), the tank 1 radius (r_1) and the total mass of liquid (m).

SUPPORTING MULTIPLE SELECTIONS OF THE STATE VARIABLES

Different interactive events may require of different selections of the state variables. In this context, the state variable selection associated to an interactive event concerns only to the solution of the re-start problem for that particular type of event. It does not condition the state variable selection for solving the dynamic problem.

Modelica and Dymola support the user's control on the state variables selection, via the *stateSelect* attribute of Real variables (Otter and Olsson 2002). The attribute values include 'never' (the variable will never be selected as state variable) and 'always' (the variable will always be used as a state). If the number of variables selected as state variables by setting the value of their stateSelect attribute to 'always' is higher than the model order, then an error message is generated by Dymola. This feature allows the user to select the model state variables without performing any manipulation on the model equations. The required model manipulations are automatically performed by Dymola. However, Modelica does not allow the model designer to modify the state variable selection during the simulation run.

A method to develop interactive models supporting simultaneously different choices of the state variables was proposed in (Martin-Villalba et al. 2008). The interactive model is described as composed of several instantiations of the physical model, each one with a different choice of the state variables and adapted to perform the interactive events that correspond to this state selection. Modelica capability for state selection control (i.e. *stateSelect* attribute) allows the model developer to select the state variables without performing any model manipulation. Therefore, the interactive model is composed of as many instantiations of the physical model as different state selections are required. The adequate instantiation is used for executing each interactive event, i.e., for changing the interactive quantity values and for solving the re-start problem. Next, these calculated values are used to re-initialize the other physical model instantiations. This action guarantees that all the instantiations of the physical model describe the same trajectory.

This method, which is based on the actual capabilities of the Modelica language, has proved to work. However, it has two main disadvantages. It makes more complex the development of the interactive model. Several models need to be solved in parallel, which negatively affects the simulation execution performance. The Modelica language could be extended to facilitate the redefinition of the state variables that have to be used for solving the model at specific events. A proposal, together with different implementation aspects, will be discussed in the next sections.

IMPLEMENTATION IN MODELICA

Different techniques to develop the interactive model from the physical model are discussed.

Interactive parameters

A method to describe interactive events on interactive parameters is to define these quantities as discrete-time variables in the interactive model, whose values change only at the interactive events. Modelica's *when* clause facilitates the description of these instantaneous changes. For instance, the cross section of tank 1 (S_1) can be transformed into an interactive parameter as shown in Table 2 - Method a). `new_S1` contains the new value of the interactive parameter and the Boolean variable `c1` is the trigger condition.

However, this is not a valid approach. Dymola automatically performs model manipulations that implies symbolic differentiation of certain equations. The purpose of these manipulations is twofold: formulating the model according to the requested state selection and reducing the model index. If these model manipulations require differentiating interactive parameters, then the attempt to differentiate a discrete-time variable produces an error. For instance, using this procedure to describe the interactive events A_3 and A_1 of the two-tank model (cf. Eqs. (14), (12)) produces an error.

The approach proposed in (Martin-Villalba 2007) was to define the interactive parameters as continuous-time state variables of the interactive model. The derivative of these state variables is set to zero. As a result, the value of these states remain constant between interactive actions. The changes in the interactive parameters and input variables due to the interactive actions are described as state re-initialization events by using the Modelica's *reinit(x,expr)* function. It re-initializes a state variable (x) with the value obtained of evaluating an expression (*expr*), at the event instant. These changes are triggered using *when* clauses. An example is shown in Table 2 - Method b).

Table 2: Interactive changes in S_1.

Method	Code in the interactive model

a)	`Real S1;`
	`equation`
	`  when c1 then`
	`  S1 = new_S1;`
	`  end when;`

b)	`Real S1;`
	`equation`
	`  der(S1) = 0;`
	`  when c1 then`
	`  reinit(S1,new_S1);`
	`  end when;`

Interactive variables

Interactive events on continuous-time interactive variables can be described using the Modelica's *when* clause and the *reinit* function. The *reinit* function has only effect when applied to state variables. This is always the case, according to Statement 3. Interactive events on discrete-time interactive variables can be described modifying the *when* clause where the variable is evaluated.

STATE REDEFINITION FOR SOLVING THE RE-START PROBLEM

The capability of changing the state variable selection for solving the event restart problems would facilitate the description of interactive models. The state variable selection associated to a particular event would be active only during the solution of the restart problem of this event.

Modelica provides a mechanism to define the model state variables. However, the model developer is not allowed to describe dynamic changes in this selection. The Modelica language could be extended with two additional functions, *stateSelect* and *stateUnselect*, which accept a continuous-time variable as argument, and select or unselect the variable as state variable. The calls to these functions could only be placed inside *when* clauses. While the *when* clause is inactive, the function calls inside the clause are ignored. A function call would be executed only when the corresponding clause is triggered. The model developer needs to guarantee that the number of selected and unselected state variables is equal for any event action. The interactive events of the two-tank model are described below.

```
   Real m1(start=2, fixed=true,
           stateSelect=StateSelect.always);
   Real S1(start=0.5, fixed=true,
           stateSelect=StateSelect.always);
   Real v(start=5, fixed=true,
           stateSelect=StateSelect.always);
   Real rho(start=1000, fixed=true,
           stateSelect=StateSelect.always);
   Real r1;
equation
   der(S1) = 0; der(v) = 0; der(rho) = 0;
   S1 = pi*r1^2;
   when {c1,c2} then
     if c2 then
       stateSelect(p1); stateUnselect(m1);
     end if;
     reinit(S1,new_S1);
   end when;
   when c3 then
     stateSelect(h2); stateUnselect(m1);
     reinit(rho, new_rho);
   end when;
   when c4 then
      stateSelect(m); stateUnselect(m1);
      stateSelect(r1); stateUnselect(S1);
      reinit(v, new_v); reinit(m, new_m);
      reinit(r1, new_r1);
   end when;
```

The value of S_1 can not be set independently of the value of r_1. For this reason, only one of these parameters is defined as state variable. In this case, S_1. This selection is modified for solving the restart problem of the interactive event A_4.

CONCLUSIONS

A formal description of interactive events has been introduced. Criteria to decide whether a set of parameters and time-dependent variables of the physical model can be selected as interactive quantities have been proposed. Different procedures to describe the interactive events in Modelica have been discussed and an extension to the Modelica language has been proposed, whose goal is to facilitate performing changes in the state variable selection for solving the restart problem of interactive events.

ACKNOWLEDGEMENTS

This work has been supported by UNED, under the grant "Proyectos de Investigación propia de la UNED 2010".

REFERENCES

Åström K.J.; Elmqvist H.; and Mattsson S.E., 1998. *Evolution of Continuous-Time Modeling and Simulation.* In *Proceedings of the* 12^{th} *European Simulation Multiconference.* Manchester, UK, 9–18.

Martin-Villalba C., 2007. *Object-Oriented Modeling of Virtual Laboratories for Control Education.* PhD Dissertation, Dept Informática y Automática, UNED, Madrid, Spain.

Martin-Villalba C.; Martinez F.; Urquia A.; and Dormido S., 2010. *Development of virtual-labs based on complex Modelica models using VirtualLabBuilder. International Journal of Modeling, Identification and Control*, 9, no. 1/2, 98–107.

Martin-Villalba C.; Urquia A.; and Dormido S., 2008. *An approach to virtual-lab implementation using Modelica. Mathematical and Computer Modelling of Dynamical Systems*, 14, no. 4, 341–360.

Modelica Association, 2010. Website: `http://www.modelica.org`.

Otter M. and Olsson H., 2002. *New features in Modelica 2.0.* In *Proceedings of the* 2^{nd} *International Modelica Conference.* Oberpfaffenhofen, Germany.

THE ARCHITECTURE AND COMPONENTS OF LIBROS: STRENGTHS, LIMITATIONS, AND PLANS

Yilin Huang, Mamadou D. Seck and Alexander Verbraeck
Systems Engineering Group
Faculty of Technology, Policy and Management
Delft University of Technology
PO Box 5015, NL-2600GA Delft
The Netherlands
E-mail: {y.huang,m.d.seck,a.verbraeck}@tudelft.nl

KEYWORDS
Railway Simulation, Simulation Library

ABSTRACT

Railway systems have long life spans, during which changes take place that lead to new issues to study. These changes can ask for the construction or alteration of simulation models for an analysis of the rail system. LIBROS is an open source java package that supports distributed microscopic multi-formalism simulation of heavy and light rail operations. Since its development, the library has been applied for simulations that successfully assisted decision making for the rail infrastructures design in a couple of projects. Each project focused on one specific part of a rail-based network. During the past year, LIBROS has been updated and extended as new simulation requirements emerged. This paper addresses the strengths and limitations of LIBROS by discussing its structural design, model components, functionality, and applications. Further research of using the DEVS formalism in LIBROS is proposed to transform the library for the future challenges of rail-based network design and simulation.

INTRODUCTION

Rail transport, as one of the major forms of public transport, plays a vital role that affects our daily life (Button and Hensher, 2001). In order to increase public transport's share compared to private transport modes and to maintain and improve its competitiveness, more reliable services should be offered (Tahmassseby, 2009). Modeling and simulation of transport systems have had important developments since the mid 1970s, and now received better recognition by transport designers in decision support (Ortzar and Willumsen, 2001). A microscopic rail network model is deemed not only suitable, but also mandatory, for exact running time calculation, timetable construction, and conflict detection and resolution (Hansen and Pachl, 2008). For large and complex rail-based networks, the planning and design of the infrastructure and operation are cumbersome and time-consuming; so is the modeling of the networks. Inevitably, working with complex infrastructure networks (total or partial) increasingly becomes a standard approach (Hansen and Pachl, 2008). A medium sized urban light rail operation, for example, may already concern a dozen lines and hundreds of vehicles and stops. A typical microscopic model contains all tracks of the routes, which have to be modeled at a high resolution at stations and junctions where two or more routes converge, diverge or cross-over. Each rail-based network is unique in terms of its technical specifications and available services (Ho et al., 2002). Different aspects of a network, such as the infrastructure, signaling control, and timetables at various locations and/or time periods, are full of variety. This further increases the complexity of rail-based network modeling. Moreover, because of the inherent long life span of rail infrastructure and services, new issues and questions often come up during the lifetime of the infrastructure. Many of these are about improving or at least maintaining the quality of service after the changes, e.g. by the adaptation of timetables, acquiring new equipment, and infrastructure changes and alterations. As such, model construction and reconstruction of rail-based networks particularly require flexible model composition and configuration in order to enhance reusability and reduce time and human resource investment in this regard.

LIBROS (Library for Rail Operations Simulation) is an open source java package that supports distributed microscopic multi-formalism simulation of heavy and light rail operations. It is designed for development of rail simulation models. The library development started at the request of HTM (The Urban Public Transport, The Hague, The Netherlands), as the rail simulation tools in existence could not meet the specific needs of urban tramway and light rail operation design. Since then LIBROS has been applied for simulations that successfully assist decision making for the design of rail-based infrastructures in a number of projects (Kanacilo and Verbraeck, 2006, 2007; Kanacilo and Oort, 2008; Huang et al., 2010). Each project focused on one specific part

of an urban tramway and light rail network, e.g. the modeling and analysis of a crossing where the infrastructure would be extended and design alternatives should be evaluated and compared. The results show that LIBROS is suitable to help forecast the operations, and it enables prediction of the quality of service of a certain urban rail infrastructure configuration (Kanacilo and Oort, 2008). This paper addresses the strengths and limitations of LIBROS. Its structure design, model components, functionality, and applications are discussed in relation with the strengths and limitations. LIBROS' underlying simulation environment DSOL (Distributed Simulation Object Library) (Jacobs, 2005) is briefly explained. Recent research (Seck and Verbraeck, 2009) added the DEVS (Discrete Event System Specification) formalism to DSOL which could facilitate the building and simulation of DEVS models for LIBROS. This would especially benefit LIBROS in terms of modularity and providing a hierarchical structure of model components. Given the challenges and the complexity of network simulation for rail infrastructure, an extension of LIBROS with the DEVS formalism is one of the main additions planned.

The remainder of this paper is organized as follows. In the next section, the motivation of developing the LIBROS simulation tool is explained. It is followed by a description of how rail simulations can be carried out with LIBROS. Some encountered challenges are stated. Section 4 describes DSOL and DEVS, and how DEVS-DSOL would benefit LIBROS. In Section 5, the architecture and main components of LIBROS are discussed, as well as some opportunities for enhancements of the simulation library.

THE NEED FOR LIBROS

In the field of rail transport network planning and design (or transport in general), a number of simulation tools have been developed, e.g. simulation models of stations or terminals (Carey and Lockwood, 1995; Carey and Carville, 2002, 2003; Rizzoli et al., 2002), and train network simulators, such as Simon/TTS (Wahlborg, 1996), TOPSim (Sandblad et al., 2000), SIMONE (Middelkoop and Bouwman, 2001), OpenTrack (Nash and Huerlimann, 2004), VirtuOS (Kavicka and Klima, 2000), RailSys (Bendfeldt et al., 2000), UX-SIMU (Kaas, 2000), Multi-train simulator (Ho et al., 2002), SimMETRO (Koutsopoulos and Wang, 2007).

The motivation of developing a new rail-based simulation tool is multi-faceted. First, many railway models and simulators are designed to assess a limited number of aspects (e.g. timetabling, signaling control) of rail operations or to study a particular part (e.g. a station, a junction) of the rail network (Ho et al., 2002). It is impractical to carry out various studies with different simulation tools. Some models have a high abstraction level (Vromans et al., 2006), which may cause a significant difference between model outcomes and real operations on a lower abstraction level (Ferreira, 1997). Although rail operations can be decomposed into different aspects, all should be taken into consideration in a self-contained simulation package for analyzing the rail network on the micro level (Krueger et al., 2000). Second, very few simulation tools support tramway or light rail operations. To the authors' knowledge, one of the few is RailSys (Rudolph, 2000). In comparison with heavy rail operations, many differences occur when simulating light rail operations. Heavy rail vehicles drive in signalled blocks, while light rail vehicles also "drive on sight" (Overton, 1989). Given the large number of cities with metro and tramway systems, there is a growing need for tools that specifically aim at light rail simulation. Third, tools designed for diverse transport agencies are very specific to individual agencies' needs, often making the tools less suitable for other agencies or transport operators. This asks for generic tools that can be applied for different situations and allow for analysis from different viewpoints. Fourth, commercial rail simulators generally have good performance, but they raise proprietary issues. Concerning inter-operability, they are difficult to be modified or linked with other tools or information systems such as databases or GIS (Kanacilo and Verbraeck, 2006). Cost is obviously another concern.

The research team thus decided to develop an open source rail simulation library. On the one hand, it is tailored for light rail simulation, considering the combined impact of different aspects of the infrastructure design in one self-contained simulation package. On the other hand efforts are taken to make the library also suitable for heavy rail simulation. The fact that LIBROS is developed as an open-source project provides a unique possibility to improve the package, and adds flexibility for further research. The package is available for any party to conduct research.

RAIL SIMULATION WITH LIBROS

The use of LIBROS is straightforward, as shown in Fig. 1. Users need to specify the simulation model and its parameters in XML format. They can define the rail infrastructure, control measures, timetables, and change options such as if animation and data visualization are needed. The model generator of LIBROS verifies the XML input, then creates and initializes the simulation model using the available model components in the package. If needed, other data sources such as timetables or GIS maps can be added. Experiments are generated according to the user configuration. In the model, the vehicle movement is simulated continuously and the other components such as the traffic lights and switches are simulated using event scheduling. The results can be animated, plotted, and (t-v, t-x, x-v) graphs are generated. Data files recording the vehicle movements, waiting times, etc., are generated and categorized.

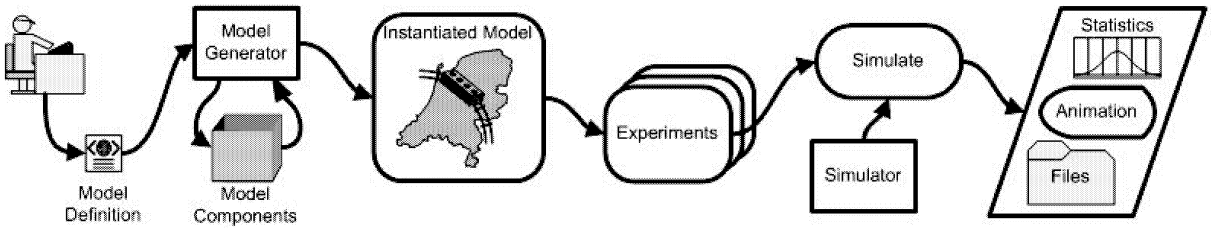


Figure 1: Rail Simulation with LIBROS

As stated earlier, LIBROS has been used in a number of projects that showed good results. Some challenges we encountered include the following. For example, the infrastructure configuration is XML-based, making the task difficult for non-experts. Here a GUI could help by providing drag-and-drop model components which has become quite common in commercial simulation tools. Modeling by means of drag-and-drop requires the relation between model components to be cut-and-dried. The DEVS formalism (Zeigler et al., 2000) provides a modeling theory of such modularity and hierarchical structure which the LIBROS model lacks. Some transformation of the library is therefore desired. The DEVS formalism could also benefit the library design in terms of information exchange between the components and model state saving. These issues are discussed in relation with the library design in the following sections.

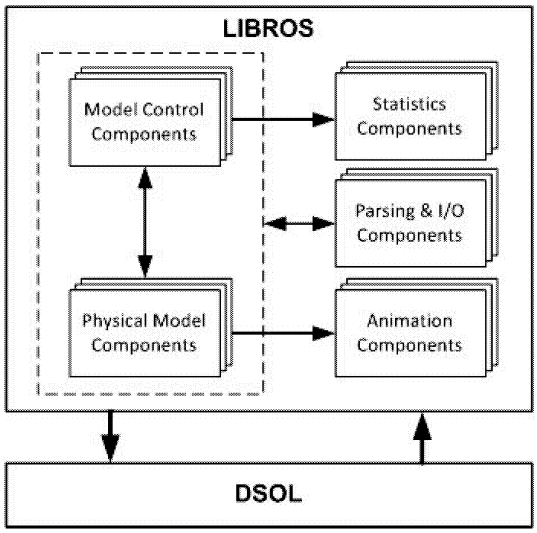


Figure 2: The LIBROS Architecture

DSOL AND DEVS

LIBROS is built as an extension of DSOL (Jacobs, 2005), an open source java simulation library that supports discrete and continuous formalisms, and provides generic simulation services such as various simulators, specification of experiments, event scheduling, and probability distributions. As a recent development (Seck and Verbraeck, 2009), DSOL ES-DEVS implements the parallel DEVS formalism on top of the DSOL library. DEVS (Zeigler et al., 2000) is a modeling and simulation formalism that allows for formal specifications of systems. Two levels of specifications are possible. The atomic DEVS formalism consists of (input/output/state) sets, and functions on the sets allowing complete and unambiguous specification of systems according to the discrete event abstraction. The coupled formalism consists of input, output, components (either atomic or coupled), and coupling relation sets. Along with the formalism, modularity is guaranteed and the closure under coupling property (Zeigler et al., 2000) allows for the construction of hierarchical models. The ES-DEVS simulation protocol is based on the event-scheduling worldview wherein executions of the internal transition function are scheduled according to the specified time advance function and unscheduled at the reception of external events. The ES-DEVS implementation strictly follows the DEVS formal specification, and the separation of concerns between models and simulators is respected. Dynamic structure DEVS is also implemented so that components and coupling relations can be added and removed dynamically during simulation runtime. This feature can be especially useful in dynamic systems such as transport systems in general and rail transport in particular, where the relations between the simulation components (e.g. a vehicle and a traffic light) are temporary and subject to change.

THE LIBROS SIMULATION LIBRARY

In the development process of LIBROS, an effort is made to overcome the issues addressed in Sect. and to design, develop and test a component-based, loosely-coupled rail simulation package. LIBROS uses part of the DSOL services, and extends them to define more rail specific simulation components. Fig. 2 illustrates a simplified component view of its architecture. The library contains two major groups of components: those that form the building blocks of the simulation model, which is the core of LIBROS; and those that offer peripheral simulation services such as parsing model definition files, generating statistics, animation, and outputs. The rail model structure can be divided into two layers. The first layer is a collection of the physical model components.

Most of them represent the (physical) rail and road infrastructure elements, such as tracks, stations, vehicles and intersections. But there can also be some virtual elements needed by the simulation, e.g., the locations where vehicles enter and exit the (simulation) system, or where data shall be collected for statistics. Instantiation of a model component creates the representation of a physical or virtual element and its initial state. The second layer is composed of the model control components, which define the control logic (i.e. state transition) of the physical model components in the first layer. The control logic can be rule-based or dependent on the interactions between different model components, e.g. the rules that define how several traffic lights shall coordinate their signals, how priorities are given to vehicles at junctions, the acceleration or deceleration of a vehicle according to the speed limits or based on obstacles. The separation of physical model and model control components simplifies the setup by which a physical model component can implement different model control strategies. A traffic light, for example, may have fixed time intervals for signal changes, or change signals depending on the traffic conditions. In this way, components can be easily extended and updated. The communications between different model components are also handled by the model control components using the publish-subscribe interaction scheme, which is discussed later. LIBROS provides XML schema for model configurations. The XML definition is parsed and verified by the input (processing) components, which then creates models and simulation replications. A physical model or model control component is associated with one or more statistics components or animation components (if needed). The statistics components collect important model states, and generate graphs of the key performance indicators for the rail network operation. The animation components are able to plot GIS data as background maps, and display the signalling changes, vehicle movements, etc. Output components generate and save the simulation results including graphs. Fig. 3 shows a simplified diagram of the model components. (Some interfaces and classes are omitted for clarity.) As mentioned earlier, the model structure has a physical layer and a control layer.

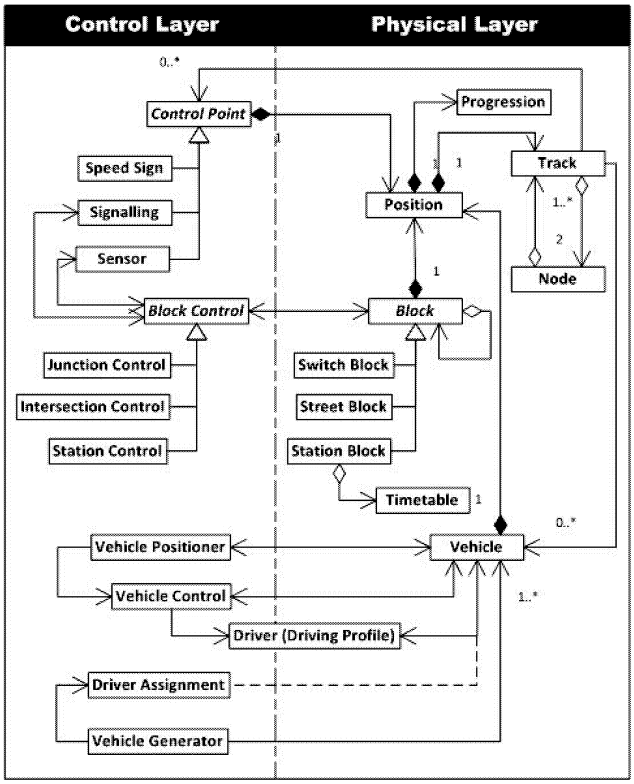


Figure 3: Overview of the LIBROS Components

Infrastructure Modeling

An accurate infrastructure model is important for rail simulation, as it is the basis for all calculations. In LIBROS, a combination of link-oriented and node-oriented approaches is chosen, as both have advantages and disadvantages (Hansen and Pachl, 2008). A rail network is a directed graph $N = (V, T)$ following Bang-Jensen and Gutin (2009). V is a set of vertices (or *node*s). T is a set of directed track segments (or *track*s). Each track is an ordered pair of distinct nodes (i.e. the two ends of the rail axis). The location (real world coordinates) of a track is saved in its starting and ending nodes. A rail switch is located at a node $v \in V$ with an out-degree $d_N^-(v) = 2$. A (switch) node contains information of which direction (i.e. the next track) a vehicle shall move to. The *position* of the other infrastructure elements (e.g. stops, traffic lights, sensors, speed limits) are saved in association with tracks. The *progression* is defined as the distance between the element and the starting node of the track.

Strengths: The infrastructure model is microscopic. This enables precise calculation of running times and evaluation of control strategies. High quality animation is possible; an example is shown in Fig. 4.

Limitations: The infrastructure configuration is time-consuming. Each node is defined by coordinates. Track lengths are calculated by end nodes and curvatures. Detailed geographical data are required for model setup.

Plans: The model should allow for simpler track definition, e.g. with only lengths and speed limits. This will reduce the complexity of the infrastructure configuration. Methods can be added for track information refinement if more detailed visualization of the infrastructure is desired.

Block System Modeling

In railway control, a block system is a signaling system that provides safe spacings for vehicles (Hansen and Pachl, 2008). A block section is a section of track where

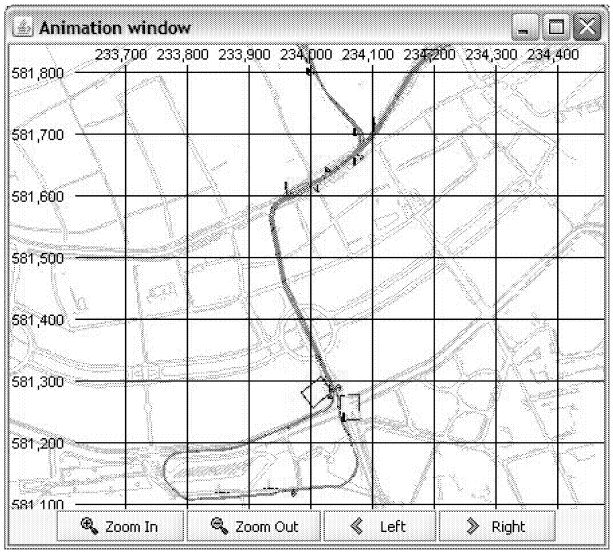


Figure 4: An Example of the Animation

a vehicle may only enter when the section is not occupied by other vehicles (Pachl, 2002); each section is guarded by a signal at the entrance. In the model, the concept is extended. Interlocking (at e.g. junctions and for single-tracks) also uses the concept of *block* and *block control*, because from a modeling perspective, block systems and interlocking operate in a very similar way, i.e. vehicles may not enter a locked section, and the signalling logic of different traffic lights in one area is often interdependent. The block control components define the diverse signalling logic. It can be traffic dependent. At major intersections in a city, trams may not have priority over the other street traffic; hence users can configure the vehicles' waiting time distributions. Alternatively, the signalling logic can depend on the occupation state of another block. In this case, users can define the dependencies between different blocks using sub-blocks. A simple example is shown in Fig. 5. The crossing is guided by 3 traffic lights (thus 3 blocks); 4 driving directions are possible. Three sub-blocks are defined in the example. Passing through direction 1 needs sub-block 1; direction 2 needs sub-blocks 2 and 1; directions 3 and 4 need sub-blocks 3 and 2. The logic is very intuitive: when a sub-block is occupied by a vehicle, it can not be used by another. Thus the state of the traffic light depends on the availability of the required sub-blocks. The release time of a sub-block is identified precisely, e.g. once a vehicle at direction 2 releases sub-block 2 (i.e. the vehicle's tail left sub-block 2, but sub-block 1 is still occupied), the access of direction 3 or 4 is granted if there is any request. A vehicle requests the access of a block by triggering a request sensor placed in front of the traffic light. If the access is granted, the required sub-blocks are set to be locked immediately. The *sensor*s can be of different types. The most common ones are request sensors and release sensors. The former are

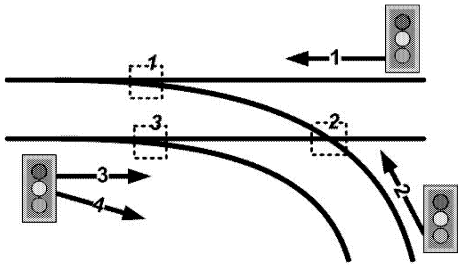


Figure 5: Block Control using Sub-blocks

triggered by a vehicle's head, and the latter by a vehicle's tail. If there are conflicts at a crossing and requests are queued, the access is granted to the vehicle that has the highest priority.

Strengths: The block (and sub-block) components provide users the possibility to configure different signalling logic for e.g. crossings, stations, single tracks, and any combination of these infrastructures. The signalling logic for sophisticated infrastructure settings, e.g. where different light rail lines intersect in city centers and near train stations, has been modeled by this method.

Limitations: Configuration of block system models is hard for non-experts. The model components lack modularity. For example, to model a crossing, each block (sub-block), switch (direction), and sensor have to be configured individually.

Plans: Higher level components should be constructed. The authors are aware of the complexity of rail infrastructures. Because each element is unique, constructing components for all situations is impossible and unnecessary. But for some standard and often occurring situations, components can be designed to simplify the configuration. It is important to separate the functional definition of a component with its geoinformation.

Vehicle Modeling

The *vehicle* component saves the vehicle's state and other information used for simulation and statistics, e.g. its size, speed, position, and "visible" objects (speed signs, signals, or a vehicle in front). Vehicles are generated according to timetables.A probability distribution can be introduced to simulate the earlier and delayed departure. The *vehicle generator* also assigns each vehicle a *driver/driving profile* that determines how a vehicle accelerates, cruises, and brakes. The vehicle movement is calculated by *vehicle positioner* using the Runge-Kutta integrator. Given the speed of a vehicle at time t_n, it computes the vehicle's speed and position (distance) at time t_{n+1} based on the vehicle's acceleration changes during the integration time-step. The acceleration rates are determined by the *vehicle controller*. As stated before, the vehicles can "drive on sight", meaning that a vehicle "sees" also other objects besides traffic lights. The objects being considered in the model are

(1) traffic lights, (2) changes of infrastructure, e.g. stops or stations, curves, (3) speed signs, (4) obstacles, e.g. a vehicle in front, and (5) other objects, e.g. an intersection or a switch, that by regulation a vehicle shall pass through with reduced speed. In principle, at each integration time-step, the vehicle "checks" the tracks in front of it, whether there are any objects associated with the tracks. The vehicle controller then decides if the vehicle shall accelerate, cruise or brake.An object may change state, e.g. a traffic light or a vehicle. If so, the object notifies the approaching vehicle using the publish-subscribe interaction scheme.
Strengths: Driving profiles introduce randomness into the driving behavior. "Drive on sight" is important for modeling light rail operation. The calculation of vehicle movement is precise and in detail.
Limitations: Checking acceleration and integration at each time-step has a high computational cost. Profiling the simulation program shows that this part of the calculation takes 74% of the execution time. The performance decreases when simulating large scale networks with many vehicles.
Plans: More efficient algorithms can be developed. A vehicle can check as far as possible until an object of interest is found, e.g. a traffic light, a sharp curvature, or another vehicle. Once an object of interest is found, the vehicle decides if and when it shall accelerate or brake; and the object "informs" the vehicle if the state of the object has any change, based on which the vehicle may react.

Asynchronous Messaging

The object-to-object communication in LIBROS (and DSOL) is accomplished by the publish-subscribe (or event notification) interaction scheme. With systems based on this scheme, subscribers register their interest in an event, or a pattern of events, and are subsequently asynchronously notified of events generated by the publishers (Eugster et al., 2003). The strength of this event-based interaction style lies in the full decoupling in time and space between publishers and subscribers (Eugster et al., 2003). Furthermore, asynchronous communication prevents disproportionate polling between objects and enables well tailored communication between potentially distributed objects (Jacobs, 2005). The publish-subscribe scheme is implemented following the *observer pattern* (also known as the *publish-subscribe pattern*) which defines a non static one-to-many dependency between objects so that when one object changes state, all of its dependents are notified and updated automatically (Gamma et al., 1994). For example, when simulating a vehicle approaching a traffic light, the state change of the traffic light is obviously of interest to the vehicle. Thus the vehicle registers to the traffic light's *subscribers list* and becomes a listener of the state changes of this traffic light in order to be notified. Once the vehicle passed the traffic light, it deregisters itself from the list. In LIBROS, not only the communication between the model components uses such a scheme but also the simulation services (animation, statistics, file generation, etc). Concerning implementation of the observer pattern (or the publish-subscribe scheme in general), common issues include unexpected updates (Gamma et al., 1994), thread-safety, and lapsed listeners (Goetz, 2005). When these issues are treated with prudence, the publish-subscribe interaction scheme is an efficient and convenient method for asynchronous communication.

Data-Driven Simulation

In a recent paper (Huang and Verbraeck, 2009), the authors proposed a dynamic data-driven approach for rail transport simulation. The idea is to automatically perform model calibration and validation by comparing the model output with the rail operation data. The model states at each state transition are saved to compare with the available data. The model is duplicated so that different model calibrations can be simulated in parallel to evaluate which parameter configuration is better. In this regard, efficient model state save and component copy are of importance. With the current library, the model states are periodically written to output streams for state saving. With the DEVS formalism, state changes are formally defined by the internal and external transition functions ($\delta_{int}, \delta_{ext}$), which make it easier to trace the state transition and causality. The model can be saved or copied at each state transition. Therefore, transformation of LIBROS model components by using the DEVS formalism would also benefit the development of automatic model calibration and validation.

CONCLUSIONS

This paper discussed the architecture and model components of LIBROS, and the strengths and limitations of its design. The library supports distributed rail simulation that uses configurable components as model building blocks. Not only does the library support rail transport design from an engineering perspective, its advanced animation and visualization capabilities makes it an efficient means of communication and enforces common understanding between transit authorities, service providers, as well as other parties involved. The library has been used in projects that showed good results (Kanacilo and Verbraeck, 2006, 2007; Kanacilo and Oort, 2008; Huang et al., 2010). Microscopic modeling offers simulation experiments with high detail that is important for timetable construction and conflict detection and resolution. The library currently models the vehicle movement using differential equations solved by numerical integrators. This solution offers high simulation precision but comes with a high computational cost, which hinders software performance. Infrastruc-

ture configuration can become very complex for the configuration of large scale rail networks. Both limitations can be mitigated by using the DEVS system theoretical formalism which offers a formal specification for modular and hierarchical discrete event systems. The transformation of LIBROS using the DEVS formalism is underway. When LIBROS uses DEVS formalism, the communication between model components would naturally use message transmission through ports. The communication between the simulation services will remain using the publish-subscribe scheme. Our next step is to extend the library for data-driven simulation through which automatic model calibration and validation can be performed (Huang and Verbraeck, 2009). In this context, the DEVS formalism will also benefit the library design.

ACKNOWLEDGMENTS

The authors would like to acknowledge the support of HTM Urban Public Transport, the Netherlands.

REFERENCES

Bang-Jensen J. and Gutin G., 2009. *Digraphs: Theory, Algorithms and Applications.* Springer Monographs in Mathematics. Springer Science, 2nd ed.

Bendfeldt J.P.; Mohr U.; and Mller L., 2000. *RailSys, a system to plan future railway needs. Advances in Transport*, 7, 249–255.

Button K.J. and Hensher D.A. (Eds.), 2001. *Handbooks in Transport 3: Handbook of Transport Systems and Traffic Control.* Elsevier Science.

Carey M. and Carville S., 2002. *Testing schedule performance and reliability for train stations. Journal of the Operational Research Society*, 51, no. 6, 666–682.

Carey M. and Carville S., 2003. *Scheduling and platforming trains at busy complex stations. Transportation Research Part A: Policy and Practice*, 37, no. 3, 195–224.

Carey M. and Lockwood D., 1995. *A Model, Algorithms and Strategy for Train Pathing. The Journal of the Operational Research Society*, 46, no. 8, 988–1005.

Eugster P.T.; Felber P.A.; Guerraoui R.; and Kermarrec A.M., 2003. *The many faces of publish/subscribe. ACM Computing Surveys*, 35, no. 2, 114–131. ISSN 0360-0300. doi:http://doi.acm.org/10.1145/857076.857078.

Ferreira L., 1997. *Planning Australian freight rail operations: An overview. Transportation Research Part A: Policy and Practice*, 31, no. 4, 335–348.

Gamma E.; Helm R.; Johnson R.; and Vlissides J., 1994. *Design Patterns: Elements of Reusable Object-Oriented Software.* Addison-Wesley.

Goetz B., 2005. *Java theory and practice: Be a good (event) listener.* IBM Java Technology Technical Library.

Hansen I.A. and Pachl J. (Eds.), 2008. *Railway Timetable & Traffic: Analysis-Modelling-Simulation.* Eurailpress.

Ho T.; Mao B.; Yuan Z.; Liu H.; and Fung Y., 2002. *Computer simulation and modeling in railway applications. Computer Physics Communications*, 143, no. 1, 1–10.

Huang Y. and Verbraeck A., 2009. *A Dynamic Data-Driven Approach For Rail Transport System Simulation.* In M.D. Rossetti; R.R. Hill; B. Johansson; A. Dunkin; and R.G. Ingalls (Eds.), *Proceedings of the 2009 Winter Simulation Conference.* IEEE, 2553–2562.

Huang Y.; Verbraeck A.; van Oort N.; and Veldhoen H., 2010. *Rail Transit Network Design Supported by an Open Source Simulation Library: Towards Reliability Improvement.* In *Transportation Research Board 89th Annual Meeting Compendium of Papers.* 10-0310.

Jacobs P.H.M., 2005. *The DSOL simulation suite - Enabling multi-formalism simulation in a distributed context.* Ph.D. thesis, Delft University of Technology, the Netherlands.

Kaas A., 2000. *Punctuality model for railways. Advances in Transport*, 7, 853–860.

Kanacilo E.M. and Oort N.v., 2008. *Using a rail simulation library to assess impacts of transit network planning on operational quality.* In *WIT Transactions on the Built Environment*, WIT Press, 103. 35–43.

Kanacilo E.M. and Verbraeck A., 2006. *Simulation services to support the control design of rail infrastructures.* In *Proceedings of the 2006 Winter Simulation Conference.* IEEE, 1372–1379.

Kanacilo E.M. and Verbraeck A., 2007. *Assessing tram schedules using a library of simulation components.* In *Proceedings of the 2007 Winter Simulation Conference.* IEEE, 1878–1886.

Kavicka A. and Klima V., 2000. *Simulation support for railway infrastructure design and planning processes. Advances in Transport*, 7, 447–456.

Koutsopoulos H. and Wang Z., 2007. *Simulation of Urban Rail Operations: Application Framework. Transportation Research Record*, 2006, 84–91.

Krueger H.; Vaillancourt E.; Drummie A.M.; Vucko S.J.; and Bekavac J., 2000. *Simulation within the Railroad Environment.* In *Proceedings of the 2000 Winter Simulation Conference*. 1191–1200.

Middelkoop D. and Bouwman M., 2001. *Simone: Large Scale Train Network Simulations.* In *Proceedings of the 2001 Winter Simulation Conference*. IEEE, 1042–1047.

Nash A. and Huerlimann D., 2004. *Railroad simulation using OpenTrack. Advances in Transport*, 15, 45–54.

Ortzar J. and Willumsen L., 2001. *Modelling Transport.* John Wiley & Sons, 3rd ed.

Overton D., 1989. *Traffic signal control of LRVs.* In *IEE Colloquium on Light Rapid Transit On-Street.* 9/1–9/3.

Pachl J., 2002. *Railway Operation and Control.* VTD Rail Publishing.

Rizzoli A.E.; Fornara N.; and Gambardella L.M., 2002. *A simulation tool for combined rail/road transport in intermodal terminals. Journal of Mathematics and Computers in Simulation*, 59, no. 1-3, 57–71.

Rudolph R., 2000. *Operational simulation of light rail systems.* In *Proceedings of the European Transport Conference*. 167-178.

Sandblad B.; Andersson A.; Jonsson K.E.; Hellstrm P.; Lindstrm P.; Rudolf J.; Storck J.; and Wahl-borg M., 2000. *A train traffic operation and planning Simulator. Advances in Transport*, 7, 241–248.

Seck M.D. and Verbraeck A., 2009. *DEVS in DSOL: Adding DEVS operational semantics to a generic Event-Scheduling Simulation Environment.* In *Proceedings of the 2009 Summer Computer Simulation Conference.*

Tahmassseby S., 2009. *Reliability in Urban Public Transport Network Assessment and Design.* Ph.D. thesis, Delft University of Technology, The Netherlands.

Vromans M.J.C.M.; Dekker R.; and Kroon L.G., 2006. *Reliability and heterogeneity of railway services. European Journal of Operational Research*, 172, no. 2, 647–665.

Wahlborg M., 1996. *Simulation models: Important aids for Banverket's planning process.* In *Computers in Railways*, WIT Press, vol. V. 175–181.

Zeigler B.P.; Praehofer H.; and Kim T.G., 2000. *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems.* Elsevier/Academic Press, 2nd ed.

MODELING AND SIMULATION OF THE BIOFUEL ELECTRO HYDRAULIC INJECTION SYSTEMS BY AMESIM

Nicolae Vasiliu, Daniela Vasiliu, Constantin Calinoiu
Power Faculty
University Politehnica of Bucharest,
RO 060042, Bucharest, Romania
vasiliu@fluid-power.pub.ro

Ion Manea
Flower Power USA Inc.
902 4th St., SW Suite B, Auburn, WA. 98001
U.S.A.
imanea@flowerpowerfuel.com

KEYWORDS
Modeling, simulation, biofuel injection, AMESim

ABSTRACT

This paper contains the comparative results of preliminary computer simulations of a mechanically and an electronically controlled injectors running on diesel fuel and high oleic sunflower oil. The models are taking into account the specific viscosity, density and bulk modulus of elasticity at fuel temperatures at 40°C and 95 °C.
The aim of the paper was to use a model of the two-stage electro hydraulic and a mechanical injection system in relation to two main parameters: injection timing and injection duration. The computer models and simulation software models were those provided by LMS Imagine.Lab-AMESim software. The vegetable oil data was provided by Flower Power USA. The simulation models were obtained by reducing the initial complex model using different model reduction techniques like activity index and state count. Different levels of injection valve models were considered in order to keep the time continuity and to evaluate the simplification techniques.
The results show a considerable distortion of injection timing and rate of fuel injection and demonstrate that even with fuel heated at engine coolant temperature, further injection system modifications are necessary in order to address injection timing and fuel injection fuel rate issues.

INTRODUCTION

Pure plant oil (PPO) represents a fuel alternative option that, in some technical and environmental aspects, is superior to other alternative fuels for which use entails a need for a specially designed engine (or modifications for current engines) as well as a need for a separate distribution infrastructure, thus it is assumed that there is little reason to see PPO as the primary fuel of the future. However, if used in agricultural equipment as fuel, coolant, lubricant and hydraulic fluid, PPO (let's refer to it as AgroFuel (AGF) does have its benefits, and therefore should be given equal treatment as compared to other biomass-based, carbon dioxide (CO2) neutral, bio-degradable, multi-purpose fluids. The vast majority of the current farm mobile equipment cannot run on ethanol, wind, solar or plug-in electricity but AGF could be farm produced and used by farmers to continue to economically grow wholesome food as long as the Sun shines and regardless of the future of fossil fuels. AGF can be economically manufactured and used where it is needed the most, at the farm and for the farm. Unlike other bio-fuels, its production is simpler, requires much less energy than other bio-fuels. and it has no fire, explosion, and toxic hazards and waste disposal problems.

AGF'S SUSTAINABILITY

By using the sunflower meal as soil fertilizer, farm producing biofuel is a 100% sustainable activity. All we take from the land will be oil, a result of plant magic that tracks the sun, absorbs the energy, and uses it to split the water and carbon dioxide, release most of the oxygen and combine the remaining oxygen, hydrogen and carbon into an energy rich oil for later use (it is solar technology at its best). Designed by God, the sunflower is the best natural solar panel and long energy storage, accomplished in a process that is renewable and self-replicable.
A few generations ago, our forefathers dedicated 25% of their land to crops for their draft horses. Today, we know for a fact that with only 12% of land for sunflower, a farm can be fuel sufficient while continuing to produce safe abundant and affordable food, feed, fiber and a better bottom line. That is a 13% increase in land utilization for food and 100% decrease of back breaking labor.
Good food growing is farmers' business and responsibility. As stewards of the land and front line environmentalists, they will produce and utilize AGF purposely so as not to compete with food production but to reduce production costs while improving the land.
The concept of using biomass-based liquid fuels, specifically vegetable oils (VO) as diesel fuel alternatives, is not new. Rudolf Diesel himself envisioned that his engine could run on VO. It appears that Rudolph Diesel himself originally thought that a high-efficiency engine and utilization of locally available fuels would enable independent countries and people to *"be supplied with power and industry from their own resources, without being compelled to buy and import coal or liquid fuel"*(Diesel, 1912,1913). In 1912 he stated that: *"The diesel engine can be fed with vegetable oils ... and as it can also be used for lubricating oil, the whole work can be carried out with a single kind of oil produced directly on the spot. Thus this engine becomes a really independent engine...The fact ...may seem insignificant to-day, but such oils may perhaps become in course of time of the same importance as some natural*

mineral oils and the tar products are now... In any case, they make it certain that motor-power can still be produced from the heat of the sun... even when all our natural stores of solid and liquid fuels are exhausted." (Diesel, 1913).
Already tapping into our strategic reserves while at war to forestall disruption of our fuel supply and forced to send our sons and daughters to shed blood for foreign oil, AGF has the potential for improving that situation.

AGF VERSUS *"BIODIESEL"*

Biodiesel is distinctly defined (only in the eyes of US regulations, for tax and registration purposes) mono alkyl methyl ester. Ultimately, both Biodiesel and AGF are used for power generation from biomass oil. Biodiesel is obtained from AGF as a feed-stock for a chemical process that may require at least two times more (additional) energy input than that required for producing AGF alone. The main chemical input in biodiesel production is methanol (a fossil fuel derived alcohol) and caustic ash, both toxic and hazardous. In the process, 10% of the AGF feedstock is lost as a low value glycerol waste stream along with ample waste water, both requiring proper disposal.
In general, production of Biodiesel on the farm is difficult and current production facilities are in essence chemical processing plants. Economy of scale, quality assurance, and public and environment safety dictates that large scale, complex chemical process technologies be integrated. Biodiesel production involves:

- Fire, explosion and toxic hazards and waste disposal problems.
- Around the clock controlled dedicated personnel.
- Quality assurance for the production, distribution and usage.
- Lengthy and costly zoning and permitting process.

The finished product has its own additional problems that need to be addressed such as: temperature and oxidation stability, fuel solvency, water and sediments, residual alcohol, completeness of catalytic reaction, and fuel systems material compatibility, dedicated fleet for distribution
Biodiesel is currently uneconomical to manufacture on a small scale, so additional road miles are needed to ferry the raw vegetable oil and chemicals to a centralized base. The finished Biodiesel must then be transported back to the various points of sale including the farms where the vegetable oil originated. It is safe to say that Biodiesel manufacture alone has a much larger environmental footprint than AGF production and use.
Biodiesel is not sustainable for a rural-local economy. Biodiesel plants tend to be expensive to set up, in order to comply with the vast array of legislation that governs chemical processing. It's not financially viable, for instance, for each town to have its own 'micro' biodiesel processor. This cannot be said for AGF. A transportable seed crusher may be a shared facility that can be used by multiple communities for a genuine example of a local, 'on the farm,' value added economy.
Biodiesel is not safe for on farm production. Unfortunately, even at industrial scale, it appears that due to negligence, incompetence and just plain safety violations, Biodiesel facilities tend to explode, burn to the ground, and in the process injure and kill people. Tragically, in the U.S alone, in a three-year period (2006-2009) there were eight fires and six explosions that killed two and injured five people (Noor Azian et al., 2001).
Taking into account all of the above, it is obvious that it is difficult to produce a commercial grade biodiesel on a family farm.
With Biodiesel at higher cost than off-road fossil fuel diesel, and no availability in rural areas, farmers continue to use polluting high sulfur fuel, bad for farmers and the environment.
Additionally, there are no incentives or benefits for farmers to sell their crops at wholesale prices and then buy back (in the form of fuel and animal feed) at the retail prices.
By contrast, advantages of the local production of vegetable oil include:

- A simple process as the only steps are cold pressing and filtering, therefore the production could be decentralized and create jobs in rural areas.
- De-centralization of production allows a minimization of raw material transport, an optimization step crucial for overall energy balance.
- Low energy consumption in optimized production, for instance compare energy utilization for the production of fossil fuel (13%), Biodiesel (26%) and AGF (12%).
- AGF is biodegradable and in Germany, for instance it is classified as no hazard to ground or surface water while Biodiesel, a water hazard, is classified the same as heavy oil.
- Producing and using their fuel and meal could reduce farm input, thus lower the cost of farming

AGF VERSUS FOSSIL DIESEL FUEL

Fossil Diesel fuel is a blend of nasty chemicals (aromatics, sulfur etc.). From the standardization point of view, the fuel is not a pure chemical but an amalgam with characteristics that vary with its geographical origin.
The current fossil fuel standard describes only its characteristics in fuel terms but not its contents.
The fossil Diesel fuel has been poisoning the air and waters for the last hundred years, and we still don't know what we pour into our engines and what we get out.
By contrast, AGF is a valuable alternative, which could open the bio-fuel business opportunity to farmers and help sarmer to survive.
AGF could provide the fuel for farming to continue if, God forbid, a bomb goes off somewhere and the price of oil jumps sky high.
Plant oil has been embraced by humans for a millennia for food, fuel, medicinal and spiritual use, and there are no known adverse effects (humans adapted and evolved with it) thus it is safe.
Among oilseed plants, sunflower is the best solar panel and energy storage designed by God. It is self replicating year after year and it produces fuel, food and fodder where it is needed the most, at the farm.

Sunflower oil is intrinsically safe and may be stored at home, kids and pets can drink it, and it is not going to set the house or car on fire.
If AGF is made from only one seed variety and refined with identical technology by trained farmers, fuel variance is drastically minimized and the quality is built in naturally. The sunflower plant can only produce what is genetically programmed for.
In order to explore the suitability of using AGF in diesel engines, the authors has conducted a simulation of two injection parameters, timing and rate of injection as applicable for mechanical and common rail solenoid controlled injectors with fossil diesel fuel and AGF at 40°C an 95°C.

MATERIALS AND METHODS

Short season high oleic sunflower (HOSF) cultivars were grown and harvested in Washington and Oregon States under non irrigated and irrigated conditions and as a single (spring planting) or double (summer planting) crop by utilizing existing farm infrastructure and agricultural equipment (specific corn or wheat farming).
The yields were specific to those under similar conditions in traditional sunflower regions (Dakotas, Kansas and Texas). The oil was extracted by cold pressing in mechanical press as those typically in farm operations.
The seeds moisture level was below 12% and the oil was clarified by mechanical settling and fine filtration. As is, the oil meets and exceeds most of the specifications of DIN V 51 605 rape oil fuel standard as follows:

Table 1: HOSF Properties

Property	Results	Specification limits	Units
Density (15°C)	915 to 915	900-930	kg/m^3
Flash point	201 to 270	min. 220	°C
Kin. Viscosity (40°C)	39.42-39.63	max. 36	mm^2/s
Calorific value, lower	37,149-37,30	min. 36,000	kJ/kg
Cetane Number (DCN)	49.6 to 62.8	min. 39	-
Carbon residue	0.16	max. 0.40	% (m/m)
Iodine value	84	95-125	g I/100g
Sulfur content	1< to 1.2	max 10	mg/kg
Total contamination	11.33 to 45.0	max. 24	mg/kg
Acid Value	7.45 to 11.33	max. 2,0	mgKOH /g
Oxidation stability 110°C	5.6 to 12.2	min. 6.0	h
Phosphorus content	13.8 to 21.1	max. 12	mg/kg
Earth alkali (Ca + Mg)	2.9 to 22.4	max. 20	mg/kg
Ash content	<0.001 to 0.001	max. 0.01	% (m/m)
Water content	495 to 595	max. 750	mg/kg

For reference here Diesel Fuel Oils as per ASTM 675 standard specifications are as follows:

Table 2: Diesel Fuel Oils Properties

Property	Specification/grade 2D	Unit
Density (40°C)	849	kg/m^3
65°C	869	
95 °C	893	
Flash point	52	°C
Kin.Viscosity (40°C)	min. 1.9	mm^2/s
Kin.Viscosity (40°C)	max. 4.9	mm^2/s
Calorific value (lower)	42.0	kJ/kg
Cetane Number, min.	40	-
Carbon residue	0.35	on 10% mass max
Sulfur	2.0	% mass max (m/m)
Ash	0.10	% mass max (m/m)
Water sediment	0.05	% vol. max.

The HOSF oil under investigation has a ultra high content of monounsaturated fatty acids (>90%) and a low content of polyunsaturated fatty acids (5%<) thus the viscosity-temperature relationship can be predicted based on the amount of polyunsaturated PUFA and monounsaturated (MUFA) fatty acids (FA) present. A mathematical equation developed in (Fasina et al., 2006) relates absolute viscosity (μ) to temperature and mass fraction (y) of monounsaturated fatty acids as follows:

$$\mu = A \exp(B/RT) + y \exp(C/RT) \quad (1)$$

Where R is the universal gas constant (8.314 kJ/kg mol K) and T is the absolute temperature (K) with the values of constants A, B and C as 3.31×10^{-5}, 3.55×10^{4} and 5.17×10^{3} respectively.
By using the method described in (Fasina et al., 2006) it was determined that for an plant oil having 76% MUFA and 7% PUFA, the kinematic viscosity temperature relationship (correlation coefficient 0.82 to .90) is as follows:

Table 3: Kinematic Viscosity Temperature Relationship

Temperature (°C)	40	50	65	80	95
Viscosity [mm^2/s]	39.92	27.18	18.07	12.57	9.45

By extrapolation the results form (Fasina et al., 2006) we can predict HOSF's kinematic viscosity and temperature relationship as follows:

Table 4: Kinematic viscosity and temperature relationship

Temperature (°C)	40	50	65	80	95
Viscosity [mm2/s]	39.53	26.91	17.89	12.45	9.36

This is in a good correlation with HOSF's kinematic viscosity measured at 40°C of 39.42 to 39.63 [mm^2/s]

For comparison, nr.2 Diesel fuel's kinematic viscosity varies from approx. 2.61 mm^2/s at 40°C to approx. 0.86 mm^2/s at 95°C as determined according to ASTM D 445 procedure.

For oil cooled engines specific to some off-road applications, the coolant temperature could be as high as 140°C thus the minute fuel volume confined in the injector's body and ready to be injected will reach a temperature of at least 150°C. Under those conditions, for instance Diesel fuel kinematic viscosity could be not sufficient to assure adequate lubrication and minimal leakages in the injector's needle body-assembly.Taking into account that at 95 °C HOSF's viscosity is of the same order or magnitude as of the Diesel fuel at 40 °C it is conceivable that under those circumstances and from hydraulic viewpoint only, we assume that HOSF is a viable diesel fuel alternative. Beside viscosity, fuel density and bulk modulus of elasticity are assumed to have an important role in fuel injection timing and fuel metering.

Density of vegetable oils (VO) is decreasing linearly with increasing temperature. In general, vegetable oils have molecular chains mostly of 18 carbons, thus for all practical purposes their density and its variation with temperature are pretty much the same. According to data available in the literature (Davis et al., 2008) for VOs, HOSF' density can be predicted as follows:

Table 5: HOSF' Density Versus Temperature

Temperature (°C)	15	40	50	65	80	95
Density [kg/m3]	914	896	891	883	870	862

This is in a good correlation with HOSF's density of 915.0 to 915.8 kg/m3 measured at measured at 15°C.

For comparison, diesel fuel's density varies from approx. 893kg/m^3 at 40°C to approx. 849 kg/m^3 at 95°C.

Bulk modulus of HOSF is higher then that of diesel fuel. It decrease with rise of temperature but it increases with increase of pressure and it is by far more dependent of VOs' air content than fatty acids profile. From (Varde, 1984) for VO at 25°C, bulk modulus of elasticity dependence of pressure is as follows:

Table 6: Bulk Modulus of Elasticity Dependence of Pressure

Pressure [MPa]	25	50	100	150
Bulk modulus [MPa]	1900	2100	2500	2750

From (Varde, 1984) for VO at 100 MPa, bulk modulus of elasticity dependence on temperature is as follows:

Table 7: Bulk Modulus of Elasticity Versus Temperature

Temperature °C	40	50	65	80	95
Bulk modulus [MPa]	2370	2342	2076	1884	1796

For comparison diesel bulk modulus has approx.1550 MPa at 40°C and approx.1.250 MPa at 95°C. AMESim (Advanced Modeling Environment for performing Simulations) is a 1D lumped parameter time domain simulation platform for engineering systems. AMESim uses symbols to represent individual components within the system which are either: based on the standard symbols used in the engineering field such as ISO symbols for hydraulic components or block diagram symbols for control systems or when no such standard symbols exist, symbols which give an easily recognizable pictorial representation of the system. The fuel specific values for viscosity, density and bulk modulus of elasticity were introduced as parameters into the fluid module of the AMESim injection simulation software to determine injection timing and injection fuel rate of HOSF and diesel fuel at low (40°C) and high fuel temperatures (95°C). The simulation was made for the following set of values:

Table 8: Numerical Simulations Parameters

	HOSF fuel		**Diesel fuel**	
Temperature (°C)	40	95	40	95
Viscosity [cSt]	39.92	9.45	2.61	0.86
Density [g/cm3]	0.896	0.862	0.893	0.849
Absolute viscosity [cP]	3576	814	222	77
Bulk modulus [MPa]	2370	1796	1553	1250

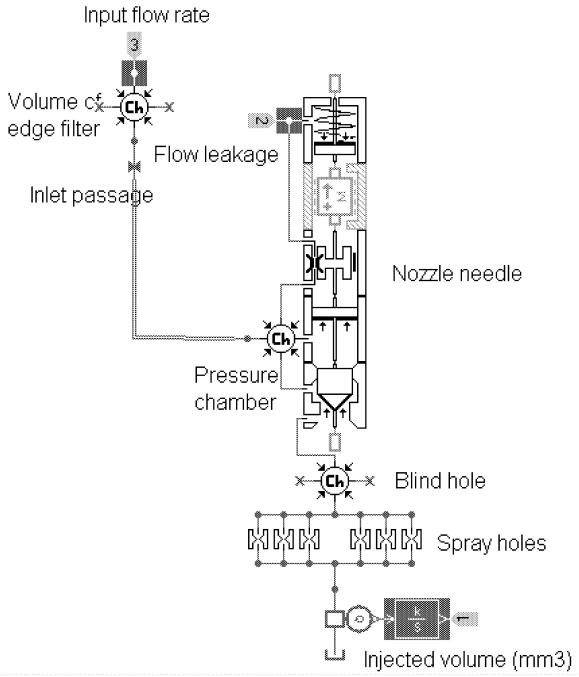
RESULTS AND DISCUSSION

The numerical simulation by AMESim (LMS International, 2009) results show substantial differences on the injection timing and rate and duration of injection that could be the root cause of catastrophic failure of diesel engines not adapted to run on straight vegetable due to late and incomplete combustion, lubricating oil dilution, plugging of injectors and piston rings sticking due to oil polymerization, low power and higher fuel consumption. However, a dramatic improvement has been observed by heating the fuel from 40°C to 95°C and it is expected that further simulations at higher fuel temperatures (150°C) will reveal that the performance will get closed to that of diesel fuel injection.

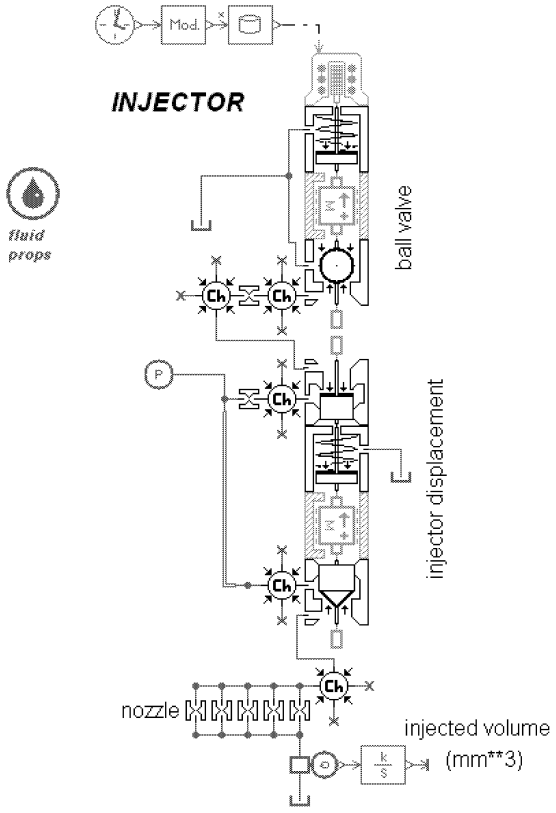
Extensive research works carried out in FLOWER POWER USA Inc., in the frame of a NSF contract. The main problem which was solved by an iterative procedure was the increase of the bulk modulus of elasticity by eliminating the micronic air bubble using a new type of ultrasonic

generator. The sun flower oil preheating by passing it through the engine lubricating system was also successfully tested.

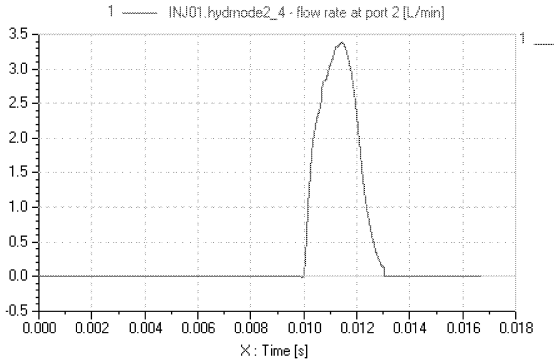
The practical conclusion of the entire research is the great time and financial efforts avoided by using AMESim.



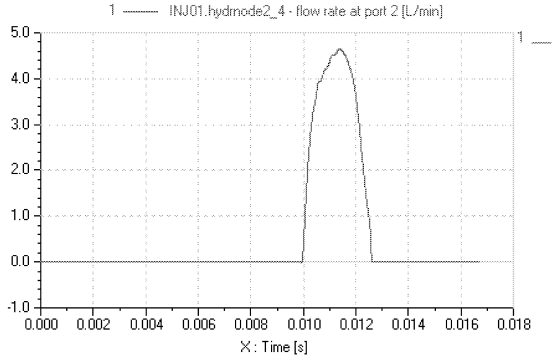
Figures 1: AMESim model for mechanical injector



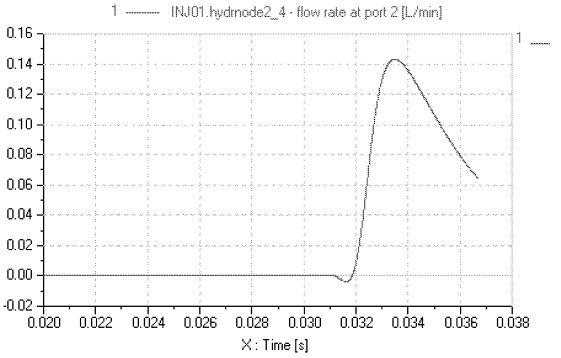
Figures 2: AMESim model for common rail (CR) solenoid actuated injector



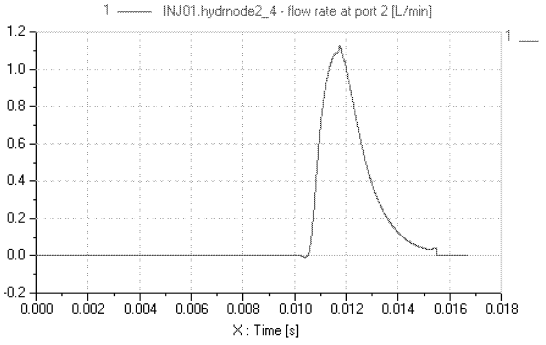
Figures 3: 40°C diesel fuel mechanical injection flow rate



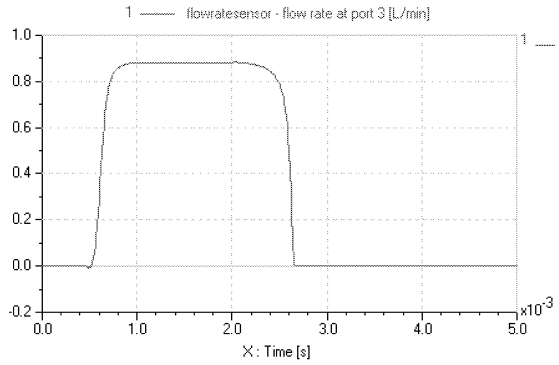
Figures 4: 95°C diesel fuel mechanical injection flow rate



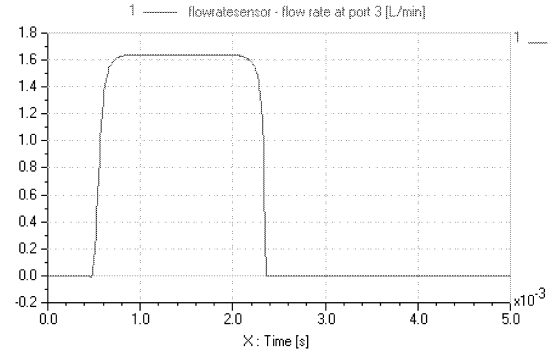
Figures 5: 40°C HOSF fuel mechanical injection flow rate



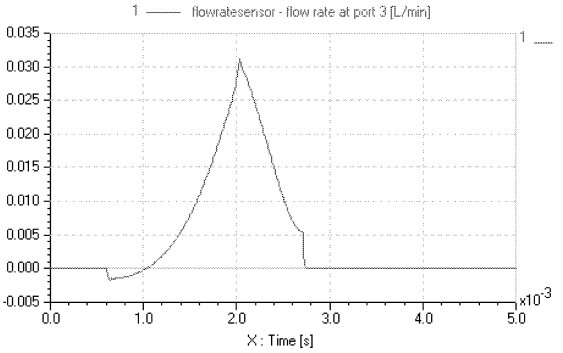
Figures 6: 95°C HOSF fuel mechanical injection flow rate



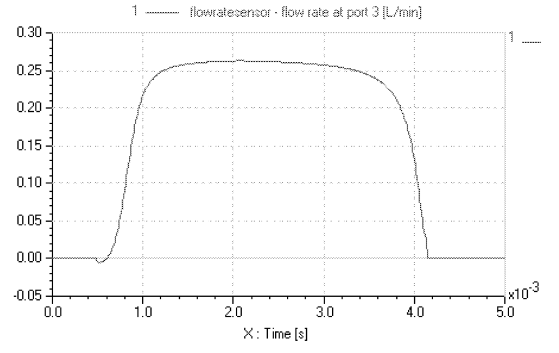
Figures 7: 40°C diesel fuel solenoid CR flow rate



Figures 8: 95°C diesel fuel solenoid CR flow rate



Figures 9: 40°C HOSF fuel solenoid CR flow rate



Figures 10: 95°C HOSF fuel solenoid CR flow rate

REFERENCES

Rauber, M., Russ, W., Winthuis, N., Werkmeister, R., Meyer-Pittroff, R. 2008. "The influence of temperature and mixing ratio on the kinematics viscosity of mixtures from rape seed oil and diesel fuel." *The International Conference on Hydraulic Machinery and Equipments,* October 16-17, Timisoara, Romania.

Ozaktas, T., Baris Cigizoglu, K. 1997. "Alternative Diesel Fuel Study on Four Different Types of Vegetable Oils of Turkish Origin." *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects,* 19, pp. 173 - 181, 01 Feb.

Hersey, M. D. 1929. "Viscosity of Diesel fuel oil under pressure". *National Advisory Board for Aeronautics*. Technical Note no. 315, Sept. 3rd.

Rodenbush, C.M., Hsieh, F.H., Viswanath, D.S. 1999. "Density and Viscosity of Vegetable Oils." *JACOS,* Vol. 76, no. 12.

Fasina, O.O., Hallman, H., Vraig-Schmidt, M. and Clements C. 2006. "Predicting Temperature - Dependence Viscosity of Vegetable Oils from fatty Acid Composition." *JACOS,* Vol. 83, no. 10.

Noor Azian, M., Mustafa Kemal, A.A., Panau, F., Ten, W.K. 2001. "Viscosity Estimation of Triglycerols and Some Vegetable Oils, Based on Their Triglycerols Composition." *JAOCS,* Vol. 78, no. 10.

Diesel, R. 1984 (1912). "The Diesel oil-engine." *Engineering,* 93, pp. 395-406 (1912), Chem. Abstr., 6.

Diesel, R. (1912). "The Diesel oil-engine and its industrial importance particularly for Great Britain." Proc. Inst. Mech. Eng., pp.179-280 Chem. Abstr., 7, 1605 (1913).

Diesel, R. 1913. "Die Entstehung des Dieselmotors. Verlag von Julius Springer." Berlin, p. 115.

Davis, J.P., Dean, L.O., Faircloth, W.H., Sanders, T.H. 2008. "Physical and Chemical Characterization of Normal and High-Oleic Oils from Nine Commercial Cultivars of Peanut." *JAOCS,* 85, pp. 235-243.

Varde, K.S. 1984. "Bulk modulus of vegetable oil-diesel fuel blends." *Fuel,* Vol. 63, May.

Yang, H., Briker, Y., Szynkarczuk, R. and Ring, Z. "Prediction of Density and Cetane Number of Diesel fuel from GC-FIMS and Piona Hydrocarbon Composition by neutral network." *The National Centre for Upgrading Technology,* Devon, AB, Canada, T9G 1A8.

LMS International. 2009. "LMS Imagine.Lab-AMESim R8.2b." Roanne.

BIOGRAPHY

Nicolae Vasiliu was born in Buzau, Romania, in 1946, and graduated in Hydropower Engineering from Polytechnical University of Bucharest in 1969. He became a Ph.D. in Fluid Mechanics at the same university, after a research stage in Gand State University and Von Karman Institute from Bruxelles. He is state professor in the Polytechnical University of Bucharest from 1994, as the head of the Fluid Control Laboratory from Power Faculty. He worked always for the industry, as project manager or scientific advisor. In 1980 he joint the Hydraulic Control Team from the Romanian Aerospace Institute. He is working mainly in modelling, simulation and dynamic identification of the hydraulic and electrohydraulic control systems. He joined LMS International) Company from Belgium for scientific consultance in Real Time Simulation.

DEVS DIAGRAM REVISED: A STRUCTURED APPROACH FOR DEVS MODELING

Hae Sang Song
Dept. Computer Engineering
Seowon University
Cheongju, 361-742
Korea
hssong@seowon.ac.kr

Tag Gon Kim
Dept. Electrical Engineering
KAIST
Daejon, 305-701
Korea
tkim@ee.kaist.ac.kr

KEYWORDS
DEVS Diagram, Ports, Variables, Phase Transition Diagram

ABSTRACT

Discrete Event Systems Specification (DEVS) formalism has been used in recent decades for modeling and simulation of discrete event systems as well as for some hybrid systems. It is because the formalism has a sound semantics for modular hierarchical modeling and good simulation development environments. However, there have been still deep gaps between understanding the formalism and applying to the practical modeling of complex systems, as the numbers of events and states of such systems are so huge to handle in the classical DEVS formalism. To solve this problem, this paper proposes the DEVS diagram, a structured diagramic form of the DEVS formalism, in both graphical notations and mathematical formulations. For this, the essential notions of state variables and phases are introduced for structuring sequential states. The concepts of ports and messages for structuring sequential events are proposed as well. Finally, the phase transition diagram is formally defined to simply represent the state transitions. A simple example illustrates the modeling method in the DEVS diagram.

1. INTRODUCTION

Demands for Modeling and Simulation (M&S) has grown in recent decades in the domain of man-made complex discrete event systems such as war games, communication networks, semiconductor fabs and so on. Discrete Event Systems specification (DEVS) for such simulation applications have become more prolific, because DEVS has a sound mathematical semantics for modular hierarchical modeling and good simulation development environments (Kim 2010). However, a fresh M&S engineer still has been experienced difficulty in applying the classical DEVS formalism to the practical modeling of such complex discrete event systems. The difficulties arise from mainly two reasons. First, the formalism itself is too mathematical for practical use and the behavior is distributed into four separated functions; thus, no integrated picture of a model is provided for the modeler to model and understand intuitively. Second, in reality, the number of (sequential) states or events are too large, because of the complexity of systems under consideration.

For these reasons, we need a more structured form of states and events to deal with those complexity in M&S. Among DEVS-based simulation development environments, DEVSim++ (Kim 1992) introduces the notions of messages and phases rather than events and states. Nevertheless, the environment does neither fully implement these concepts nor explicitly fomalize this idea. Moreover, in the real-world, a few informal graphical modeling notations for DEVS have been utilized as an essential step before translating the graphical models into corresponding mathematical DEVS models. There has been, however, no research on *formally* defining a graphical language conforming to the DEVS formalism in higher level modeling.

There have been a few efforts on how to represent DEVS models in easier ways rather than in the mathematical one. Most of them are either graphical approaches or language based approaches. The original version of DEVS diagram revised here is depicted at an example of RT-DEVS model (Hong and Song et al. 1997). The diagram notation with enhanced expressiveness has been used for a large number of commercial projects for large sized defense modeling simulation fields (Kim et al. 2010). GGAD (Generic Graphical Advanced environment for DEVS Modeling and simulation) is a tool that adopts the diagram but with somewhat different appearance (Moallemi and Wainer 2010). Language approach for DEVS modeling is also presented by a work (Hong and Kim 2006). An extension of UML for DEVS is proposed as a SysML/DEVS profile (Nikolaidou et al. 2008). Also the DEVS standardization group has tried to make standards for the exchangibility of DEVS models. However, we think that DEVS has to have its original diagram for modeling standard, rather than depending on existing one like UML to best express the mathematical model. It should also support necessary new notions for modeling complex systems, such as co-modeling (Kim 2006) and structuring states and events. Thus, this paper makes an effort to revise the original diagram (Hong and Song et al. 1997) as well as to support the mathematical foundation of the diagram.

Consequently, this paper attempts to clearly define the graphical language for DEVS implementation models, called the DEVS diagram. The diagram playes a key role to narrow the gaps of mathematical models in the classical DEVS formalism and the implementation models in the graphical language. For these, it is required to introduce the notions of ports and messages as a structured form of events, and that of state variables and phases for a structured form of sequential states as well. Finally, the concept of phase transitions is also proposed to simplify the state transitions. These efforts would make it easier for M&S engineers to

model and implement complex discrete event systems based on the DEVS formalism.

This paper is organized as follows. The next section introduces the classical DEVS modeling background. Then, we propose the definition of the DEVS diagram. A simple example illustrates the modeling approach. Finally, we conclude the discussion.

2. BACKGROUND

2.1 The Classical DEVS Formalism

As well-known, the classical DEVS formalism can specify a system with two types of models: one for describing the behavior of a basic component, and the other for the overall structure of components interconnected together to form a system. An atomic DEVS formalism describes the behavior of a basic component, which consists of three sets and four functions.

$$\mathrm{AM} =< X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta >,$$

where

X: input event set,

Y: output event set,

S: sequential state set, and total state set is defined by $\mathrm{Q} = \{(\mathrm{s,e}) \mid \mathrm{s} \in \mathrm{S}, 0 \le \mathrm{e} \le \mathrm{ta(s)}\}$,

$\delta_{\mathrm{ext}}: \mathrm{Q} \times \mathrm{X} \to \mathrm{Q}$: external transition function, for $\delta_{\mathrm{ext}}(\mathrm{s,e,x}) = (\mathrm{s'}, \mathrm{e'}), \mathrm{e} < ta(\mathrm{s}),\ \mathrm{e'} = 0$,

$\delta_{\mathrm{int}}: \mathrm{Q} \to \mathrm{Q}$: internal transition function, for $\delta_{\mathrm{int}}(\mathrm{s,e}) = (\mathrm{s'}, \mathrm{e'}), \mathrm{e} = \mathrm{ta(s)},\ \mathrm{e'} = 0$,

$\lambda: \mathrm{Q} \to \mathrm{Y}$: output function, for $(\mathrm{s,e}) \in \mathrm{Q}, \mathrm{e} = \mathrm{ta(s)}$

$\mathrm{ta}: \mathrm{S} \to \mathrm{R}_0^+$: time advance function, and

R_0^+ is the non-negative real number set.

Note that there are two types of state transitions of a model: 1) external transitions caused by external events; and 2) internal transitions in the case of no event occurrence until current state's time advance has elapsed. In the latter case, just before the internal transition, an output event is producted at the state, as specified by the output function. In an analogy to the continuous systems, the external transitions of DEVS atomic model correspond to the input-driven state transition of a continuous system and internal transitions the input-free state transitions.

The coupled DEVS formalism specifies the structure of discrete event systems composed of components communicating with each other through event couplings. A coupled model is specified as follows.

$$\mathrm{DN} =< X, Y, M, EIC, EOC, IC, SELECT >,$$

where

X: input event set,

Y: output event set,

M: component model set, either atomic models or coupled models,

$\mathrm{EIC} \subseteq \mathrm{DN.X} \times \bigcup_{\mathrm{i}} \mathrm{M_i.X_i}$: external input coupling relation,

$\mathrm{EOC} \subseteq \bigcup_{\mathrm{j}} \mathrm{M_j.Y_j} \times \mathrm{DN.Y}$: external output coupling relation,

$\mathrm{IC} \subseteq \bigcup_{\mathrm{j}} \mathrm{M_j.Y_j} \times \bigcup_{\mathrm{i}} \mathrm{M_i.X_i}$: internal coupling relation,

$\mathrm{SELECT}: 2^{\mathrm{M}} - \varnothing \longrightarrow \mathrm{M}$: select function.

Notice that the coupled DEVS formalism above has the closure property; that is, a coupled model may contain other coupled models as well as atomic models inside. The coupled models captures the structure of a system with the components hierarchy and the interfaces between components. The SELECT function resolves the simultaneous scheduling problem of simulation, by arranging the priorities of the components to be scheduled at the same time.

2.2 DEVS Graph

In a graphical way, the behavior of an atomic model $\mathrm{M} =< X, \mathrm{Y}, \mathrm{S}, \delta_{ext}, \delta_{int}, \lambda, ta >$ has been depicted in a *DEVS graph*:

$$\mathrm{AG} =< \mathrm{N}, \mathrm{E} >,$$

where

$\mathrm{N} = \mathrm{S}$: the same as the sequential state set of M,

$\mathrm{E} \subseteq \mathrm{N} \times (\mathrm{X} \bigcup \mathrm{Y} \bigcup \{\varepsilon\}) \times \mathrm{R}_0^+ \times \mathrm{N}$, ε is the null event, where for an external transition $(s, x, e, s') \in \mathrm{E}$, $x \in \mathrm{X}, 0 \le \mathrm{e} < ta(\mathrm{s}), \delta_{ext}(s, e, x) = s'$, and for an internal transition $(\mathrm{s,y,e,s'}) \in \mathrm{E}, y \in \mathrm{Y}, \delta_{int}(s, e) = s', \lambda(s) = y, e = ta(s)$.

Note that, by the definition of the atomic DEVS formalism, there is only one internal transition at a state $\mathrm{s} \in \mathrm{S}$. The null event ε is a pseudo-event that represents no event occurrence at a time.

Figure 1 shows the notation in graphical form. Subfigure (a) depicts the atomic DEVS model behavior in graphical form. At the initial state (s_0,0), if there is no event (ε) until elapsed time e, the total state becomes (s_0,e). If there has been no event until (ta(s_0)-e) elapses from (s_0,e), an internal transition occurs to reach state (s_1,0); or, if an input event x arrives at that total state (s_0,e), an external transition takes place to the state (s_2,0). For simplicity, this behavior can be depicted as in (b). Furthermore, it becomes a more compact form as in (c). Thus, the original DEVS graph can also be represented by $\mathrm{AG'} =< N', E' >$, where $\mathrm{N'} \subseteq \mathrm{S} \times \mathrm{R}_0^+$, and $\mathrm{N'} = \{(\mathrm{s,r}) | \forall \mathrm{s} \in |\mathrm{S}, \mathrm{r} = \mathrm{ta(s)}\}$, $|\mathrm{N'}| = |\mathrm{S}|$ and $\mathrm{E'} \subseteq \mathrm{N'} \times (\mathrm{X} \bigcup \mathrm{Y}) \times \mathrm{N'}$ as in (c). If no time advance is specified at a state, it is assumed to be infinity (waiting forever at that state). Note that, by definition, there is at least one internal transition defined at a node.

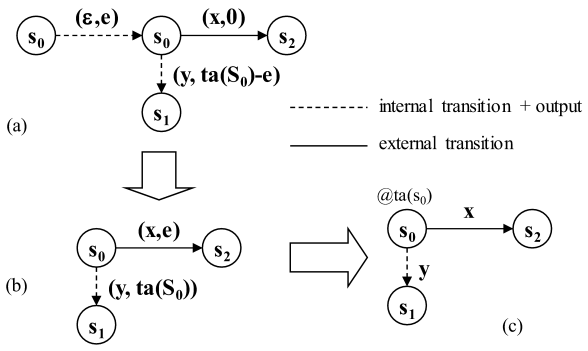


Figure 1: DEVS Graph Notation

As both atomic and coupled models have their external interfaces sets X and Y, we represent a model *m* as a box B_m

with event interfaces $B_m.I$ = X ⋃ Y on the border. We denote by $B_m.X$ the set of input event interfaces set of box B_m. Now a coupled model N =< X, Y, M, $EIC, EOC, IC, SELECT$ >can be represented in a graphic form in a straightforward manner; a coupled box model of model N is represented by B_N = <B_M, C> where B_M is the set of component box models B_M = $\{B_{M_i} | M_i \in M\}$, and $C \subseteq B.I \times B.I, B = B_M \cup \{B_N\}$ is the arc set, each arc of which links an interface on a box to other interfaces as specified in N as EIC, EOC, and IC. Figure 2 is an example of the box representation of a coupled model M12, where EIC={(M12.in, m1.x1)}, EOC = {(m2.out1,M12.out)}, and IC={(m1.out, m2.in), (m2.out2, m1.x2)}. Note that model m1 is in reality an instantiation of model M1 and m2 M2. The SELECT function will be described in a list of maps in text format {m1, m2,m3} → m1, elsewhere in the graph.

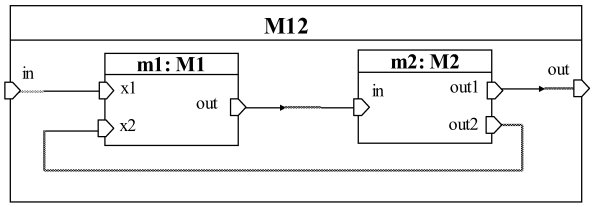


Figure 2: Coupled DEVS Graph Notation

2.3 Problem Statement

The DEVS formalism and the DEVS graph can specify discrete systems in a sound modular and hierarchical manner in system theoretical form to be adequate for logical analysis. However, in our experience, the DEVS graph of the classical DEVS formalism is too low-level representation for the modeling and implementation of systems simulation. Thus, a higher level of model representation has often been used in ad-hoc graphical forms in war game modeling simulations in the design stage (Kim 2010). However, to authors' best knowledge, there has been no explicit literature dealing with a higher level graphical representation of the DEVS formalism in structured form. Therefore, this paper focuses on explicitly defining a higher level diagram, called the *DEVS diagram*, which is a structured form of DEVS graph presented earlier. For this purpose, we introduce the notion of port and message as a structured form of events, and that of state variable and phase for a structured form of sequential states to handle complex discrete event systems. This paper focuses on presenting the DEVS diagram in graphical notations as well as mathematical forms.

3. STRUCTURING EVENTS AND STATES

The term 'structuring' means doublefold: one for state structuring and the other for event structuring. the event will be structured to 'ports' and the sequential states structured to 'variables' and 'phases'

3.1 Structuring Events: Port and Message

Descrete event systems interact with each other by sequential events in the classical DEVS formalism. In real-systems modeling, however, the number of events are so huge and sometimes infinite, which we cannot treat it easily by directly applying the DEVS formalism in modeling. Thus, one way of coping with this problem is to group these sequential events into related categories to handle it easier way.

In practical applications, there is a collection of an equivalent class of events, in the sense that these events influence from the same source model to the same destination model. Those events can be regarded as the same class or a group. The equivalent class of events is said to be a *message*. The flow of messages can be represented via the notions of *ports* and *channels*. A *port* is an arrival or departure place of a model. Each port has the *port type* where the messages of the same type or sub type of the port can be handled. Generally, we denote the type of a variable or a port using the operator 'domain()' or simply $dom(\cdot)$. Meanwhile, a *channel* is a connection from a port to another port. It is assumed a message at a departure port can be transferred instantly to the arrival ports through a channel. Hereafter, the input port set is denoted by $\mathbf{P_X} = \{(\mathbf{p_x}, \mathbf{d_x})\}$, where $\mathbf{p_x}$ is a port and $\mathbf{d_x}$ is the domain of the port. Similary, the output ports set is denoted by $\mathbf{P_Y} = \{(\mathbf{p_Y}, \mathbf{d_Y})\}$. A channel connects a source port to the target ports of the same type.

Note that an event is really an occurrence of sending or receiving a message of specific value though a channel. As the channel transmission is assumed to be instantaneous, an output event and the input event caused by a message transmission occur at the same time. In this sense, a channel can be said to be a collection of event couplings with the same source and destination models, compared to the classical DEVS formalism.

With the notions of port, message, and channel we can define a structured form of the classical DEVS formalism, which we call the *Structured DEVS Formalism*. We will not present it here due to the lack of space. Note that the structured DEVS formalism looks similar to the classical atomic DEVS formalism except that it is composed of structured states and events using the notions of port and message, and state variables, presented below.

3.2 Structuring States: State Variables and Phase

Compared to the state variables in the continuous systems specified in differential equations, the same concepts of the state variables can be applied to the discrete event systems, especially specified in the DEVS formalism. A system state can be specified by a set of system variables, each of which has its domain. In general, a set of related sequential states is usually grouped into a state variable. Formally, a *state variable* is a container that can accommodate a group of related (sequential) states. Usually, the group is called the domain of the variable. Thus, a set of system sequential states can be structured to a set of system variables. Then, a system state is a combination of values that the state variables have at some time, which is called a *composite state*. Therefore, the notion of state variables is a useful means to form a structure of the flat sequential states.

Although the notion of state variables is a good way enough to structure the sequential states, it is required to further abstract them to a higher level of states to simplify the specification of state transitions. Formally, a *phase* is a representative value of a set of equivalent composite states which produce the same output event and/or have the same time advance at the states. If we add phase variable(s) to the

system state variables set, we can simplify the state transitions even more by fewer number of phase transitions. Consider that we partition the composite state set V into equivalent classes such that each class has a set of composite states with the same time advance and/or output event. Let each class have a single representative name, called a phase name, and we can add an additive or sometimes redundant phase variable $\boldsymbol{\psi}$ to the state variable set $\boldsymbol{S_V}$; it then becomes that $\boldsymbol{S'_V = S_V \cup \{\psi\}}$. We designate a phase variable as a high-level state variable, and the original ones as the low-level state variables. Then, a state of the system becomes the combination of a phase value and a composite state of the state variables. An extremely trivial redundant case occurs when a phase has only a state as its member, a one-to-one correspondence.

A phase can be hierarchical; that is, a phase can be decomposed more into *sub-phases* having disjoint composite state members, and so on. It is always true that a union of sub-phases within a phase gives the total states set of the phase and the intersection of all sub-phases results in an empty set, i.e., for a phase variable $\boldsymbol{\psi} \in \mathbf{S_V}$, $dom(\boldsymbol{\psi}) = \{\varphi_i\}$, let $V_{\varphi_i} \subseteq \mathbf{V}$ be the phase member states set of phase φ_i, then $\bigcup_i V_{\varphi_i} = \mathbf{V}$, and, for any two phases, it should be $\varphi_i, \varphi_j \in \boldsymbol{\psi}, V_{\varphi_i} \cap V_{\varphi_j} = \emptyset$. To discriminate phase member states, we define a *guard* on the composite state set $\mathbf{V}$ as a logical expression on low-level state variables that further filters a phase into a subset of states of the phase, or a sub-phase. For a guard G we denote by $V_G = \{v \mid G(v) = \text{true}, v \in \mathbf{V}\}$ the *guard member states set*. Then, for phase $\varphi \in \boldsymbol{\psi}$ and the phase member states set V_φ, if for a guard G, $V_G = V_\varphi$ we then call G the *phase guard* and denote it by G_φ. We also call a function $A: \mathbf{V} \to \mathbf{V}$ an (state transition) *action*. We can now define the notion of a phase transition diagram.

3.3 Phase Transition Diagram

Let $\mathbf{P_X}, \mathbf{P_Y}, \mathbf{S_V}$ be an input ports set, an output ports set and state variables set of a structured atomic model. Let $\boldsymbol{\psi} \in \mathbf{S_V}$ be a phase variable where each $\varphi_i \in \text{dom}(\boldsymbol{\psi})$ has a disjoint composite member states set $V_{\varphi_i} \subseteq V$ with its unique phase guard G_{φ_i}. We then formally define the specification of the *phase transition diagram* as the following.

$$PD = < N, E >,$$

where

- $N = \{(\varphi, T_\varphi, V_\varphi) \mid \text{phase } \varphi \in dom(\boldsymbol{\psi}), \boldsymbol{\psi} \in \mathbf{S_V}\}$: a node set, where T_φ is the time advance of a phase φ, V_φ is the disjoint composite member states set that can be also described by a guard G_φ, $V_{G_\varphi} = V_\varphi$,
- $E \subseteq N \times (\{G_i\} \times \Sigma \times \{A_i\}) \times N$: a phase transitions set, $\{G_i\}$ is a sub-guards set, $\Sigma = \mathbf{X} \cup \mathbf{Y}$, the union of structured input / output event sets obtained from $\mathbf{P_X}, \mathbf{P_Y}$, respectively. $\{A_i\}$ is an actions set such that $A_i : \mathbf{V} \to \mathbf{V}$,

with three constraints for any two phase transitions on a phase,

1) (functionality) if the two events are the same, then the guards are disjointed with each other, i.e., $V_{G_i} \cap V_{G_j} = \emptyset$, and
2) (uniqueness) if the two events are different output events, then the guards are disjointed with each other, and
3) (integrity) the target state which is caused by the action of a phase transition is a member of the target phase members identified by the target phase guard for integrity.

A phase transition occurs by a **guard/event/action** pair. In essence a phase transition is a collection of sequential state transitions of equivalent sequential states. Formally the source states of two state transitions are said to be equivalent if the state sojourn time is the same; and the event; and destination states reached by the action are equal. Figure 3 illustrates the notion of phase transition; there are three phases $\varphi_1, \varphi_2, \varphi_3$ where φ_1 is further decomposed by guards on φ_1, G_1, G_2, G_3 such that $G_1(\varphi_1) = \varphi_{11}$, $G_2(\varphi_1) = \varphi_{12}$, $G_2(\varphi_1) = \varphi_{13}$. The states of sub-phase $\varphi_{11} = \{s_1, s_2, s_3\}$ that are all mapped to state s_9 of phase φ_2 by event $x_1?m_1$ are equivalent by definition. Conversely it can be regarded that we group these three sequential equivalent state transitions into a phase transition with a guard $G_1(\varphi_1) = \varphi_{11}$, an event $x_1?m_1$, and an action $A_1(\mathbf{S_V})=s_9 \in \varphi_2$, which is denoted by a phase transition $(\varphi_1, G_1, x_1?m_1, A_1, \varphi_2)$.

.

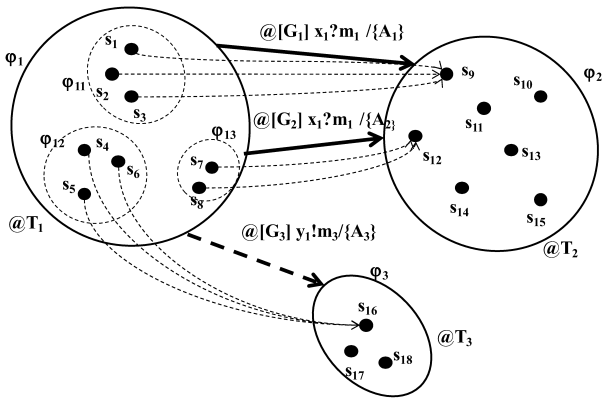


Figure 3: Illustration of the Notion of Phase Transition

We define a graphical form of a phase transition diagram as follows. A phase is represented by a rounded box with its phase name and its time advance **@T** or multiple time advances with disjoint guards upon the phase, **{@[G$_i$]T$_i$}** below the name**.** An internal phase transition is drawn as a dotted arc from a source phase to a target phase along with an annotation, ***outport!message@[guard] / {action}***. This implies that when a composite state of the source phase meets the guard *guard* and the phase time advance elapses without any external event, we take the internal transition. Before the internal transition, the model sends a message of value *message* to the output port *outport* and then change the state variable as specified by the action *action*. Likewise, an external phase transition is described by a solid arc with the annotation ***inport?message @[guard] /{action}***. The notation means that when a composite state of the source phase meets the guard *guard* and an input message arrived at the input port *inport* is the expected value specified by *message*, then take the external transition while executing the action *action*. If multiple messages cause the same action, it can then be specified by a message guard such as ***inport?[message guard] @[guard] /{action}***. If the message is not specified, it

means a null message. If the guard is not specified, then it means the transition applies to all states of the source phase. In case that the action is not specified, it implies that only the phase variable changes are specified in the diagram. Finally, it is assumed that the phase guards are described implicitly or explicitly somewhere elsewhere in the diagram if a phase node has a small space.

Let us take a look at variables in more detail. Recall that a guard is a logical expression on state / phase variables which results in a subset of composite states that meet the guard. For practical usage, we can divide the state variables into two categories: those that affect state transition directly; and the others that are irrelevant to any state transition in any way. The former are called the primary state variables which appear on the guard, and the latter user variables are those that any guard does not care about, though the user variables may be changed by the action. It is important to discriminate whether a state variable is a primary or a user variable in modeling an atomic model since the complexity of modeling can be reduced.

4. THE DEVS DIAGRAM

In general, for platform engineers, sometimes a graphical language-based approach is preferred to the equivalent mathematical one, especially when modeling and design stage in the system simulation development. it is partially because we can easily capture the intuitive picture of a system under consideration. As noted before, the DEVS graph for the classical DEVS is not adequate to take a picture of a complex system due to the huge number of states. The DEVS diagram proposed and refined here is another instrument for modeling such a system in the structured form. The notion of the phase described above is an essential tool for structuring and abstracting state transitions while that of the state variable is essential for sequential states and that of the port for sequential discrete events. As described earlier, a phase variable containes a set of phase values, each of which represents a subset of composite states. We can now define the DEVS diagram formally.

4.1 Atomic DEVS Diagram

The DEVS diagram consists of descriptions of an atomic DEVS diagram and a coupled DEVS diagram. An atomic DEVS model can be specified as an *atomic model box* with input and output ports on the border and the model name at the top; a variable box inside the model box has a list of state variables with their initial values, and most importantly a phase transition diagram. Formally, an atomic DEVS diagram of atomic model M is specified by

$$\mathrm{ADD_M} =< \mathrm{AB_M}, \mathrm{P_M} >,$$

where

$\mathrm{AB_M} =< \mathbf{P_X}, \mathbf{P_Y}, \{\mathbf{v}: \mathbf{d_v} = v_0\} >$: an atomic model box of name M, with ports on the box boundary, where an input(output) port is depicted by an inward(outward) box-arrow annotated by ***port:type***, $\mathbf{p_x}: \mathbf{d_x} \in \mathbf{P_X}, (\mathbf{p_y}: \mathbf{d_y} \in \mathbf{P_Y})$, respectively. A variable box inside the model box lists the variables, each variable of which is described by $\mathbf{v}: \mathbf{d_v} = v_0$, variable name $\mathbf{v}$, domain of the variable $\mathbf{d_v}$ and an initial value v_0,

$\mathrm{P_M} =< N, E >$: a phase transition diagram inside the model box.

Figure 4 shows an example of an atomic DEVS diagram whose model name is 'ABuffer'; it has two input ports named 'in' with type 'Job' and 'done' and one output port 'out' with type 'Job'. In the variable box there is a phase variable usually named 'phase' and two primary state variables, 'p' of type {B,F} and 'n' of non-zero integer. There is one internal phase transition from phase SEND to WAIT by 'out!w/{n--,p=B}', which means there is no guard, and sends a message 'w' to the output port 'out'; then change the state variables 'n' and 'p'. The default initial value of variable 'phase' is WAIT which is also bolded in the phase transition diagram. The initial values of the variable box can be changed at a coupled model specification if required. By convention prefix 'A' of a model name means an atomic model and 'C' a coupled model.

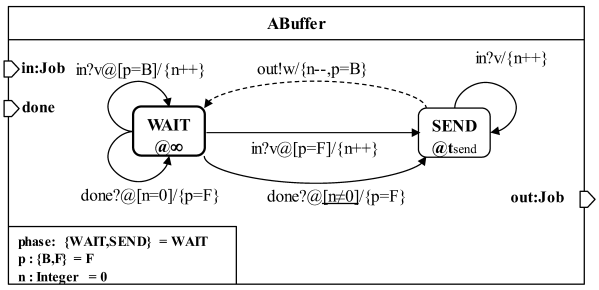


Figure 4: Example of an Atomic DEVS Diagram

In more detail the above phase transition diagram is an example of a case that can be derived from a state transition graph in Figure 5. For two state variables 'p' and 'n', *type(*p)={B,F} and *type(*n) = {0,1,2,..}, the composite states in the Figure are partitioned into two phases: WAIT={(n,p)|(n=0 and p=F) or p = B} and SEND = {(n,p)| n > 0 and p = F}. The time advance or state sojourn time at phase WAIT is infinite and that of SEND is t_{send}. At phase WAIT if a message 'v' arrives at port 'in' at a sub-phase by identified guard @[p=B] (left top) then it returns to phase WAIT taking action {n++} or for the same event at a sub-phase @[p=F] (center), then it goes to phase SEND with doing action {n++}. The phase variable 'phase' is implicitly added to the system variables and the change of the phase variable is implicitly assumed.

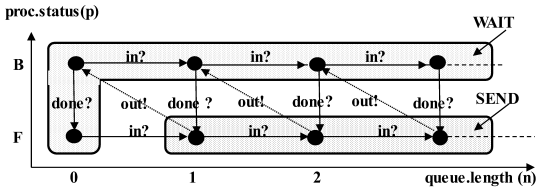


Figure 5: Mapping from a State Transition Graph

Conversely we can extract a state transition graph from a phase transition diagram. Note that the phase transition diagram above is much simpler than the state transition graph below; this is the reason why we prefer the phase

transition diagram to the state transition diagram of the DEVS graph.

We remark that, although there is usually one phase diagram for an atomic model, there may be two or more phase transition diagrams in special cases in which the state variables can be grouped into two or more and the groups are independent of each other. By the term 'independent', we mean any composite state of a group will never affect any state transition of the other group. A phase may also have the closure property; that is, a phase can have a phase diagram in it with its sub-phases. The former case is a kind of vertical partition of state variables and the latter is a horizontal partition of a phase. For example, in Figure 5, the phase WAIT can be divided further into BUSY_WAIT = WAIT ∩ {(n,p)|p=B} and FREE_WAIT = WAIT ∩ {(n,p)|p=F} = {(0,F)}. Then some of the guards in Figure 4 become unnecessary; so we can obtain a revised phase diagram, as in Figure 6.

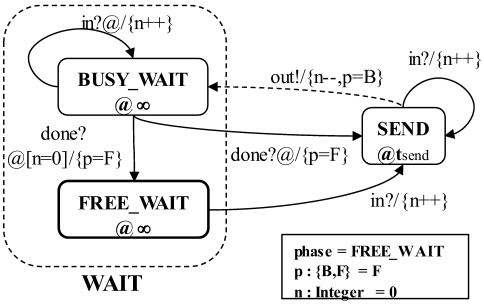


Figure 6: Phase Partitioning

An example of variable partitioning is illustrated in Figure 7. To illustrate this notion, add a counter variable 'c,' accumulating up the number of inputs from port 'in'. We can add this variable and a new phase diagram with a new phase variable 'phase2,' while the rest of the model is the same as before, for the counter variable does not affect the existing phase diagram at all.

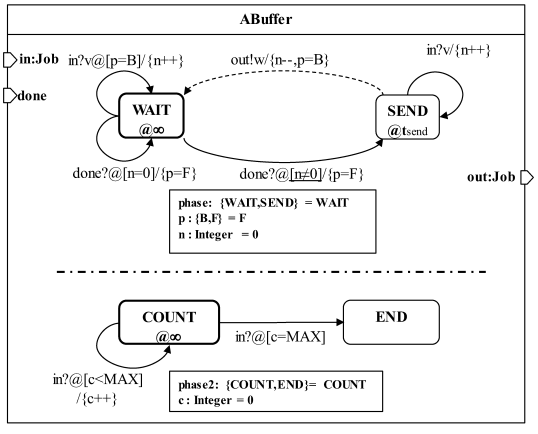


Figure 7: Parallel Phase Diagrams

This parallel modeling has advantages in its simplicity and comprehensiveness compared to an equivalent composite phase diagram. However we need take care of the time advance in the implementation stage. The time advance of the parallel phase diagrams should be the minimum of the remained time advances of the current phases, which should be managed by the developer in the time advance function. We recommend the developer that it is easy this model be implemented in two atomic models with the same interface, for the time management is delegated to the simulation engine.

On the other hand the methodology of collaborative modeling or co-modeling has had a good impact on multi-party M&S projects. The DEVS diagram supports this methodology in a simple way: 1) replace a state variable with an object, and 2) replace the action related to the state variable with the methods of the object. Recalling the Figure 4 of the atomic buffer model, we can know that the state variable 'n' needs to be more refined, and we design a corresponding co-object 'q' of type 'Queue' instead of queue length 'n'. Related actions of the variable 'n' are either increased (refined to insert) or decreased (refined to delete); and, a required guard is to quest the length to the co-object 'q'. Furthermore we can add the input message type 'Job' for the port 'in'. Then we can obtain a refined and detailed atomic model based on the co-modeling methodology as shown in Figure 8. Action 'n++' is replaced by q.insert() , where the object 'q' will increase the queue length. The queue length can be queried by q.length() operation. Note that the input event is denoted by 'in?v' where 'v' is a temporary variable storing the job from input port 'in'. In this way, we can gradually refine a very abstract atomic model with only state variables at first to a detailed model with corresponding co-objects. Figure 9 is a sketch of simulation software for the model specification in Figure 8.

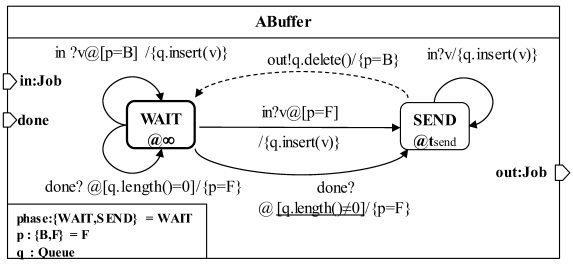


Figure 8: Refining with Co-modeling Methodology

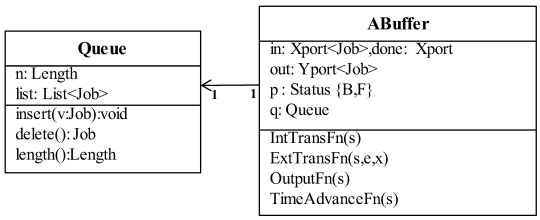


Figure 9: Sample Software Design of the Buffer Model

4.2 Coupled DEVS diagram

Formally, a coupled DEVS model N is specified by a coupled DEVS diagram,

$$\mathrm{CDD_N} =< \mathrm{B_N}, \{\mathrm{B_{m:M}}\}, \mathrm{C_N} >,$$

where

$\mathrm{B_N} =< \mathbf{P_X}, \mathbf{P_Y} >$: a outmost coupled model box of name M with ports on its boundary, where an input(output) port is depicted by an inward(outward) box-arrow

annotated by ***port:type***, $\mathbf{p_x}: \mathbf{d_x} \in \mathbf{P_X}, (\mathbf{p_y}: \mathbf{d_y} \in \mathbf{P_Y})$, respectively,

$\{B_{m_i:M_i}\}$: a set of model object boxes located inside the coupled model box B_N, where a model object is denoted by $m_i: M_i$, a combination of object name 'm_i' of model 'M_i',

$C_N =< C_{EIC}, C_{EOC}, C_{IC} >$: the channel specification, where a solid line connects each port pair in the sets,

$C_{EIC} \subseteq \{(\mathbf{p_x}, \mathbf{p'_x}) | \mathbf{p_x} \in B_N.\mathbf{P_X}, \mathbf{p'_x} \in B_{m_i:M_i}.\mathbf{P_X}$ for any component model object $m_i: M_i\}$,

$C_{EOC} \subseteq \{(\mathbf{p_Y}, \mathbf{p'_Y}) | \mathbf{P'_Y} \in B_N.\mathbf{P_Y}, \mathbf{p_Y} \in B_{m_i:M_i}.\mathbf{P_Y}$ for any component model object $m_i: M_i\}$,

$C_{IC} \subseteq \{(\mathbf{p_Y}, \mathbf{p_x}) | \mathbf{p_Y} \in B_{m_i:M_i}.\mathbf{P_Y}, \mathbf{p_X} \in B_{m_j:M_j}.\mathbf{P_X}$, for any i, j$\}$.

The couple model box B_N is similar to the atomic DEVS model box except for the variable box omitting. Component object boxes are to be placed inside the coupled model box. Recall that a model object differs from a model for it is an instantiation of a model. A model object box is the same as the model box except for the box name of the form of ***object:Model*** which represents a model object '***object'*** , an instantiation of the model '***Model***.' A channel is drawn with a solid line from a source port to the destination port.

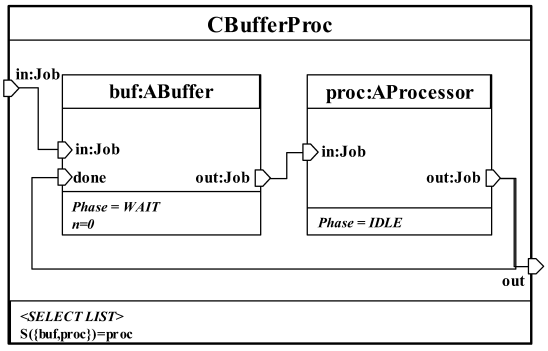


Figure 10: Example of Coupled DEVS Diagram

Figure 10 illustrates the notation by a coupled model named 'CBufferProc'. The type of port 'in' of the coupled model is 'Job'. There are two component objects: 'buf' of model 'ABuffer' and object 'proc' of model 'AProcessor'. There are four channels: (CBufferProc.in, buf.in), (buf.out,proc.in), (proc.out,buf.done), and (proc.out,CBufferProc.out). The SELECT function is described as a list at the lowest box.

5. CASE STUDY: A GBP MODEL

Using the DEVS diagram defined above, we model a simple but full simulation model called GBP (generator-buffer-processor) model. GBP model is a kind of queuing model such as bank teller model.

Figure 11 depicts the whole system model 'CGBPSim', that has two atomic models 'gen:AGenerator', 'trn:ATransducer', and a coupled model 'bp;CBufferProc' as shown in Figure 10. A generator model object 'gen' creates jobs periodically; the transducer object 'trn' manages the statistics of the simulation result by accepting completed jobs from BufferProc model object 'bp'; and, BufferProc 'bp' is a coupled model object again composed of 'buf:ABuffer' and 'proc:AProcessor'; The model 'bp' is buffering and processing the job and outputs the completed job to the output port 'out:Job'. SELECT list is not depicted here, but we assume that the priority is transitively 'trn' > 'bp' > 'gen'.

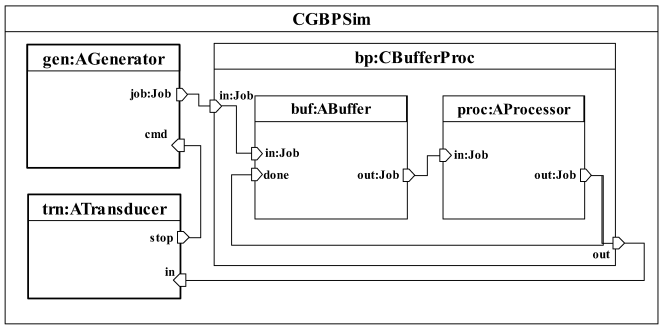


Figure 11: GBP Model in Coupled DEVS Diagram

In summary, a job message created by the generator model goes through the buffer-proc coupled model, and then finally reached to the transducer. When the transducer determines to stop simulation, it sends 'STOP' command to the generator. The ports are connected from an output port to input ports according to the model specification. Note that 'CGBPSim' is not an object but a model.

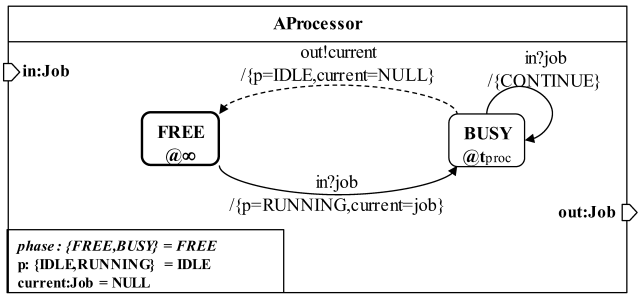


Figure 12: AProcessor: Atomic Processor Model

Figure 12 is an atomic model of the processor. It has variables of the processor status 'p: {IDLE,RUNNING}', a temporary variables storing a job message currently processed, and the phase variable with trivial two phases: FREE={(p,-)| p=IDLE}, BUSY={(p,-)|p=RUNNING}.

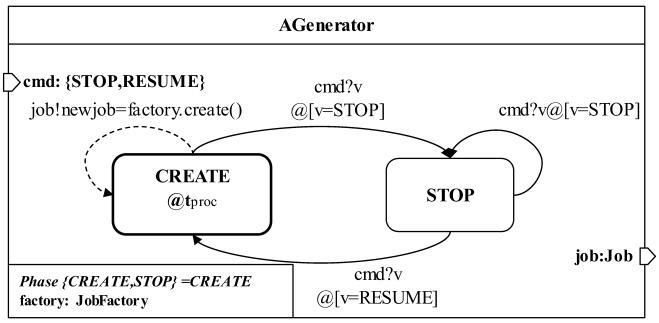


Figure 13: AGenerator: Atomic Generator Model

The generator model in atomic DEVS diagram is shown in Figure 13. It receives commands via port 'cmd', whose type is {STOP,RESUME}. The main activity of the model is to create jobs periodically. We can observe the creation activity is delegated to a co-object 'factory:JobFactory', whose

operation ‘factory. create()’ is periodically called and the resultant job is sent to the output port ‘job:Job’. At phase ‘CREATE’, if it receives a command through temporary variable ‘v’ and the value of the variable is STOP, then the model goes to the phase ‘STOP’.

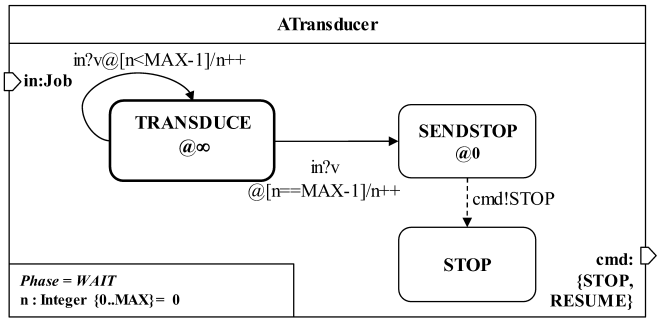


Figure 14: ATransducer: Atomic Transducer Model

Finally, the transducer model, shown in Figure 14, collects the completed jobs until the number reaches to a pre-specified maximum, and then it sends a ‘STOP’ command.

7. CONCLUSIONS

This paper proposed the DEVS diagram for practical modeling of complex discrete event systems. The main idea was to make structures of events and states by introducing the notions of port and message, and state variable and phase. Those notions group sequential states and events into ports and variables. The diagram itself has been used for years but not defined formally, lacking the mathematical foundation. Thus, sometimes the diagram models has lacked semantics, been incomplete, even violated the DEVS formalism. We did comprehensive and through work for the definition of the DEVS diagram to provide engineers with an efficient tool for modeling and design of simulation software. For lack of space, this paper can not provide the through mathematical foundation based on the structured DEVS formalism; rather, we made focus on the diagram itself. In the subsequence full paper to be submitted, we will prove that the DEVS diagram is based on the through mathematical foundation. Based on this work, we will update DEVS Specification Language in script form and plan to make a computer-aided modeling tool for DEVS-based simulation development. More work, however, is required to implement DEVS simulation tool conforming the proposed DEVS diagram. As we expect DEVS applications to become more prolific as M&S demands grow, our work will be utilized more in the domain of complex M&S.

REFERENCES

Hong, Jun S.; Hae S. Song; Tag G. Kim; and K.H. Park. 1997. "A Real-time Discrete Event System Specification Formalism for Seamless Real-time Software Development," *Discrete Event Dynamic Systems*, Vol. 7, No. 4, pp. 355 – 375.

Hong, Ki J. and Tag G. Kim. 2006. "DEVSpecL-DEVS specification language for modeling, simulation and analysis of discrete event systems," *Information and Software Technology*, Vol. 48, No. 4, pp. 221 – 234.

Kim, Jae H. and Tag G. Kim. 2006. "Parametric Behavior Modeling Framework for War Game Models Development Using OO Co-Modeling Methodology," *2006 Sping Simulation MultiConf.*, Huntsville, USA, pp. 69 – 75.

Kim, Tag G.; C. H. Sung; S.Y. Hong; J.H. Hong; C.B. Choi, J.H. Kim; K.M. Seo; and J.W. Bae. 2010. “DEVSim++ Tools Set for Defense M&S and Interoperation,” *Journal of Defense Modeling and Simulation,* Sumitted.

Kim, Tag G. and S. B. Park. 1992. “The DEVS formalism: hierarchical modular systems specification in C++,” 1992 European Simulation MUlticonference, York, United Kingdom. Pp. 152-156.

Moallemi, M. and Gabriel A. Wainer. 2010. “Designing and Interface for Real-Time and Embedded DEVS,” *Proceedings of 2010 Spring Simulation Conference*, pp.154-161.

Nikolaidou, M.; V. Dalakas; L. Mitsi; G.D. Kapos; and D. Anagnostopoulos. 2008. “A SysML Profile for Classical DEVS Simulators,” *Proceedings of 3rd Internal Conference on Software Engineering Advances, pp. 445-450.*

Song, Hae S. and Tag G. Kim. 2005. "Application of Real-time DEVS to Analysis of Safety-critical Embedded Control Systems: Railroad-crossing Control Example," *Simulation*, Vol. 81, No. 2, pp. 119 - 136.

Zeigler, B. P. and Tag G. Kim. 2000. *Theory of Modelling and Simulation (2nd Ed.),* Acedemic Press.

BIOGRAPHY

Hae Sang Song was born in Damyang, Korea, and studied his MS.D and Ph.D courses in Electrical Engineering in KAIST (Korea Advanced Institute of Science and Technology) . He worked for a couple of years in an R&D lab, IAE(Institued of Advanced Engineering) in 1999-2000. He also worked in a venture company for about two years, and has been a professor of Dept. Computer Engineering of Seowon Univisity, Korea, since 2002. He spends his sabbatical year 2010 in Systems Modeling Simulation Lab of the co-author Tag Gon Kim, KAIST, as a visiting schalor for defense M&S projects. His major interest resides in modeling simulation, analysis, and control of discrete event dynamic systems.

Concurrent Discrete Event Simulation in Java

John I. Dalseng, Senior Lecturer
Finnmark College, Alta, Norway

Keywords: discrete simulation, object-oriented, concurrency, synchronization

Abstract

A concurrent simulation technique based on the Java programming language and Java threads (lightweight processes) is introduced. The simulation events and process actions are represented by Java threads, and the simulation model structure is represented by shared variables and data structures. The threads may be executed interleaved on a single-processor machine, or in parallel on a multi-processor shared memory machine. The read and write operations on the shared variables are synchronized in simulated time by the semaphore and barrier constructs. A semaphore restricts the number of threads that can access some variable, and a barrier is a rendezvous technique used to block threads until a specific number of threads have reached the barrier.

Introduction

Multithreading allows a computer program to run concurrently. A simulation program may be divided into multiple execution threads, and the threads may share the same variables and data structures. The access to the shared variables is synchronized by semaphore and barrier constructs. We will define a simulation package based on Java® threads and the package java.util.concurrent.

The simulation model is represented by objects of the Java class Thread. Each thread defines a sequence of simulation events and actions. The threads may have shared variables and data structures, and the access to the shared variables is synchronized in simulated time.

Simulation thread synchronization mechanisms

The package java.util.concurrent defines synchronization constructs for threads [3]. The semaphore and barrier constructs are mapped into corresponding simulation constructs that operate in simulated time.

A semaphore maintains a set of permits to a resource and is used to restrict the number of threads that can access some resource. A thread must wait for a resource until the resource is available. [1], [2], [3].

A barrier is a rendezvous mechanism where a number of threads must wait until a specific number of threads have reached the barrier. The barrier is then opened and all the waiting threads may proceed. [3].

The simulation semaphore is used for mutual exclusion of threads to shared resources in simulated time. The simulation semaphore is defined by the class SimSemaphore:

```
class SimSemaphore extends Semaphore {
        SimSemaphore(int permits);
        public void acquire();
        public void release();
}
```

The class Semaphore is defined in the package java.util.concurrent. [3].

The constructor parameter int permits gives the initial number of permits, i.e. the capacity of the resource. The methods acquire() and release() controls the access to the shared resource in simulated time. The method acquire() decrements the number of permits by one. If the result is negative, the thread blocks itself and waits until a permit is released by another thread. If the release has happened before (in simulated time) the thread is blocked, the thread is unblocked and may continue immediately. If the release happens after the thread is blocked, then the event-time of the blocking thread is increased to the event-time of the release and the blocking thread resumes its actions at that time.

The acquire method may be described by the activity flow diagram shown on Figure 1.

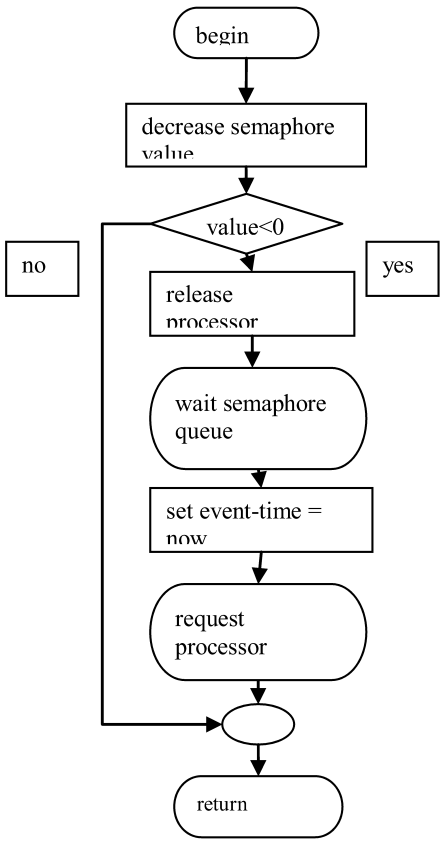


Figure 1 Acquire flow diagram

The method decrements the semaphore value by one. If the result is not negative, the calling thread returns and continues running. If the result is negative, then the logical processor is released and the calling thread blocks itself and waits in the semaphore queue until the value of the semaphore is incremented by another thread. When the thread continues, the event-time of the thread is increased to the event-time when the thread is unblocked (now). Finally, the thread requests a logical processor and waits until a processor becomes available.

The method release() adds a permit to the semaphore, and one of the pending threads which have been blocked before the release is unblocked and leaves the semaphore queue.

The barrier is defined as a rendezvous mechanism. A barrier allows a number of threads to wait at the barrier until all parties have reached the barrier. When all parties have reached the barrier, the barrier is opened and all the waiting threads are unblocked and may continue.

The simulation barrier is defined by class SimBarrier:

```
class SimBarrier extends CyclicBarrier {
        public SimBarrier (int parties);
        public void await();
}
```

The class CyclicBarrier is defined in the package java.util.concurrent. [3].

The constructor SimBarrier (int parties) creates a simulation barrier with the number of parties given by the parameter parties.

The method void await() blocks a calling thread until the prescribed number of threads have reached the barrier in simulated time. The barrier is then opened and the threads may proceed concurrently at the event-time of the latest arrival.

Simulation Thread

A simulation thread defines a sequence of events and simulation actions that happen at specific moments of simulated time. The simulation threads are defined by the following class:

```
class SimThread extends Thread {
        private Event event;
        public void hold(int t);}
```

The variable event defines the event-time of the next event belonging to the thread. The class Event is defined by:

```
class Event {
        public  CyclicBarrier schedule =
        new CyclicBarrier(2);
        private int clock = 0;}
```

The class Event contains a binary barrier called schedule, which is used to block the corresponding simulation thread until its next event actions may be processed. The barrier is opened by a Thread called Scheduler. The Scheduler is described in the next section.

The variable clock contains the next event-time of the thread. The method hold(int t) increases the clock with t units of time, and blocks the thread until the next event of the thread may safely be executed (conservative simulation approach). We may also define a method advance(int t), which may increase the clock with t units of time without blocking the thread (optimistic simulation approach).

A simulation thread may be in one of four states:

- Blocking@processor. The thread is waiting until a logical processor becomes available. When a processor is released,

one of the waiting threads is selected for execution.

- Running. The thread is busy running model state operations. The number of threads running at the same time is limited by the number of available logical processors. When a running thread is blocked by hold, await or acquire, the processor is released and another thread may get the processor.
- Blocking@simulation barrier. The thread waits until a specific number of simulation threads have reached the barrier. The barrier is then opened and all the blocking threads may proceed at the event-time of the latest arrival.
- Blocking@simulation semaphore. The thread waits until the semaphore value is positive. When the semaphore value is increased, one of the blocking threads may proceed.

The Simulation Scheduler

The execution of the simulation is controlled by a thread called *Scheduler:*

```
class Scheduler extends Thread {
        public Semaphore startEvent=
        new Semaphore(0);
        public EventList eventQueue =
         new EventList();
        public Semaphore processor =
        new Semaphore(number of processors};
}
```

The scheduler maintains a time-ordered queue of events, EventList eventQueue, and has access to a pool of logical processors, Semaphore processor. If the pool contains a processor and there is an event waiting in the eventQueue, the most imminent event, current, is selected from the queue, and initiated for execution.

The scheduler is initially blocked by the semaphore startEvent until the main program has initiated the simulation. The scheduler then tries to acquire a processor and waits until a processor is available. When the scheduler gets a processor, it selects the next event, if any, from the eventQueue, and unblocks the corresponding simulation thread (nextEvent.schedule.await()). If the eventQueue is empty, an exception may be raised

Conclusions

The aim of this study has been to investigate a simulation technique based on an object-oriented programming approach in Java and with some concurrency in mind.

We have introduced three simulation constructs:

- SimThread derived from Java Thread and representing simulation threads. Each thread keeps track of its own simulation clock. The simulation clock is increased by the operation hold(int t).
- SimSemaphore derived from Semaphore defined in java.util.concurrent. The mechanism maps the semaphore synchronization mechanism into simulations. The SimSemaphore operations acquire() and release() synchronizes concurrent simulation threads in simulated time.
- SimBarrier derived from CyclicBarrier defined in java.util.concurrent. The mechanism represents the barrier synchronization mechanism in the simulation time space. The SimBarrier operation await() synchronizes concurrent simulation threads in simulated time.

Some effort has been made to achieve syntactical similarity between a simulation program and the corresponding real-time concurrent program defined by java.util.concurrent constructs. An advantage with this approach is that the verification of simulation programs may be easier by applying Java concurrent program debugging facilities. On the other hand, testing an debugging of Java concurrent programs may be easier by mapping the Java program into a corresponding simulation program by replacing Thread by SimThread, Semaphore by SimSemaphore, CyclicBarrier by SimBarrier, and sleep(t) by hold(t). The testing of the real-time program may be performed in controlled simulation time space.

References

1. Edsger Dijkstra. Cooperating sequential processes. 1965. Reprinted in Programming Languages, F. Genuys, ed., Academic Press, New York 1968.
2. Edsger W. Dijkstra. Solution of a problem in concurrent programming control. Communications of the ACM 8, 9 (1965), 569.
3. java.util.concurrent. Java 2 Plattform Standard Ed. 5.0.
4. Richard Fujimoto: Parallel Discrete Event Simulation. CACM No. 10, Oct 1990.

A TOOL FOR ANALYTICAL SIMULATION OF B-SPLINES SURFACE DEFORMATION

Manuel González-Hidalgo Antoni Jaume-i-Capó Arnau Mir
Gabriel Nicolau-Bestard
Computer Graphics, Vision and Artificial Intelligence Group. Maths. and
Computer Science Dept. University of the Balearic Islands. Spain
email: {manuel.gonzalez,antoni.jaume,arnau.mir,gabriel.nicolau}@uib.es

KEYWORDS
Computer graphics, surface deformation, finite elements, B-splines

ABSTRACT

A tool to deform non-planar parametric surfaces based on B-splines is presented. This tool is based on an energy functional and its variational formulation. The deformation of the non-planar surface is made moving the control points of the surface. In order to do that, the space will be discretized and an ordinary differential equation has to be solved. To do it, an analytical solution will be used taking into account the features of B-splines as a finite elements. Our method will be fast because only a reduced number of control points will be moved instead of all the surface points. So, our method can be used to make simulations.

INTRODUCTION

Models lead to large number of applications, and they have been used in fields as edge detection, computer animation, geometric modeling, and so on.
This work can be viewed as a continuation of the work done by (González-Hidalgo et al., 2008a) where the deformation of planar surfaces, using an analytical solution of the associated system of differential equations, is introduced and developed. So, the same deformation model of the work of (González-Hidalgo et al., 2008a) is used in this work. The main difference between the two works is the type of deformed surfaces we perform. In this previous work, planar surfaces are deformed but in this work, we will deform non-planar surfaces as a half sphere.
First of all, a deformation model will be introduced that uses B-splines as finite elements. The model includes deformation equation, its analytical solution, examples of deformations and computational cost.
Höllig (Höllig, 2003) was the first that introduces the use of B-splines and their properties as finite elements. Our deformation model is similar to the model introduced in (Cohen, 1992). In that work, the deformation model is solved using classical finite elements (triangles and squares). In our work, we use B-splines finite elements instead.
Classical finite elements are commonly used to solve models that involves partial differential equations but it implies big data structures. On the other hand, the use of B-splines as finite elements reduces the data structure of the model. Moreover, our model has the advantage that we can solve it analytically.

B-SPLINES

The B-splines are piecewise polynomial functions with good local approximations for smooth function and local support (Piegl, 1997). Uniform B-splines are introduced in (de Boor, 1978; Farin, 1997; Piegl, 1997; Höllig, 2003). The chosen definition is given in (Höllig, 2003):

Definition 1 *An uniform B-spline of degree n, b^n, is defined by the following recurrence formula:*

$$b^n(x) = \int_{x-1}^{x} b^{n-1}(t)dt$$

starting with $b^0(x) = \begin{cases} 1, & x \in [0,1[, \\ 0, & otherwise. \end{cases}$ *(Höllig, 2003)*

To evaluate the B-splines in a simple form and fast, computationally speaking, we use the following recurrence equation (Cox, 1972; de Boor, 1978):

$$b^n(x) = \frac{x}{n} b^{n-1}(x) + \frac{(n+1-x)}{n} b^{n-1}(x-1) \qquad (1)$$

The finite element base of B-splines is defined upon a grid $h\mathbb{Z} = \{..., -2h, h, 0, h, 2h, ...\}$, where h is the scaled step:

Definition 2 *The transformation for $h > 0$ and $k \in \mathbb{Z}$ is $b^n_{k,h}(x) = b^n(\frac{x}{h} - k)$. The support of this function is $[k, k+n+1[\, h$*

The generalization to more dimensions can be performed in the following way: The N-variate B-spline of degree $\mathbf{n} = (n_1, ..., n_N)$, of index $\mathbf{k} = (k_1, ..., k_N)$ and the space discretization $\mathbf{h} = (h_1, ..., h_N)$ is defined as

$$B^{\mathbf{n}}_{\mathbf{k},\mathbf{h}}(\mathbf{x}) = \prod_{i=1}^{N} b^{n_i}_{k_i,h_i}(x_i). \qquad (2)$$

The support of this function is $\prod_{i=1}^{N}[k_i, k_i + n_i + 1[\, h_i$.

The derivatives of B-splines can be computed easily as it is shown in (González-Hidalgo et al., 2008a).
A B-spline parametric surface is a linear combination function of the B-spline base functions: $S : \Omega \subset I\!R^2 \rightarrow I\!R^3$ where

$$S(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^2} P_{\mathbf{k}} B^{\mathbf{n}}_{\mathbf{k},\mathbf{h}}(\mathbf{x}). \tag{3}$$

(see (González-Hidalgo et al., 2006) and (González-Hidalgo et al., 2008b)) where $P_{\mathbf{k}}$ are the so called *control points* and they are the elements that determine the B-spline surface.

PROPOSED MODEL

Let E be the following energy functional: $E : \Phi(S) \rightarrow I\!R$, $S \mapsto E(S)$,

$$E(S) = \int_\Omega \left(\omega_{10} \left| \frac{\partial S}{\partial u} \right|^2 + \omega_{01} \left| \frac{\partial S}{\partial v} \right|^2 + \omega_{11} \left| \frac{\partial S}{\partial u \partial v} \right|^2 + \omega_{20} \left| \frac{\partial^2 S}{\partial u^2} \right|^2 + \omega_{02} \left| \frac{\partial^2 S}{\partial v^2} \right|^2 + \mathcal{P}(S(u,v)) \right) dudv,$$

where $\Phi(S)$ is the set of all B-spline parametric surfaces, Ω is the domain of the surface S and $\mathcal{P}$ is a potential of the forces that works on the surface (Terzopoulos, 1986; Cohen, 1992; Montagnat et al., 2001).
Our goal is to achieve the minimum with the previous energy functional using an evolution model. This minimum depends on the initial surface and the used evolving model.
Using the equations of Euler-Lagrange, it can be proved (Cohen, 1992) that a local minimum of energy must satisfy:

$$-\omega_{10}\frac{\partial^2 S}{\partial u^2} - \omega_{01}\frac{\partial^2 S}{\partial v^2} + 2\omega_{11}\frac{\partial^4 S}{\partial u^2 \partial v^2} + \omega_{20}\frac{\partial^4 S}{\partial u^4} + \omega_{02}\frac{\partial^4 S}{\partial v^4} = -\nabla \mathcal{P}(S(u,v)) + \text{boundary conditions} \tag{4}$$

The surface domain is $\Omega = [0,1]^2$. Let S_0 be the initial surface or the surface to be deformed. There are four boundary conditions that corresponds to the four "fixed" edges of our surface domain: $S(u,0) = S_0(u,0)$, $S(u,1) = S_0(u,1)$, $S(0,v) = S_0(0,v)$ and $S(1,v) = S_0(1,v)$. For example, if S_0 is a plane, the previous boundary conditions will be: $S(u,0) = (u,0,0)$, $S(u,1) = (u,1,0)$, $S(0,v) = (0,v,0)$, $S(1,v) = (1,v,0)$.

The next step is to develop the variational formulation of our problem and to discretize the equation to be solved (see (González-Hidalgo et al., 2008a) for details).
At the end, the following differential equation has to be solved:

$$M\frac{d^2 P_i}{dt^2} + C\frac{dP_i}{dt} + AP_i = L_i,\ i = 1,2,3. \tag{5}$$

which corresponds to our dynamic evolution model. The matrix M is the mass matrix, C is the damping matrix, A is the stiffness matrix and L_i is the applied force on the surface.
The mass matrix M and the damping matrix C are diagonal and constant during all the time evolution.
The previous differential equation (5) can be solved analytically with a computational cost of order $O(\mathbf{N}^2)$, where $\mathbf{N} \times \mathbf{N}$ are the dimensions of matrix A. (see (González-Hidalgo et al., 2008a) for details).

COMPUTATION OF THE CONTROL POINTS OF THE INITIAL SURFACE

In this section, we are going to find the control points associated to an initial surface F. That is, if the spatial components of F are $F(\mathbf{x}) = (X(\mathbf{x}), Y(\mathbf{x}), Z(\mathbf{x}))$, where $\mathbf{x} \in \Omega = [0,1]^2$, we want to find control points $P_{\mathbf{k}}$ such that:

$$S_0(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^2} P_{\mathbf{k}} B^{\mathbf{n}}_{\mathbf{k},\mathbf{h}}(\mathbf{x}), \tag{6}$$

and the difference between $F(\mathbf{x})$ and $S_0(\mathbf{x})$ has to be as small as possible.
The set of the previous summation indexes is $\mathcal{M} = \{-n_x, \ldots, M_1\} \times \{-n_y, \ldots, M_2\}$. This set gives us the B-splines bases we use in order to find the control points. These control points are found solving the linear system of equations $S_0(\mathbf{x_j}) = F(\mathbf{x_j})$, where the points $\mathbf{x_j}$ are chosen in Ω, $\mathbf{j} \in \mathcal{M}$ and the unknowns are the control points. The previous linear system of equations takes the form $D \cdot P_i = b_i$, where $D = (B^{\mathbf{n}}_{\mathbf{i,j}}(\mathbf{x_j}))_{\mathbf{i,j} \in \mathcal{M}}$, P_i is the vector of ith coordinate of the control points $i = 1,2,3$ and b_i is: $b_1 = (X(\mathbf{x_j}))$ for the first coordinate, $b_2 = (Y(\mathbf{x_j}))$ for the second coordinate and $b_3 = (Z(\mathbf{x_j}))$ for the third coordinate.

SURFACE AND DEFORMATION REPRESENTATION

In order to simplify the user interaction, the surface representation and the application of a deformation, we need some graphic representation system. For this purpose a WYSIWYG environment has been developed.
The most extended API used to develop 2D and 3D graphics applications is OpenGL. With this environment, it is possible to implement interactive graphical applications in an easy way and it is developed strictly for graphics (Hawkins and Astle, 2004),this being a good reason to choose it. OpenInventor is an OpenGL based API, with which the representation of complex scenes and the development of complex visualization applications becomes more simple than using only OpenGL. In fact, it is the standard *de facto* for 3D

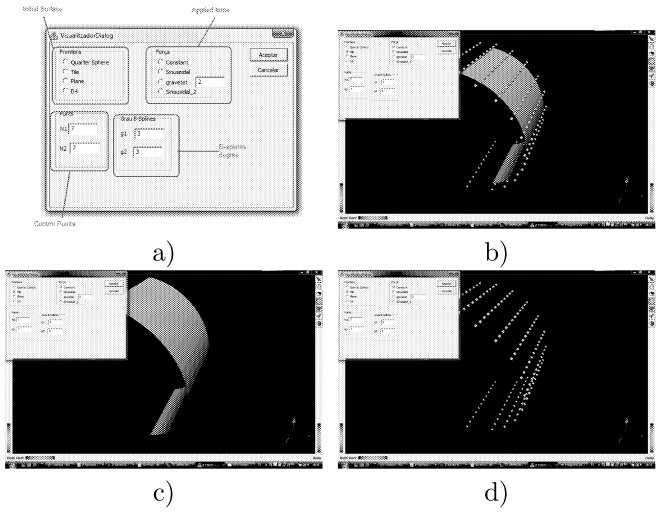


Figure 1: General appearance of the application showing the Coin3D/OpenInventor window with different enabled options. The initial menu is also shown.

Computer Graphics API for complex visualization applications (Wernecke, 1994, 1995). Unfortunately, the evolution of OpenInventor has been different from the evolution of OpenGL. The only way to have it available was under proprietary licensing from TGS (*Template Graphics Software*), but finally Silicon Graphics has released an open source specification.

In this work, we have used Coin3D, an open source OpenInventor derived API, fully compatible with the original specification. Moreover, Coin3D brings the possibility to integrate a great powerful scene representation engine with a platform-independent interface which can be integrated in a broad range of windows environments (Windows, XWindow System, Aqua, and more).

In figure 1 several views of the developed application can be seen. The surface to be deformed and the force that defines the deformation to apply to it are chosen in the initial menu. This menu has four parts (see Fig. 1.a)), from left to right and from top to bottom one can choose the initial surface, the force to apply, the number of control points of the surface and the B-splines degree. The default number of control points is 7 and the default degree of B-splines is 3.

Once the options are chosen, a window opens. This is the Coin3D examinator viewer. There are three keyboard buttons: the button s that enables or disables the display of the surface, the button p that enables or disables the display of the control points and, finally the button n that performs one step in the evolution model of the deformation. The three states that can be obtained with the buttons s and p in surface visualization can be seen in the figure 1. Our application allows to save the obtained images in such a way that a video of the deformation can be performed.

The data needed to implement the model involves the B-spline shape to be deformed but also requires the data describing the way in which the deformation has to be done. These can be supplied in two ways: requesting the information directly of the user via the graphical interface and accessing to files describing the data. There are two kinds of data files, one describing the B-Spline shape (the control points, the degree, and so on), and the other one describing the data related with the deformation, that is the forces to be applied to the shape in order to deform it, the boundary conditions and the parameters ω_{ij}, $i, j = 0, 1, 2$.

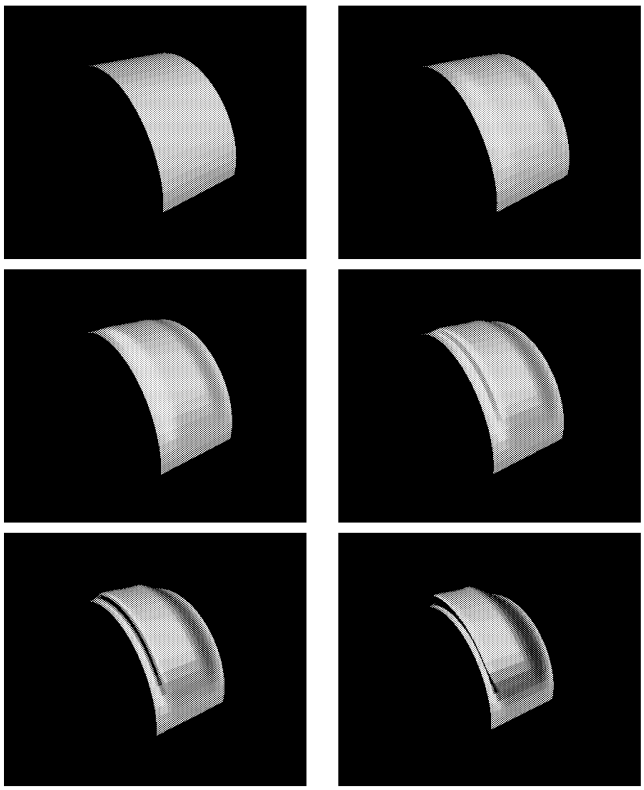


Figure 2: Six iterations of a deformation of a half cylinder are shown. The figure on the upper left is the initial surface. The deformation was made using a constant (versus time) force in the direction $(1, 4, 1)$ with all boundaries fixed.

NUMERICAL EXAMPLES

In this section, we show several examples of deformations using the model presented in the previous section. The applied forces are constant in the curse of time, because this condition is necessary for the analytical solution. If this is not the case we should use the numerical solution that can be found in the previous work of (González-Hidalgo et al., 2008b). We have applied our model to three well-known surfaces: a tile, a half cylinder and a half sphere, all these surfaces parametrized on

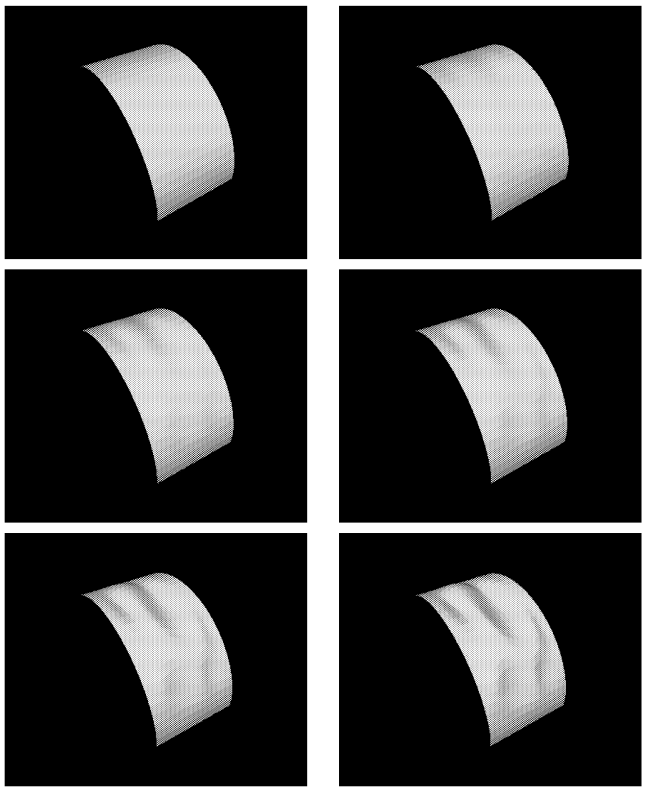


Figure 3: Six iterations of a deformation of a half cylinder are shown. The figure on the upper left is the initial surface. The deformation was made using a constant (versus time) sinusoidal force with all boundaries fixed.

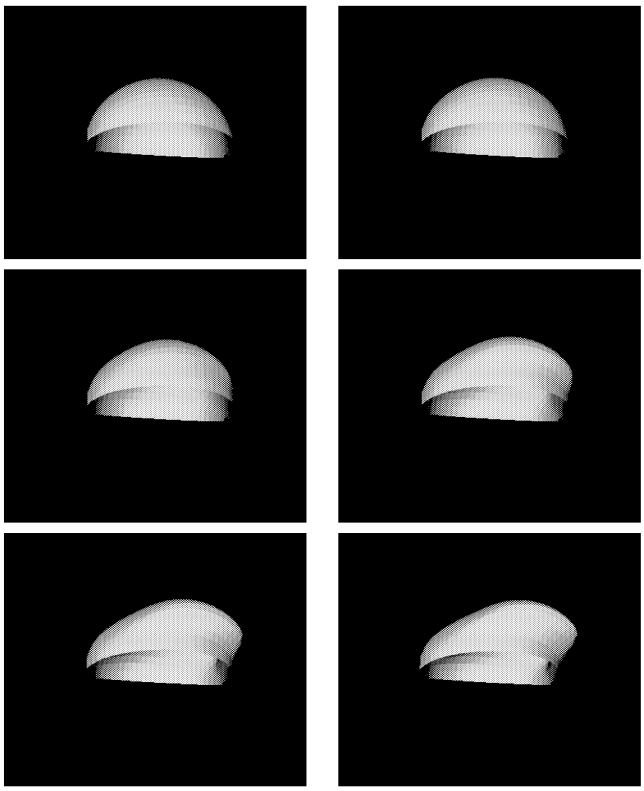


Figure 4: Six iterations of a deformation of a half sphere are shown. The figure on the upper left is the initial surface. The deformation was made using a constant (versus time) force in the direction $(1, 4, 1)$ with all boundaries fixed.

the square $[0, 1]^2$. Moreover we have considered different types of boundary conditions. First, we have considered all the boundaries fixed, and secondly, only a part of the boundary is fixed. In previous works, in order to validate the proposed method, the deformed surface was a plane defined on the $[0, 1]^2$ domain with all the boundaries fixed.

The first step of our algorithm is to compute the control points of the considered surface as it is explained in previous sections . Next, we have to set the energy functional parameters of our deformation. These are $\omega_{10} = \omega_{01} = 0.1$ and $\omega_{11} = \omega_{20} = \omega_{02} = 0.01$.

In the simulations of figures 2, 3, 4 and 5, $N_1 \times N_2 = 49$ control points are considered and bicubic B-splines are used.

In figures 2 and 4, the deformation of a half cylinder and a half sphere can be seen using a constant force in direction $(1, 4, 1)$ and the module 42.42 and all fixed boundaries.

In figures 3 and 5, the deformation of the same kind of surfaces can also be seen using the following sinusoidal force $(\sin(4\pi x), \sin(4\pi y), \cos(4\pi x)\cos(4\pi y))$, $\mathbf{x} = (x, y) \in [0, 1]^2$ and all fixed boundaries.

Last experiment presented in this work is based on the change of the boundary conditions. Now, the plane has got a free part of its boundary. The plane has $N_1 \times N_2 = 108$ control points that can be evolved. The applied force is $(\sin(4\pi x), \cos(10\pi x)\cos(10\pi y), 2\sin(10\pi y))$, $\mathbf{x} = (x, y) \in \Omega$.

CONCLUSIONS AND FUTURE WORK

A model which allows deformations of B-splines parametric surfaces is introduced. This model includes the variational formulation, the analytical solution of the corresponding differential equation and the computational cost.

To check the model, different kind of surfaces have been tested with different kind of applied forces and different boundary conditions. The experimental results show that the model is efficient and gives good deformations. All the examples have been made using C++ and Coin3D libraries.

For the time being, non constant forces versus time are studied. Moreover, we are working on another kind of surfaces, as paraboloids, ellipsoids and closed surfaces in general. Also, more boundary conditions will be con-

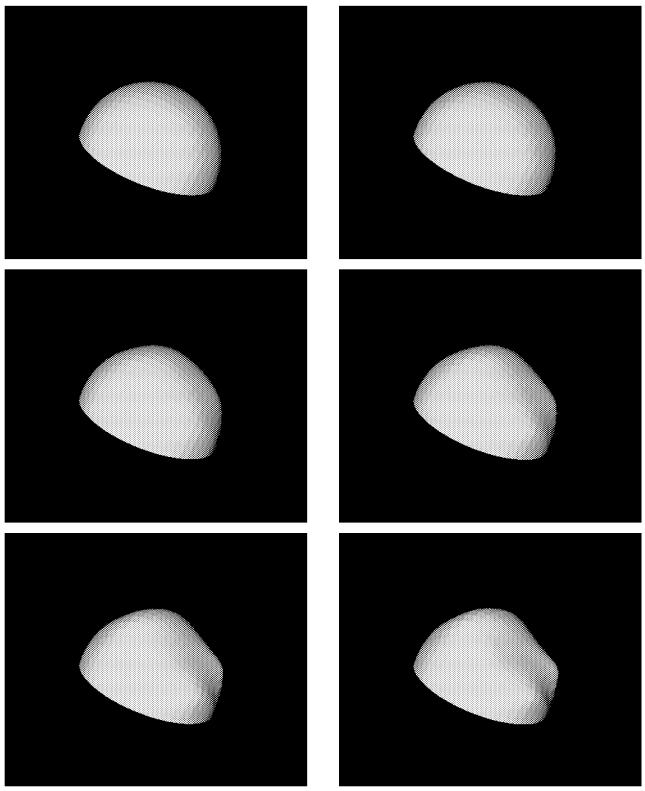


Figure 5: Six iterations of a deformation of a half sphere are shown. The figure on the upper left is the initial surface. The deformation was made using a constant (versus time) sinusoidal force with all boundaries fixed.

sidered.

ACKNOWLEDGEMENTS

This work is supported by the projects TIN2007-67993, ITADA, and TIN2007-67896, PINes, of the Spanish Government, with FEDER support. The authors would like to thank to the Department of Mathematics and Computer Science of University of the Balearic Islands.

REFERENCES

Cohen I., 1992. *Modèles Déformables 2-D et 3-D: Application à la Segmentation d'Images Médicales.* Ph.D. thesis, Université Paris IX, Dauphine.

Cox M.G., 1972. *The numerical evaluation of B-splines. IMA Journal of Applied Mathematic*, 10, no. 2, 134–149.

de Boor C., 1978. *A practical guide to splines.* Springer Verlag, New York.

Farin G., 1997. *Curves and Surfaces for Computer-Aided Geometric Desing: A Practical Guide.* Academic Press.

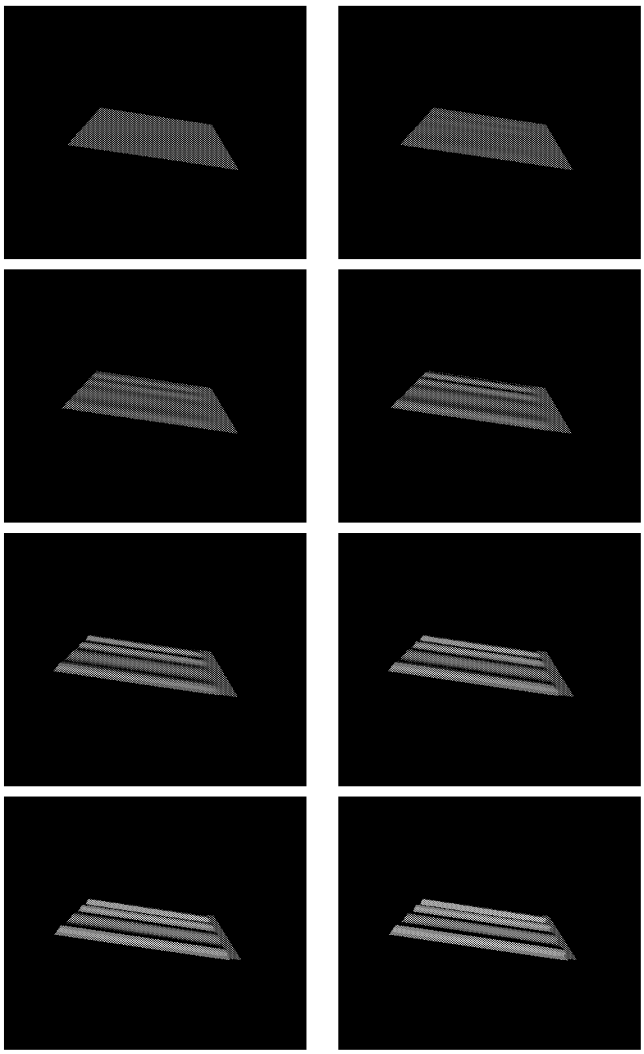


Figure 6: Six iterations of a deformation of a plane are shown. The figure on the upper left is the initial surface. The deformation was made using the following sinusoidal force $(\sin(4\pi x), \cos(10\pi x)\cos(10\pi y), 2\sin(10\pi y))$ and only one side of boundary is fixed.

González-Hidalgo M.; Jaume Capó A.; Mir A.; and Nicolau-Bestardo G., 2008a. *Analytical Simulation of B-spline Surfaces Deformation. LNCS*, 5098, 338–348.

González-Hidalgo M.; Mir A.; and Nicolau G., 2006. *An evolution model of parametric surface deformation using finite elements based on B-splines.* In *Proceedings of CompImage'2006 Conference, Computational Modelling of Objects Represented in Images: Fundamentals, Methods and Applications.* Coimbra, Portugal.

González-Hidalgo M.; Mir A.; and Nicolau G., 2008b. *Dynamic parametric surface deformation using finite elements based on B-splines. International Journal for Computational Vision and Biomechanics*, 1, no. 2, 151–611.

Hawkins K. and Astle D., 2004. *OpenGL game programming*. Course Technology PTR; 1 edition (May 1, 2002). Premier Press.

Höllig K., 2003. *Finite element methods with B-splines*. Frontiers in Applied Mathematics. SIAM, Philadelphya.

Montagnat J.; Delingette H.; and Ayache N., 2001. *A review of deformable surfaces: topology, geometry and deformation. Image and Vision Computing*, 19, no. 14, 1023–1040. URL `citeseer.nj.nec.com/498203.html`.

Piegl L. & Tiller W., 1997. *The NURBS book*. Springer Verlag, Berlin.

Terzopoulos D., 1986. *Regularization of inverse visual problems involving discontinuities. IEEE PAMI*, 8, no. 4, 413–424.

Wernecke J., 1994. *The Inventor Toolmaker: Extending Open Inventor, release 2*. Addison–Wesley. ISBN 0-201-62493-1.

Wernecke J., 1995. *The Inventor Mentor: Programming Object–oriented 3D graphics with Open Inventor release 2*. Addison–Wesley. ISBN 0-201-62495-8.

SIMULATION DATA ANALYSIS TOOLS

EVALUATING THE POTENTIAL OF ORTHOGONAL DEFECT CLASSIFICATION FOR VERIFICATION AND VALIDATION OF MODELLING AND SIMULATION APPLICATIONS

Zhongshi Wang

Institut für Technik Intelligenter Systeme (ITIS)
D-85577 Neubiberg, Germany
zhongshi.wang@unibw.de

KEYWORDS
Verification and Validation (V&V), Orthogonal Defect Classification (ODC), Model Deficiency Classification, V&V Techniques.

ABSTRACT

Model deficiencies, despite their negative influences on assessment of modelling and simulation (M&S) applications, carry a large amount of insightful information, which can be used to measure different aspects of the M&S development process and its verification and validation (V&V). Although there already exist various categorizations of model deficiencies, none of which can be used as a measurement tool to classify and analyze deficiency data collected in practice. Since several classifications and even an IEEE standard have been developed in the software community, this work investigates a well-known classification approach, the IBM Orthogonal Defect Classification (ODC) scheme in the M&S context. Based on the findings, a framework for developing model deficiency classifications is proposed, which can be applied to any simulation study with well structured model development and V&V processes.

INTRODUCTION

Development of modelling and simulation (M&S) applications involves human- and knowledge-intensive activities, in which errors, uncertainties, or inadequacies leading to quality deficiencies in models and simulation results are inevitable. Verification and validation (V&V) focuses on assessing the accuracy quality characteristic of an M&S application with respect to its objectives (Balci 2004, Shannon 1975), and is intended to ensure that only correct and suitable models and simulation results are used in practice.
Model deficiencies, on the one hand, due to their negative impact on the assessment issue, should be prevented from the development's point of view, or be identified and corrected from the V&V's point of view. On the other hand, a large amount of information is attached to the model deficiencies, characterizing the different aspects of model development and its V&V. In order to extract the semantic information of interest from the collected deficiency data, classification of model deficiencies is required. However, there is currently not sufficient research on this area.
In the M&S literature, the term "errors" is frequently used to refer to model deficiencies. Its meaning is sometimes, however, extremely different, such as the definitions of type I, II, and III errors (Balci and Nance 1985), and of acknowledged or unacknowledged errors (Oberkampf et al. 2000). Typically, model deficiencies are classified into several coarse-grained categories relating to the corresponding development phases in the M&S life cycle (Shannon 1975, Carson 2002, Schmeiser 2001), such as data model errors, logic model errors, programming errors, experimentation errors, or interpretation errors. Suchlike taxonomies can not be used as a basic instrument to measure and analyze the semantics included in model deficiencies.
In the software community, however, several defect classification schemes have been already developed for different measurement purposes. In publications, the most referenced classification approaches are the IBM Orthogonal Defect Classification (ODC) scheme (Chilarege et al. 1992), the Hewlett-Packard (HP) scheme (Grady 1992), and an IEEE standard (IEEE 1993). This paper investigates the existing software defect classifications in the M&S and V&V context. The focus of this investigation is on the ODC scheme, because:

1. applications of ODC are widely reported;
2. this scheme has been in continuous development since its first release.

Findings of the investigation indicate that the software defect classifications (ODC and other schemes) can not be applied to analysis of model deficiencies because of the more complex application context in M&S. The general classification principles, however, are still essential for developing an M&S-specific classification scheme. Because of the huge variety of M&S applications, an overall and complete scheme is under current practice

impossible. Therefore, this paper proposes a classification framework for model deficiencies, which defines four different attributes to capture the information used to measure the current M&S and V&V processes.
The remainder of this paper is structured as follows: an overview of the IBM ODC scheme is introduced in the next section. Then the investigation of ODC is discussed. Based on the findings, a deficiency classification framework is proposed. Finally, the last section concludes the work.

OVERVIEW OF ORTHOGONAL DEFECT CLASSIFICATION

Orthogonal Defect Classification (ODC)(Chilarege et al. 1992) is one of the most prominent measurement methodologies used to classify and analyze software defects collected during the software development life cycle. Like other classification approaches, the basic concept of ODC can be outlined as follows:

1. defining a classification scheme with different attributes, which characterize the desired aspects of the development process;

2. using the defined classification scheme to extract the semantic information from individual defects;

3. analyzing and evaluating the aggregated data, which not only deliver insight into their own problem areas, but also create a relationship between cause and effect that can provide a process diagnostic.

ODC Attributes

An essential criterion of the IBM ODC scheme is to categorize a software defect into distinct classes characterized by an orthogonal set of attributes. The term "orthogonal" means that both attributes and their values do not overlap and are mutually independent. This non-redundant nature not only reduces the human error in classification, but also enables to capture the unique and unambiguous information of interest for evaluating the current development and testing process. In the ODC scheme, the following eight different attributes are defined for a software defect (Chilarege et al. 1992, Butcher et al. 2002):

- **Activity** refers to the testing activity being performed at the time the defect was discovered, i.e. when was the defect found?

- **Trigger** represents the environment or condition (during the testing activity) that had to exist for the defect to surface, i.e. how was the defect discovered?

- **Impact** describes the effect on the customer if the defect had not been found, i.e. nature and degree of pain.

- **Target** indicates the high-level identity of the entity that was fixed, i.e. what gets fixed?

- **Defect type** specifies the nature of the actual correction that was made, i.e. what had to be corrected?

- **Qualifier** captures the element of either a nonexistent or wrong or irrelevant implementation.

- **Source** identifies which part of the work product the defect target belongs to, e.g. in-house, outsourced, library reuse and so on.

- **Age** shows the developmental history of the defect target, e.g. new, old, rewritten, or refixed.

More detailed description of the eight attributes and their values defined in the ODC scheme can be found in the ODC website (IBM Research 2010).

Analysis of Classified Data

While all of the ODC attributes capture the meaningful information from a defect and therefore are valuable to analyze, the full scope of the data is not always necessary for practical use (Chillarege 2007). Depending on the objective of an analysis, the data of one certain attribute alone or in combination with other defect attributes can be classified and applied.
When alone, the trend of the defect type distribution can be used as measurement to estimate the progress of the software product through the development process. For example, the function defects should be found typically in the early phase like design inspection, and decreasingly in the late phases such as unit test, integration test, and system test. On the contrary, the timing and serialization defects should be found mainly at system test. Thus, when compared with other defect types, a decreasing trend of the function defects should be observed from phase to phase. If however, the percentage of the function defects from a current development is lower in the early phase and much higher in the testing phases than it should be, such a distribution deviation rather indicates the inadequate design inspection. In this context, a repeated process for design review seems more effective than the intensive testing (Chilarege et al. 1992).
On the other hand, if some defect attributes are applied in combination, then an analysis of multiple perspectives is possible. As defined in the ODC scheme, the defect type attribute addresses the semantics of defect correction, while the defect triggers are referred to the catalysts which allow the defect to surface, namely the facilitators that force a fault to a failure (Chillarege and

Bassin 1995, Chillarege and Prasad 2002). The attribute trigger provides insight on the defect detection process, indicating how defects are revealed. Thus, when a relationship between defect types, triggers, and activities is established, it is possible to investigate what kind of detection technique finds which type of defects for each detection activity. In practice, applications of ODC for different purposes are reported, such as for enhancing the root cause analysis, quality management, and process improvement (Schulz 1999, Chillarege 2007).

Related Classification Schemes

The Hewlett-Packard (HP) approach is another industrial classification scheme applied frequently in practice. Compared to the ODC scheme, only three attributes: origin, type, and mode are defined in the HP approach. The defect origin indicates the activity in the software life cycle, where the defect was introduced (not where it was found). Unlike ODC, the HP defect type is referred to the particular area of an origin, which is responsible for the defect. In addition, the mode attribute describes the reason of a defect, i.e. whether the information was missing, unclear, or wrong etc.

Thus, by means of the close relationship between the three defect attributes, the HP scheme focuses rather on the process improvement, namely identifying the error-prone areas for future development or testing, based on the analysis of the associated defect origins and types (Huber 2000). However, the issue on how to detect defects such as the activities and triggers in ODC, is not an emphasis discussed in the HP scheme.

In addition, since there exits a fixed relationship between the attributes origin and type, their values are already semantically dependent of each other. Thus, a customization of the HP scheme is not always possible.

The focus of the IEEE Standard Classification for Software Anomalies (IEEE 1993) is on providing a general classification process to establish a defect classification scheme for the own needs. The process is consisted of four sequential steps: Recognition, Investigation, Action, and Disposition. According to these process steps, a classification scheme is categorized into several dimensions which contain the different attributes characterizing the work flow from the recognition of a defect through to its closure. For example, the attributes activity, phase, and symptom are assigned to Recognition, the defect cause and type appear in Investigation, and the defect correction is found in Action. However, with respect to practical application, case studies on this standard classification are not sufficiently reported (Wagner 2008).

EVALUATION OF ODC IN THE M&S CONTEXT

The ODC scheme and other existing classification approaches were developed specifically for applications within the software community. Since a simulation study is conduced in a more complex context than software development (Wang and Lehmann 2008), the ODC scheme is not a suitable measurement tool to classify and analyze model deficiencies. This section discusses what M&S-specific aspects should be considered, when developing a classification scheme used to improve the V&V process for an M&S project.

Scope of the M&S Life Cycle

Commonly, regardless of how a concrete M&S process looks like, development of a simulation model could be categorized into four progress stages: Model Initialization, Model Design, Model Realization, and Model Application (Wang and Lehmann 2007a). For each progress stage, one or more intermediate products are to be prepared, and the quality of them is also to be estimated as part of the model development. As an example, a simplified M&S process (without consideration of V&V process) is shown in Figure 1. The progress stage Model Initialization includes the work product Sponsor Needs (SN), in Model Design the products Structured Problem Description (SPD), Conceptual Model (CM) and Formal Model (FM) are defined, Executable Model (EM) is prepared in Model Realization, and Simulation Results (SR) are achieved in Model Application.

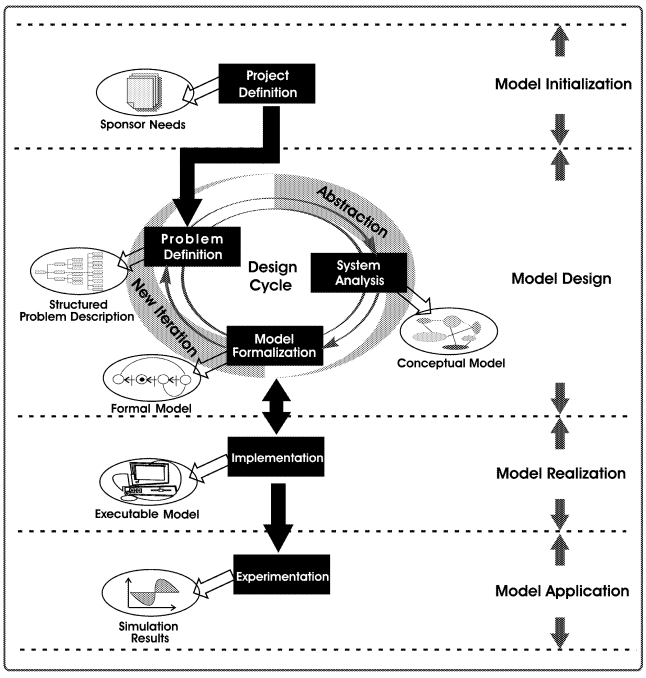


Figure 1: An Example of the M&S Progress Stages

Apart from the Initialization stage, which is available for any engineering process, the ODC attributes and their values can be applied mostly to the stage Model Realization, because this stage is concerned with converting a conceptual/mathematical model into a computerized form, a typical task of software development. The following M&S-specific aspects are, however, not completely covered by ODC:

- **Model Design**
  The specification and formalization of a well-defined simulation model is not a part of software development. During the model design process, the mathematical/logical/graphical/verbal representation of the real system of interest is developed for the objectives of a particular study. Since a typical simulation study requires multifaceted knowledge in different disciplines, a variety of representation means such as mathematical equation systems, queuing networks, Petri nets, process algebra (Bergstra and Klop 1984) and Discrete Event System Specification (DEVS) (Zeigler et al. 2000) etc. can be used for developing the work products SPD, CM, and FM.

- **Data Modelling**
  Throughout the entire M&S development life cycle, an enormous amount of information must be gathered, analyzed and modeled in terms of qualitative and quantitative data. This task is referred to as data modelling (Shannon 1975). Three types of data (Sargent 2000) are to be handled: some data are used to specify the model components, and finally, become integrated into the model built; while other data are used either to compare with the simulation results for test purpose or to perform simulation experiments. According to the different applications of input data, data modelling has to closely cooperate with each project progress stage, and therefore, is considered as an integrated part of model development (Rabe et al. 2008).

- **Model Application**
  Model Application refers to the process of experimenting with the simulation model for a specific purpose, including design of model experiments, execution of simulation runs, and analysis and interpretation of simulation results. This aspect is out of the consideration range of the ODC scheme.

Sources of M&S Deficiencies

In the software community, despite some different terminologies, the relationship between the cause of a defect and its consequences is consistently specified. In general, as defined in an IEEE Standard (IEEE 1990), an error is a human oversight or wrong decision, which results in a fault, or a defect within the software (including requirements specifications, design, and code). When the software is executed, a fault, or a combination of faults, may (or may never) cause a failure. In many cases, the term "defect" is used in a generic manner referring to a fault, a failure, or even an error, such as in the ODC, and HP schemes as well as in this paper so far.

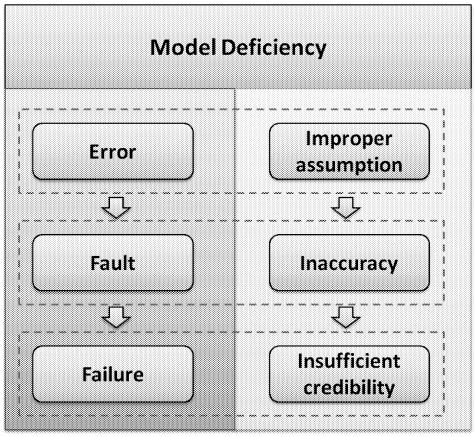


Figure 2: M&S Deficiencies

Compared to a software defect, the meaning of what is wrong or inaccurate in a model appears more complex, just as the well-known statistician George Box stated: "All models are wrong, some are useful". From the V&V viewpoint, two sources of M&S deficiencies (shown in Figure 2) can be identified within the M&S life cycle. On the one hand, since M&S requires intensive investment of human efforts, and includes software development, the causal structure of human errors, faults, and failures also exists in a model.
On the other hand, due to lack of knowledge, e.g. uncertainty (Oberkampf et al. 2002) in the different M&S development phases, or for the purpose of model simplification, assumptions and approximations must be made. Whether they are accurate enough or actually acceptable, can be only estimated with respect to the specified objectives of the model by particular V&V techniques such as face validation (Shannon 1975, Balci 1995, Law and Kelton 2007). Thus, inaccuracies are those deficiencies in the model, which result from improper or unacceptable assumptions and approximations made.
In some cases, although the individual assumptions and approximations are estimated as acceptable, the effect of their aggregation could still cause unacceptable inaccuracies. In the course of M&S development, improper assumptions and approximations cause further inaccuracies which can eventually lead to insufficient model's credibility, i.e. the model is not useful.
Thus, both sources of M&S deficiencies must be handled by the V&V activities. In the following, the term "deficiency" is used in a generic manner referring to both

defects and inaccuracies in a model.

Requirements of a Classification Scheme for Analyzing Model V&V

An analysis of the model V&V process usually needs the classified information about what type of deficiencies that are detected, how (by which V&V technique), and when (in which V&V activity) they are detected, in addition to the model development phase (such as Problem Definition, System Analysis, or Experimentation shown in Figure 1), in which they are introduced. The required information can, however, not be captured by using the software defect classifications like the ODC scheme.

As discussed above, since a simulation study requires a more complex process than software development, the ODC attributes and their range of values, such as Activity and Target, do not cover the full scope of an M&S project. In addition, the defect type attribute also can not completely and exactly specify the deficiencies appearing in a model. This means that some ODC attributes must be extended, or modified, while some new attributes should be additionally defined. For example, the ODC triggers can be only applied to identification of failures in software code. In the context of model V&V, this attribute should be preferably replaced by another attribute concerning the V&V techniques. A new attribute referring to the injection of a deficiency in the model, should be defined. In addition, some ODC attributes such as age, and source, are not of crucial importance for the purpose of analyzing model V&V.

Thus, although the software defect classifications, such as ODC, can not be used for M&S applications, their basic principles and the practical experiences of their customizations (Freimut et al. 2005, Freimut and Denger 2003, Kelly and Shepard 2001) provide valuable insight into development of an M&S-specific classification scheme.

PROPOSED DEFICIENCY CLASSIFICATION FRAMEWORK FOR MODEL V&V

Due to the huge variety of M&S applications, and their corresponding development and V&V processes, it is under current practice not possible to develop an overall and complete classification scheme applicable to any simulation study. Therefore, the aim of this work is to propose a deficiency classification framework with definitions of orthogonal attributes. The individual values of each attribute are not specified in the framework. They will be determined according to the structure of the concrete M&S processes used in practice. Thus, this framework can be implemented in any M&S application, as long as its development and V&V processes are well defined and structured.

Description of the Framework

As shown in Figure 3, four attributes: Activity, Detection, Introduction, and Type are defined in the framework, capturing the information used to characterize the V&V process for different analysis purposes, such as:

- to optimize a combination of V&V techniques by analyzing what kind of detection technique finds which type of deficiencies in which V&V activity;
- to identify the error-prove areas of the M&S development process by analyzing the deficiencies with particular types and their insertion in the model;
- to estimate the effectiveness of V&V activities by analyzing the actual deficiencies types identified in certain V&V activities and comparing them with expected types;
- to evaluate and improve V&V techniques like inspections by investigating the deficiencies they identify and do not identify;
- to evaluate the propagation of M&S deficiencies by analyzing when they are introduced in the model and when are detected.

Attribute Name	Attribute Meaning	Description
Activity	When was the deficiency identified?	The V&V activity being performed at the time the deficiency was identified. The values of this attribute are the individual activities defined in the V&V process, which is integrated into the underlying M&S development life cycle.
Detection	How was the deficiency identified?	The V&V techniques or the combination of several V&V techniques which are used to reveal the deficiency. Examples are face validation, inspections, semantic analysis, tesing techniques, proof methods and so on.
Introduction	What task causes the insertion of this deficiency?	The first phase of the M&S development life cycle, in which the deficiency could have been prevented. Values of this attribute are the different phases of model development, such as Problem Definition, System Analysis, Formalization, Implementation, and Experimentation.
Type	What is the result of a deficiency in the model?	The subject or topic of an intermediate product, which is damaged because of the deficiency. Values of this attribute are the subjects and topics of each intermediate product defined in the applied model development process. For example, simulation objectives, model boundary requirements, and model structure and behavior requirements in Structured Problem Definition (SPD).

Figure 3: Deficiency Classification Framework

An Implementation Example

When implementing the framework in a practical context, a set of values for each deficiency attribute is to be determined on basis of the M&S and V&V processes being applied. Two issues are to be considered (Freimut 2001):

- The values for each attribute should be orthogonal (or distinct), i.e. only one attribute value is appropriate for a particular model deficiency;

- The values for each attribute should be complete. This means that each deficiency could find an appropriate attribute value.

In the following, as a guidance on using this framework to establish a complete classification scheme, the selection of the attributes values is demonstrated with an example.

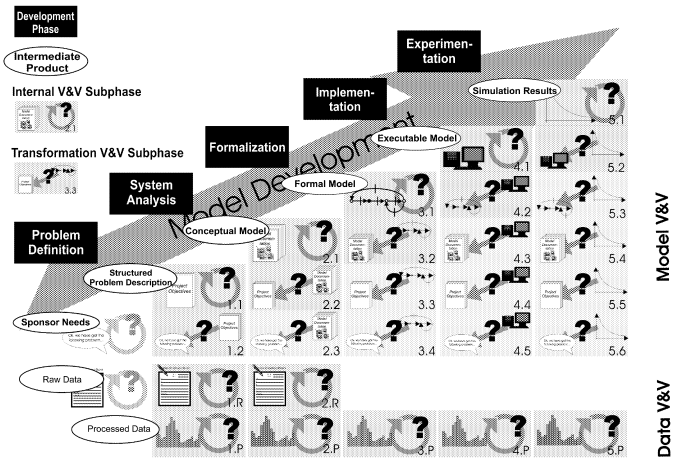


Figure 4: Example of M&S and V&V Processes

Figure 4 illustrates a generalized V&V processes with two closely associated parts of V&V activities: model V&V and data V&V (Brade 2000, Wang and Lehmann 2007b). This V&V process is integrated in a model development process which for instance defines five modelling phases (depicted as black boxes): Problem Definition, System Analysis, Formalization, Implementation, and Experimentation. The corresponding results of these development phases are referred to as intermediate products, namely Structured Problem Description (SPD), Conceptual Model (CM), Formal Model (FM), Executable Model (EM), and Simulation Results (SR). The V&V activities are organized as a triangle-like matrix. The columns of the matrix represent the V&V main phases, which are associated with the intermediate products; while intersections between the columns and rows split the V&V main phases into V&V sub-phases. Thus, according to the definitions of the Framework, the values of each attribute can be determined as follows:

- Activity
  The values are the individual V&V activities numbered from 1.1 to 5.P in Figure 4.

- Detection
  With respect to V&V techniques for the attribute detection, the Balci's taxonomy (Balci 1998) is considered, in which the most (approx. 80) of the V&V techniques applicable to M&S applications are primarily classified into four categories: informal, static, dynamic, and formal. Those classified V&V techniques can be used as attribute values for Detection.

- Introduction
  The values are clearly the different model development phases, i.g. Problem Definition, System Analysis, Formalization, Implementation, and Experimentation.

- Type
  Based on the M&S and V&V processes illustrated in Figure 4, a guideline for model development and documentation (Lehmann et al. 2005, Wang et al. 2009) was developed, which defines a hierarchical structure of the intermediate products. Each intermediate product is divided into several subjects (or topics), some of which can be further divided into aspects. This structure is used to classify the attribute type. As defined in the framework (see Figure 3), each subject of the intermediate products SPD, CM, FM, EM, and SR represents a type value. Thus, all values of this attribute can be described by means of a two dimensional matrix, as shown in Figure 5. For example, SPD includes the following subjects as type values: M&S objectives, model boundary requirements, model structure and behavior requirements, data requirements, experimental framework requirements, M&S constraints, and acceptability criteria.

SPD	CM	FM	EM	SR
• M&S objectives • Model boundary requirements • Model structure and behaviour requirements • Data requirements • Experimental framework reqirements • M&S constraints • Acceptability criteria	• System environments and subsystems • Model input and output parameters • Model structure specification • Model behaviour specification • Interactions of submodels • Input data analysis	• Identification of the formalisms and tools • Formalized model boundary description • Formalized model structure • Fomalized model behaviour • Fomalized dynamic interactions between submodels • Input model • Formalized experimental framework	• Simulation environments, language, and libraries • High level design • Low level design • Simulation infrastruture • Runtime environment • Interface • Structure and internal behaviour • Interaction of submodels • Data	• Design of experiments • Tactical planning • Input data • Execution environment • Execution of experiments • Output documentation and presentation • Results analysis • Results interpretation

Figure 5: Description of the Type Values

CONCLUSION

Findings of the investigation presented in this paper indicate the software defect classifications can not be applied in the M&S context. However, their basic principles, such as classification of software defects into distinct attributes, analysis of attributes data separately or in combination for different measurement purposes, are also essential criteria for developing model deficiency classifications.
Due to the large variations of M&S applications, this work proposes a classification framework instead of an all-embracing and detailed scheme. Four attributes are

defined for characterizing the conducted V&V activities: 1.) Activity shows when the model deficiency was identified; 2.) Detection specified how the deficiency was detected; 3.) Introduction refers to the activity causing the insertion of the deficiency into the model; and 4.) Type indicates the consequence of the deficiency in the model. Since this framework has an open structure, new attributes for additional measurement aspects can be easily added. For example, a new attribute Impact, which refers to the severity level of the deficiency for accuracy assessment, is conceivable.

Our ongoing work focuses on applying this framework to a simulation project in practice. As shown in the former section, based on the underlying M&S and V&V processes, the individual values of the defined attributes have been already determined, i.e. a detailed deficiency classification scheme has been established. Two issues of this scheme are to be estimated: 1.) the classification effort using this scheme, and 2.) the ability of this scheme to analyze the deficiency data.

Acknowledgment. The author would like to thank Dr. Junlan Qian for useful comments and suggestions.

REFERENCES

Balci O., 1995. *Principles and Techniques of Simulation Validation, Verification, and Testing.* In: Proceedings of the 1995 Winter Simulation Conference.

Balci O., 1998. *Verification, Validation, and Testing.* In J. Banks (Ed.), *Handbook of Simulation*, John Wiley & Sons, chap. 10.

Balci O., 2004. *Quality Assessment, Verification, and Validation of Modeling and Simulation Applications.* In: Proceedings of the 2004 Winter Simulation Conference.

Balci O. and Nance R.E., 1985. *Formulated Problem Verification as an Explicit Requirement of Model Credibility.* Simulation, Vol. 45, No. 2, pp. 76-86.

Bergstra J. and Klop J., 1984. *Process Algebra for Synchronous Communication.* Information and Control, Vol. 60, No. 1, pp. 109-137.

Brade D., 2000. *Enhancing Modeling and Simulation Accreditation by Structuring Verification and Validation Results.* In: Proceedings of the 2000 Winter Simulation Conference.

Butcher M.; Munro H.; and Kratschmer T., 2002. *Improving Software Testing via ODC: Three Case Studies.* IBM System Journal, Vol. 41, No. 1.

Carson J.S., 2002. *Model Verification and Validation.* In: Proceedings of the 2002 Winter Simulation Conference.

Chilarege R.; Bhandari I.S.; Chaar J.K.; Halliday M.J.; Moebus D.S.; Ray B.K.; and Wong M.Y., 1992. *Orthogonal Defect Classification - A Concept for In-Process Measurements.* IEEE Transactions on Software Engineering, Vol. 18, No. 11.

Chillarege R., 2007. *ODC Measurement and Analysis - Industry Applications.* Chillarege Inc.

Chillarege R. and Bassin K.A., 1995. *Software Triggers as a Function of Time - ODC on Field Faults.* Fifth IFIP Working Conference on Dependable Computing for Critical Applications (DCCA-5).

Chillarege R. and Prasad K.R., 2002. *Test and Development Process Retrospective - A Case Study Using ODC Triggers.* In: Proceedings of the International Conference on Dependable Systems and Networks (DSN'02).

Freimut B., 2001. *Developing and Using Defect Classification Schemes.* Fraunhofer IESE.

Freimut B. and Denger C., 2003. *A Defect Classification Scheme for the Inspection of Quasar Requirement Documents.* Fraunhofer IESE.

Freimut B.; Denger C.; and Ketterer M., 2005. *An Industrial Case Study of Implementing and Validating Defect Classification for Process Improvement and Quality Management.* 11th IEEE International Software Metrics Symposium.

Grady R.B., 1992. *Practical Software Metrics for Project Management and Process Improvement.* Prentice-Hall Inc.

Huber J.T., 2000. *A Comparison of IBM's Orthogonal Defect Classification to Hewlett Packard's Defect Origins, Types, and Modes.* In: Proceedings of International Conference on Applications of Software Measurement (ASM).

IBM Research, 2010. *Details of ODC v 5.11.* Center for Software Engineering. Http://www.research.ibm.com/softeng/ODC/DETODC.HTM.

IEEE, 1990. *IEEE Standard Glossary of Software Engineering Terminology.* IEEE std 610.12-1990.

IEEE, 1993. *IEEE Standard Classification for Software Anomalies.* IEEE std 1044-1993.

Kelly D. and Shepard T., 2001. *A Case Study in Use of Defect Classification in Inspections.* In: Proceedings of the 2001 Conference of the Centre for Advance Studies on Collaborative Research.

Law A.M. and Kelton W.D., 2007. *Simulation Modeling and Analysis.* McGraw-Hill, fourth ed.

Lehmann A.; Bel-Haj-Saad S.; Best M.; Köster A.; Pohl S.; Qian J.; Waldner C.; Wang Z.; and Xu Z., 2005. *Leitfaden für Modelldokumentation.* Abschlussbericht, ITIS e.V. an der Universtität der Bundeswehr München.

Oberkampf W.L.; DeLand S.M.; Rutherford B.M.; Diegert K.V.; and Alvin K.F., 2000. *Estimation of Total Uncertainty in Computational Simulation.* Sandia National Laboratories.

Oberkampf W.L.; DeLand S.M.; Rutherford B.M.; Diegert K.V.; and Alvin K.F., 2002. *Error and uncertainty in modeling and simulation.* Reliability Engineering and System Safety, Vol. 75, No. 3, pp. 333-357.

Rabe M.; Spieckermann S.; and Wenzel S., 2008. *Verikation und Validierung für die Simulation in Produktion und Logistik - Vorgehensmodelle und Techniken (in German)*. Springer-Verlag, Berlin.

Sargent R., 2000. *Verification, Validation and Accreditation of Simulation Models*. In: Proceedings of the 2000 Winter Simulation Conference.

Schmeiser B.W., 2001. *Some Myths and Common Errors in Simulation Experiments*. In: Proceedings of the 2001 Winter Simulation Conference.

Schulz C., 1999. *Orthogonal Defect Classification (ODC)-based Test Planning and Development*. In: Proceedings of the International Conference on Applications of Software Measurement.

Shannon R.E., 1975. *Systems Simulation - the Art and Science*. Prentice-Hall, Englewood Cliffs, N.J.

Wagner S., 2008. *Defect Classification and Defect Types Revisited*. In: Proceedings of the International Workshop on Defects in Large Software Systems (DEFECTS 2008).

Wang Z.; Kißner H.; and Siems M., 2009. *Applying a Documentation Guideline for Verification and Validation of Simulation Models and Applications: An Industrial Case Study*. In: Proceedings of the 7th Industrial Simulation Conference (ISC'09).

Wang Z. and Lehmann A., 2007a. *A Framework for Verification and Validation of Simulation Models and Applications*. In J.W. Park; T.G. Kim; and Y.B. Kim (Eds.), *AsiaSim 2007*, Springer, Heidelberg, Communications in Computer and Information Science.

Wang Z. and Lehmann A., 2007b. *Verification and Validation of Simulation Models and Applications: A Methodological Approach*. In A.N. Ince and A. Bragg (Eds.), *Recent Advances in Modeling and Simulation Tools for Communication Networks and Services*, Springer, New York.

Wang Z. and Lehmann A., 2008. *Expanding the V-Modell® XT for Verification and Validation of Modelling and Simulation Applications*. In: Proceedings of the 7th International Conference on System Simulation and Scientific Computing, IEEE.

Zeigler B.P.; Praehofer H.; and Kim T.G., 2000. *Theory of Modeling and Simulation*. Second Edition, Academic Press.

REGRESSION METAMODELS FOR TRANSIENT SIMULATION ANALYSIS

Rita Marques Brandão
Departamento de Matemática
Universidade dos Açores and CEG-IST
Ponta Delgada, Portugal
email: rita@uac.pt

Acácio M. O. Porta Nova
Departamento de Engenharia e Gestão
Instituto Superior Técnico
Lisboa, Portugal
e-mail: apnova@ist.utl.pt

KEYWORDS
Output Analysis, Regression Metamodels, Time Series Models, Initial Transient, Steady-State.

ABSTRACT

We present an approach for analyzing the transient behavior of simulation models, based on regression metamodels. The goal is to be able to detect convergence to steady-state and reduce the consequent initialization bias. The procedure is evaluated using some analytical results for a single server queue and some further experimental results are discussed.

INTRODUCTION

Simulation output analysis is certainly one of the most important areas in discrete event simulation. It is also, arguably, the one that has attracted most attention and published work, in spite of the complexity of some of its topics. The transient analysis of stationary simulation models is one such topic and one that is full of contradictions. It is a very hard problem to handle analytically, especially when time is the independent variable–the work of Bailey (1964; 1957) being a notable exception. However, in general, we only want to analyze the transient period of the system behavior in order to discard it. In fact, our main goals are usually the estimation of appropriate measures of centrality and variability, frequently the sample average and one or more types of confidence intervals. It is commonly accepted that the initial transient yields some form of undesirable bias, and many approaches have been proposed, either to reduce it, or equivalently, to detect the beginning of steady-state. The simulation community has been addressing this problem for many years, and some earlier results are reported, for instance, in the surveys by Gafarian et al. (1978), and Wilson and Pritsker (1978b;a). Other surveys have also been published since, like the ones by Pawlikowski (1990), or, more recently, Hoad et al. (2008). A significant amount of work has also been spent trying to develop automated procedures, that might be integrated with simulation software in order to relieve the user (not necessarily a simulation expert) from the burden of analyzing the initial transient.
The approach that we propose in this paper is quite different. Our contention is that the transient behavior of many stationary simulation models will follow some smooth, predictable curve, superimposed with correlated random noise. We know that this is true for some simple queueing models, namely the ubiquitous M/M/1 queue analyzed in Bailey (1957). On the other hand, we also know that this occurs for some nonstationary simulations as well; see Bailey (1964; 1957) and Brandão and Porta Nova (2009). In our previous work, we fitted classical time series models to simulation responses and derived simulation metamodels from the corresponding forecast functions, to try to capture the intrinsic system behavior. A simulation metamodel is a simpler model, usually an expression, of the simulation model itself. In this paper, we use classical regression techniques to uncover the deterministic component of the response and we rely on classical Box-Jenkins time series models to express the random perturbations, or noise; see Box et al. (1994). The goal is to detect a suitable truncation point to remove the initial bias, and then to fit an autoregressive-moving average (ARMA) model to the stationarized portion that will allow us to build appropriate confidence intervals. During the whole process, we feel that visualization of the intermediate results, past experience and intuition are all components from expert knowledge that are essential for a successful outcome of the procedure.
This paper is organized as follows. First, we present our procedure for detecting and removing initial bias from simulation output. Then, we evaluate our approach using the analytical results available for the $M/M/1$ queue, and test it on a network of queues with feedback. We conclude, with a summary and some recommendations for further work in this area.

PROCEDURE

1. *Construction of two averaged time-series.* We need to analyze the transient behavior of selected responses in the target system. To do so, we focus our attention on two different starting conditions: (i) the model is started empty and idle; and (ii)

the model is started with a number of entities in the system much larger than the one expected in steady state.

We assume that the duration of the simulation has already been chosen. For the number of independent replications, we have found out in Brandão and Porta Nova (2009) that 30 runs was a good choice, not only in terms of statistical robustness, but also from the experimental viewpoint. Then, we sample continuous time responses (number of entities in the system, or waiting) at a convenient discrete-time interval, that has to be decided. Finally, the corresponding observations across runs for both cases are averaged, and it is these two averaged time series that are analyzed in the remaining steps of our proposed procedure.

2. *Identification, estimation, and validation of the transient metamodels.* Ideally, we should select the type of the function for each transient metamodel based on physical considerations. However, on practical terms, we usually do that visually, by looking at the representation of the two averaged time-series with respect to time. In fact, it is also this way that a theoretical distribution is frequently fitted to an experimental histogram. A convenient first step is then to represent the scatterplot of the averaged response versus time. To facilitate the identification of the tentative relationships, it is advisable to build a catalog of different functional relationships, with their graphical representations.

   The estimation of the metamodel parameters is then done in two steps: first we fit preliminary metamodels, assuming that the residuals are uncorrelated and using the function *gnls* (or *gls* if the model is linear) of the package *nlme* for the *R Language*; see R Development Core Team (2009) and Pinheiro et al. (2009). Then, we identify candidate ARMA models in an automated way using the function *auto.arima* of the *forecast* package; see Hyndman and Khandakar (2008). This function uses Akaike's information criterion (AIC) to choose the best ARMA(p, q) model up to predefined orders p and q; see Akaike (1974). Finally, knowing the autocorrelation structure of the regression residuals, we incorporate this information obtaining new metamodels, with assumed independent and identically distributed (i.i.d.) residuals with zero mean and finite variance.

   To guarantee that the metamodels adequately approximate the behavior of the simulation model, some validation procedures should be performed. For example, testing the significance of the coefficients (t test), testing the normality of the residuals (Shapiro-Wilk test), and/or testing the hypotheses of uncorrelated residuals (Ljung-Box test).

   It will be necessary to repeat this step if any of the models fail the validation process. Otherwise, the metamodels can be used for the intended purpose.

3. *Identification of an appropriate truncation point.* Comparing the original time series with the fitted transient metamodels, we should be able to identify a suitable truncation point that will remove most (or all) of the initial bias. In fact, we have concluded that for many case studies that were analyzed, a single set of starting conditions will usually lead us to underestimate, or overestimate the actual, or apparent, steady state response level. With two sets of opposing initial conditions, even when the metamodels cross, and/or seem to converge to (slightly) different levels, we can accept that the actual value will lie between the two levels.

4. *Identification, estimation, and validation of the stationary time series model.* The two sections of the simulation output that were not truncated can now be joined together for the final analysis. Again, we need to identify a candidate ARMA model in an automated way using the function *auto.arima* of the *forecast* package. As we refer in step 2, this function uses Akaike's information criterion (AIC) to choose the best ARMA(p, q) model up to predefined orders p and q. Finally, using the above tests, we verify if the residuals can be considered independent and identically distributed (i.i.d.), with zero mean and finite variance.

   It may be necessary to repeat this step if the model fails the validation process. Otherwise, we can proceed to the next step.

5. *Confidence interval estimation.* As a result of our procedure, we are able to present two alternative confidence intervals for the expected system response. The first and most obvious one is based on the method of independent replications. However, since we also fitted an ARMA model to the stationary portion of the output, we can also build another confidence interval based on the coefficients of that time series model.

   Confidence interval based on the 60 independent replications:

$$\text{CI1:} \qquad \overline{\overline{X}}_{r\bullet} \pm t_{n-1;1-\alpha/2} \widehat{\sigma}_{\overline{X}_{r\bullet}} / \sqrt{r},$$

   where $\overline{\overline{X}}_{r\bullet} = r^{-1} \sum_{i=1}^{r} \overline{X}_{i\bullet}$, $\overline{X}_{r\bullet} = n^{-1} \sum_{j=1}^{n} X_{r,j}$, $X_{r,j}$ is the observation at time j in replication r, $t_{n-1;1-\alpha/2}$ denotes the $1 - \alpha/2$ quantile of Student's t-distribution with $n - 1$ degrees of freedom, $\widehat{\sigma}_{\overline{X}_{r\bullet}} = \left[(r-1)^{-1} \sum_{i=1}^{r} (\overline{X}_{i\bullet} - \overline{\overline{X}}_{r\bullet})^2\right]^{1/2}$, r is the number of independent replications, and n is the number of observations considered per run.

Confidence interval resulting from the estimation of the ARMA time series model:

$$\text{CI2:} \qquad \overline{\overline{X}}_{\bullet n} \pm t_{\hat{d};1-\alpha/2}\widetilde{\sigma}_{\overline{\overline{X}}_{\bullet n}}$$

where $\overline{\overline{X}}_{\bullet n} = n^{-1}\sum_{t=1}^{n}\overline{X}_{\bullet t}$, $\overline{X}_{\bullet t} = r^{-1}\sum_{i=1}^{r}X_{i,t}$, $\widetilde{\sigma}_{\overline{\overline{X}}_{\bullet n}} = \left[\widehat{\sigma}_{\varepsilon}^{2}n^{-1}\left(1-\sum_{j=1}^{q}\hat{\theta}_j\right)^{2}\left(1-\sum_{j=1}^{p}\hat{\phi}_j\right)^{-2}\right]^{1/2}$, $\hat{d} = (n/\hat{c}_n)-p-q-1$, $\hat{c}_n = 1+2\sum_{j=1}^{n-1}(1-j/n)\frac{\hat{\gamma}(j)}{\hat{\gamma}(0)}$. If $\hat{d} < 1$, we set the degrees of freedom equal to one, and if $n/\hat{c}_n > n$, we set the degrees of freedom equal to $n-p-q$. For $s = 0, 1, \ldots, q$,

$$\hat{\gamma}(s) = \int_0^{\pi}\frac{\widehat{\sigma}_{\varepsilon}^{2}}{\pi}\frac{\left|1-\sum_{j=1}^{q}\hat{\theta}_j e^{-iwj}\right|^{2}}{\left|1-\sum_{j=1}^{p}\hat{\phi}_j e^{-iwj}\right|^{2}}\cos(sw)\mathrm{d}w,$$

and for $s \geq q+1$, $\hat{\gamma}(s)$ can be obtained from the recursive relationship $\hat{\gamma}(s) = \sum_{j=1}^{p}\phi_j\hat{\gamma}(s-j)$; see Schriber and Andrews (1984). For AR(p) models, the simpler confidence intervals proposed by Sheth-Voss et al. (2005) and Yuan and Nelson (1994) can be used instead.

EXPERIMENTATION

In order to illustrate the use of the procedure, we applied it to two models: an $M/M/1$ queue, for which analytical results are available, and a network of queues with feedback.

The $M/M/1$ Queue

We analyze the number of entities in a stationary $M/M/1$ queueing system, with a utilization factor of $\rho = 0.9$ ($\lambda = 0.9$ and $\mu = 1$). Two different initial conditions were considered: an empty and idle system (no customers waiting, $a = 0$); and 20 customers already waiting at time zero ($a = 20$), in order to induce a large initial transient. For both cases, we performed a Monte Carlo experiment consisting of 30 independent replications with a duration of 3000 time units. The collection of the number of entities in the system started at instant 5 and had an interval width of 10 time units. We then constructed the averaged time-series.
After testing several nonlinear models, the following one showed the best results for both situations:

$$\overline{X}_{\bullet t} = \beta_0 + \beta_1\tanh(\beta_2 t) + u_t, \qquad t = 5, 15, 25, \ldots, 2995, \tag{1}$$

with u_t an ARMA(p,q) satisfying

$$u_t = \phi_1 u_{t-1} + \cdots + \phi_p u_{t-p} + \varepsilon_t + \theta_1\varepsilon_{t-1} + \ldots + \theta_q\varepsilon_{t-q},$$

and where we assume that $\{\varepsilon_t\}$ is a sequence of independent, normally distributed, random variables with mean zero and variance σ_{ε}^2, that is, Gaussian white noise. The parameters that were estimated for both metamodels are shown in Table 1. Only one parameter was not significant (β_0, for $a = 0$). In order to validate the estimated metamodels, we applied the Shapiro-Wilk and the Ljung-Box tests, the latter for 10 lags. No evidence was found to reject the hypotheses of normally distributed, uncorrelated residuals.

Table 1: Metamodel Parameters ($M/M/1$ Queue)

	β_0	β_1	β_2	ϕ_1
$a = 0$	0.923	7.791	0.013	0.903
$a = 20$	19.695	-10.411	0.009	0.945

Table 2: Main Results ($M/M/1$ Queue)

	Estimate	Bias	Rel. Error	Std. Dev.
$a = 0$	8.568	0.432	0.048	1.704
$a = 20$	9.414	0.414	0.046	2.447
PROC	8.921	0.079	0.009	1.174

In Figure 1, we show: the two original data series (thin black for $a = 0$, thin light blue for $a = 20$); the averages of the two sets (dotted dash lines, magenta for $a = 0$, purple for $a = 20$); the representation of Welch's procedure for a window of 100 observations (thick black line); the two transient metamodels (thick blue); Bailey's actual expected response (thick dark blue); the frequently recommended truncation point at 15% of the duration ($t = 445$, the vertical dash line); the averages of the two sets after truncation (dash lines, green for $a = 0$, red for $a = 20$); and, finally, our estimator, the grand average after truncation (orange long dash line). In this case, the transient is relatively short and even the bias for no truncation does not seem to be excessive. A more detailed view is shown in Figure 2. The main results for direct simulation (with $a = 0$ and $a = 20$) and for the proposed procedure (PROC) are summarized in Table 2. Of course, these results are only indicative, because in direct simulation we used 30 runs, while our procedure uses both portions (60 runs).

A Network of Queues with Feedback

This example is the *Model 5: A Network of Queues* from Schruben (1982). It is a network of three capacitated $M/M/s$ queues with feedback (blocked customers must reenter the service queue just completed); see Figure 3. The queue capacity is represented by c, s is the number of parallel servers, each with a service rate of μ, λ is the arrival rate, and p is the probability that a departing customer will follow a particular path. This time, we analyze the total number of entities waiting in the queues.
We performed again a Monte Carlo experiment consisting of 30 independent replications, this time with

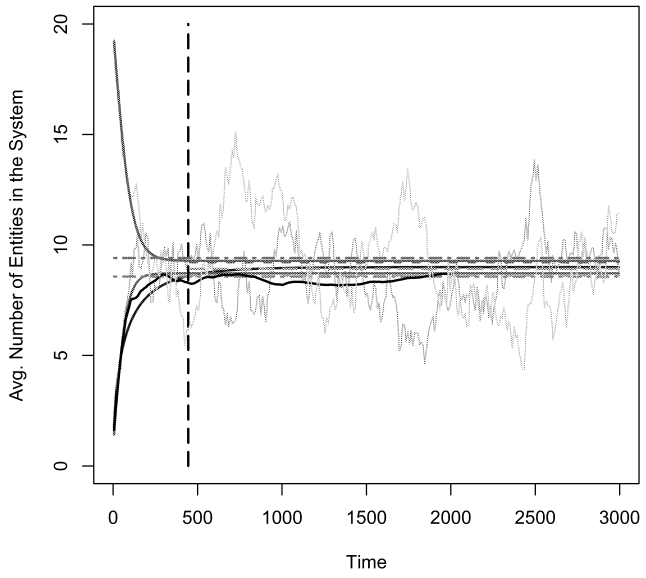


Figure 1: Original Data, Fitted Models (M/M/1 Queue)

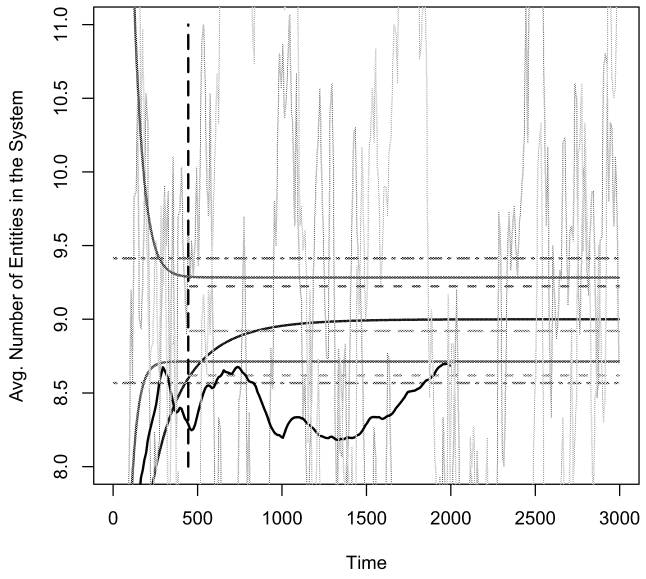


Figure 2: Detailed View (M/M/1 Queue)

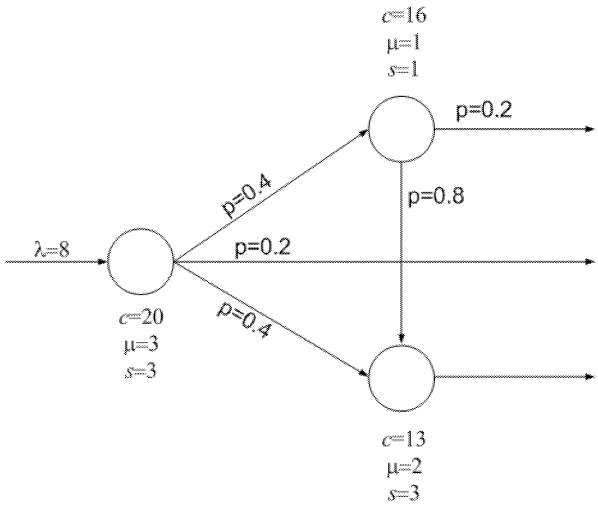


Figure 3: A Network of Queues with Feedback

Table 3: Metamodel Parameters (Network)

	$a = 0$	$a = 20$	
β_1	36.073	49.385	β_0
β_2	7.438	-13.465	β_1
β_3	2.720	0.177	β_2
ϕ_1	1.503	0.495	ϕ_1
ϕ_2	-0.828		
θ_1	-0.691		
θ_2	0.340		

a reference duration of 100 time units. The two distinct starting conditions were: (i) an empty and idle system, with the first arrival being sampled from the exponential distribution having mean 1/8; and (ii) each queue initialized at its maximum capacity c, yielding a total of $a = 49$ entities in the system at instant 0. This time, the number of entities in queues were collected at intervals of 1 time unit, beginning at time 0.5.
For $a = 0$, we fitted a logistic metamodel:

$$\overline{X}_{\cdot t} = \frac{\beta_1}{1 + e^{-\frac{t-\beta_2}{\beta_3}}} + u_t, \quad t = 0.5, 1.5, 2.5, \ldots, 99.5, \quad (2)$$

and, for $a = 49$, the metamodel given by equation (1), but now with $t = 0.5, 1.5, 2.5, \ldots, 99.5$. The parameters that were estimated for both metamodels are shown in Table 3. All parameters were significant, and the application of the Shapiro-Wilk and the Ljung-Box tests showed no evidence to reject the hypotheses of normally distributed, uncorrelated residuals. Figure 4 shows the same curves as Figure 1, except for the analytical expected response, that is not known. This time, we decided to truncate 30% of the duration (twice the recommended value). Again, a more detailed view is shown in Figure 5. The main results for direct simulation (with $a = 0$ and $a = 49$) and for the proposed procedure (PROC) are summarized in Table 4. Since we do

not have analytical results, the deviations are computed with respect to the grand average after truncation.

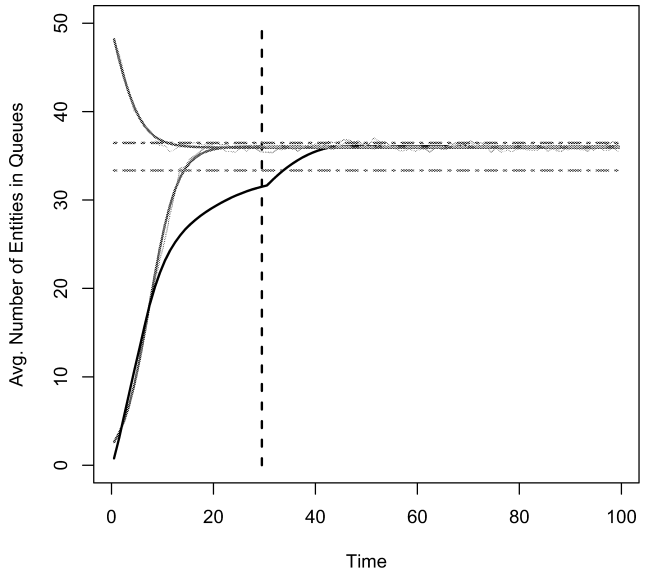


Figure 4: Data and Models (Network)

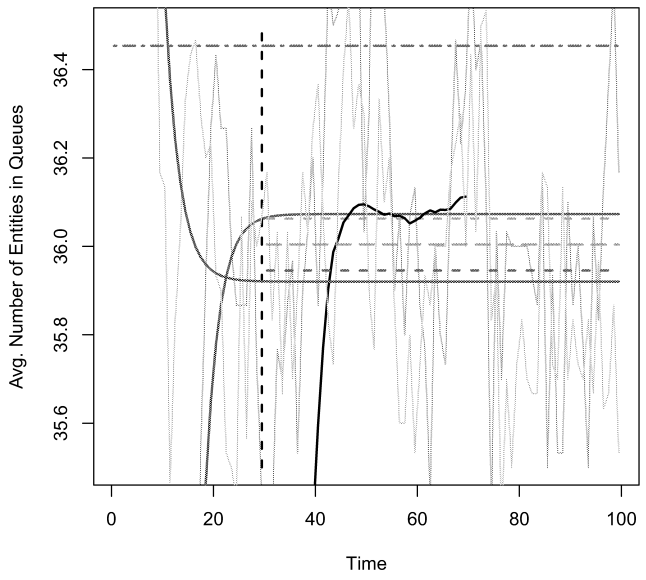


Figure 5: Detailed View (Network)

Table 4: Main Results (Network)

	Estimate	Dev.	Rel. Dev.	Std. Dev.
$a = 0$	33.343	2.661	0.074	7.651
$a = 49$	36.454	0.450	0.012	1.968
PROC	36.004	-	-	0.265

CONCLUSIONS

In this paper, we propose a procedure for coping with the initial bias in stationary simulations, based on regression metamodels for the deterministic component, and time series models for the residuals. The preliminary results for two frequently used case studies look very promising, but a more comprehensive experimentation is needed.

AUTHOR BIOGRAPHY

RITA MARQUES BRANDÃO is an Assistant Professor of the Department of Mathematics at the University of Azores and a member of CEG-IST. Her web address is `<www.uac.pt/~rbrandao>` and e-mail address is `<rita@uac.pt>`.

ACÁCIO M. O. PORTA NOVA is an Associate Professor of the Department of Engineering and Management at the Superior Technical Institute (IST) of the Technical University of Lisbon. His e-mail address is `<apnova@ist.utl.pt>`.

REFERENCES

Akaike H., 1974. *A new look at the statistical model identification. IEEE Trans Autom Contr*, AC, no. 19, 716–723.

Bailey N.T.J., 1957. *Some further results in the non-equilibrium theory of a simple queue. J R Statist Soc*, B, no. 19, 326–333.

Bailey N.T.J., 1964. *The Elements of Stochastic Processes with Applications to the Natural Sciences.* John Wiley & Sons, New York.

Box G.E.P.; Jenkins G.M.; and Reinsel G.C., 1994. *Time Series Analysis: Forecasting and Control.* Prentice Hall, Englewood Cliffs, NJ, 3rd ed.

Brandão R.M. and Porta Nova A.M.O., 2009. *Analysis of nonstationary stochastic simulations using time-series models. ACM Trans Model Comput Simul*, 19, no. 2, 1–26.

Gafarian A.V.; Ancker C.J.; and Morisaku T., 1978. *Evaluation of commonly used rules for detecting steady-state in computer simulation. Naval Research Logistics Quarterly*, 25, 511–529.

Hoad K.; Robinson S.; and Davies R., 2008. *Automatic warm-up lenght estimation.* In S.J. Mason; R.R. Hill; L. Mönch; O. Rose; T. Jefferson; and J.W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference*. IEEE, 532–540.

Hyndman R.J. and Khandakar Y., 2008. *Automatic time series forecasting: the forecast package for R. Journal of Statistical Software*, 27, Issue 3.

Pawlikowski K., 1990. *Steady-state simulation of queueing processes: a survey of problems and solutions. ACM Computing Surveys*, 22, 123170.

Pinheiro J.; Bates D.; DebRoy S.; Sarkar D.; and the R Core team, 2009. *nlme: Linear and Nonlinear Mixed Effects Models.* R package version 3.1-96.

R Development Core Team, 2009. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. URL `http://www.R-project.org`. ISBN 3-900051-07-0.

Schriber T.J. and Andrews R.W., 1984. *ARMA-based confidence intervals for simulation output analysis. Amer J Math Manage Sci*, 4, no. 3 & 4, 345–373.

Schruben L.W., 1982. *Detecting initialization bias in simulation output. Operations Research*, 30, no. 3, 569–590.

Sheth-Voss P.A.; Willemain T.R.; and Haddock J., 2005. *Estimating the steady-state mean from short transient simulations. Eur J Oper Res*, 162, 403–417.

Wilson J.R. and Pritsker A.A.B., 1978a. *Evaluation of startup policies in simulation experiments. Simulation*, 31, 79–89.

Wilson J.R. and Pritsker A.A.B., 1978b. *A survey of research on the simulation startup problem. Simulation*, 31, 55–58.

Yuan M. and Nelson B.L., 1994. *Autoregressive-output-analysis method revisited. Ann Oper Res*, 53, 391–418.

SIMULATION TOOL FOR MORPHOLOGICAL ANALYSIS

Alexandra Fronville
Université européenne de Bretagne
UBO, LISyC – EA3883
CERV -F- 29280, France

Fabrice Harrouet
Université européenne de Bretagne
ENIB, LISyC – EA3883
CERV -F- 29280 France

Anya Desilles
CREA, Polytechnique
ENSTA , Paris, France

Pierre Deloor
Université européenne de Bretagne
ENIB, LISyC – EA3883
CERV -F- 29280 France

KEYWORDS

Morphological Analysis, Virtual Reality, Biological Multi-Agent System, Mathematical Programming, Scientific Visualisation Software.

ABSTRACT

To understand the mechanisms underlying the morphogenesis of multicellular organisms we study the dynamic system of cells (cell multiplication, cell migration, apoptosis); local interactions between cells for understanding the convergence of the system to a stable form that is constantly renewed.and the controls established by the nature of the growth of the organism, and its convergence to a stable form.

We must be able to formalize it in a proper metric space a metaphor of cell dynamics to find conditions (decisions, states) in which operational constraints (such as those induced by the tissue or the use of resources) are always satisfied and therefore in which the system is viable and maintain its shape while renewing.

The aim of this paper is to explain the mathematical foundations of this work and describe a simulation tool to study the morphogenesis of a virtual organism and to describe a simulation tool to study the morphogenesis of a virtual multicellular organism. We formalize mathematically a model of cell dynamic on the principles of morphological analysis. Morphological analysis and viability theory are the mathematical foundations that motivate this work and this tool will test whether a system generated by morphological equations can maintain its shape and remains "viable" in a given environment.

INTRODUCTION

In biology, recent techniques in confocal microscopy allow experimental data on cellular dynamics and its importance in the evolution of biological shapes to be gathered. The development of an organism is an evolutionary process, not deterministic of different cells in an environment. The cells evolve and change the organism on which they act, which is changing the environment that feeds back on the cells. It is what we call co-evolution, i.e. the system changing its environment, which in turn changes the system. Epigenetics considers this coupling between organism and environment and can not be ignored in understanding the development of living(Varela, 1979).

To understand the mechanisms underlying the morphogenesis of multicellular organisms we study the dynamic system of cells (cell multiplication, cell migration, apoptosis); local interactions between cells for understanding the convergence of the system to a stable form that is constantly renewed.

Many studies already model the development of organisms, cellular automata (Forest, 2005; Hogeweg and Marée, 2001; Pena and Duthen, 2007; Graner and Glazier, 1992; Ballet et al., 2009), L-systems (Prusinkiewicz and Lindenmayer, 1990) based on a formal grammar, iterated function systems (IFS) (Siepinski sieve and fern) (Gentil et al., 2006) and finally, we use multi-agent systems to model complex systems. they offer the possibility to simulate a number of autonomous components in an environment in order to ascertain the nature of the phenomenon studied in its entirety, without central control (Stoma et al., 2007; Doursat, 2007).

The problem is that simulations using multi-agent systems approach are very difficult to formalize and to study theorically, they set problems of convergence and stability (Bonneaud et al., 2009). To study the shapes generated by multi-agent modeling using morphogenetic principles, we must be able to formalize in a metric space the cellular dynamics to study mathematically the conditions of growth,

convergence and stability of the processed shape.

In mathematics, the viability theory (Aubin, 1991) offers concepts and methods to control a dynamic system in a given fixed environment, in order to maintain it in a set of constraints of viability. This is mainly to elicit the underlying feedbacks that regulate the system and discover the mechanisms of selection for implementation. The development of mutational analysis (Aubin, 2000) was motivated because dynamic systems theory and viability theory are not suitable for studying a cell system that grows and multiplies. The concept of differential equation has been extended to the concept of mutational equation in a metric space. Morphological equations, special type of mutational equation, have properties similar to those of differential equations (Peano theorem, Cauchy-Lipschitz, Nagumo) (Lorenz, 2010). They govern the evolution of sets in the same way as differential equations govern the evolution vectors, and are used to investigate conditions (decisions, states) in which a morphological equilibrium will be maintained and where operational constraints (such as those induced by the environment or resource use) are always satisfied and in which the system will be sustainable.

If an evolutionary system is given, the viability theorem states how it governs the evolution of a viable state in a fixed environment. When the system evolves the environment changes, and this changes influence the system. This is what biologists mean by co-evolution. Mathematically, it is the joint evolution of states and sets to which they must adapt. In this case, the environments are changing under the action of a morphological equation, evolutionary systems are governing the evolution of the states and of the environment, and they depend on both the state and the environment. This is called a differential-morphological system. For such a differential-morphological system to have solutions we have to adapt the viability theorem to the differential-morphological systems. This means that there is at least one co-viable evolution of the state and the environment based on each state-environment pair.

The set of conditions for which at least one solution is viable is called viability kernel. In the case of a multicellular organism that evolves, the cells evolve with the organism and with the environment of the organism. To be viable these three levels will have to be able to react appropriately to events. Therefore anticipating the time when the state of the cell, the body reaches its limits of viability, i.e. determine their viability kernel.

Thinking of morphogenesis in this way brings new requirements, particularly in mathematics and computer science to implement efficient mutational algorithms able to inform us about the mechanisms used by multicellular organisms to survive. Their structure could be indicative of internal consistency mechanisms of morphogenesis in living organisms.

The definition of the viability kernel remains difficult: specific algorithms have been developed however their application requires an exponential memory space with the dimension of space, and the outcome is difficult to handle. The viability theory provides tools and methods to control a dynamic system in order to keep it in a set of eligible states, called the set of constraints(Aubin, 1991). Mutational algorithms have yet to be conceived.

In this context, the aim of this work is to formalize mathematically a model of cell dynamic on the principles of morphological analysis and to describe a simulation tool for studying morphogenesis of virtual multicellular organisms. Morphological analysis and viability theory are the mathematical foundations that motivate this work and this tool will test whether a system generated by morphological equations can maintain its shape and remains "viable" in a given environment.

MORPHOLOGICAL DYNAMIC OF CELLULAR TISSUE EVOLUTION

The purpose of this paragraph is to formalize in the context of mutational and morphological analysis, the evolution of cellular tissues during embryogenesis. This question motivates the study of a **discrete morphological dynamics** governing the evolution of tissues.

During an infinitesimal change of tissue, each element of the form is not only "move" to another point of the form that follows it, but eventually moved and "multiplied" when multiple daughter cells succeed to this element, multivalent character which leads to the concept of speed form (Aubin, 2000).

Figure 1: Univalued analysis to formalize a cell that moves

During embryonic development, the confinement is imposed by the cohesion of tissues and the presence of an envelope, such as the epithelial layer covering the embryo. There is a co-evolution of the cellular membrane and the dynamics of each cell, confinement shapes that can evolve only by respecting the constraints that we want to study using morphological analysis.
In biological morphogenesis, the vitellius is the energy reserves used by the embryos during embryonic development.

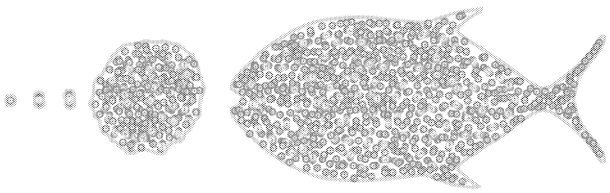


Figure 2: Multivalued analysis to formalize a cell that multiplies and moves

M denotes the set of containment cells, contained in the complement of vitellius.

$K \subset \mathbb{R}^3$ representing tissue cells, the cells are designated by $x \in K \subset \mathbb{R}^3$.

If we restrict morphogenesis in the plan,

$$(D) := \{(0,1),(0,-1),(-1.0),(1,0)\}$$

denotes the set of 4 planes directions and $\overline{\mathcal{D}} := \mathcal{D} \cup \{(0,0)\} \cup \emptyset$ means the 6 "extended " directions

For morphogenesis in the space $\mathbb{R}^3$,

$$\mathcal{D} := \{(0,0,1),(0,0,-1),(0,1,0),(0,-1,0),(-1,0,0),(1,0,0)\}$$

denotes the set of six directions and $\overline{\mathcal{D}} := \mathcal{D} \cup \{(0,0,0)\} \cup \emptyset$ means the eight "extended" directions.
We denote by $A + \emptyset = \emptyset$ in the max-plus algebra for the operations $\cup$ and $+$.

We will note $\Xi_M(K,x) := \{u \in \mathcal{D}$ such that $x + u \in \{x\} \cup (M \setminus K)\}$ et $R_M(K,x) := \Xi_M(K,x) \times \Xi_M(K,x)$.

Then we introduce the correspondence $\Psi(x,u,v) := \{x + u\} \cup \{x+v\}_{(u,v)\in R_M(K,x)}$.

The morphological dynamic Φ_M is then defined by

$$\Phi_M(K) := \bigcup_{x\in K} \bigcup_{(u,v)\in R_M(K,x)} \Psi(x,u,v) \qquad (1)$$

And the discrete morphological dynamic $K_{n+1} = \Phi_M(K_n)$.

This gives the different cases of cell behavior:

1. *apoptosis* , obtained by taking $(\emptyset,\emptyset) \in R_M(K,x)$ since $\Psi(x,\emptyset,\emptyset) := \emptyset \cup \emptyset = \emptyset$
2. *migration* by taking $u \in \mathcal{D}$ et $v = \emptyset$ or $u = \emptyset$ and $v \in \mathcal{D}$ or further $u = v$
3. *stationarity*, which is a migration obtained by taking u and v equal to $(0,0,0)$
4. *cell division* by taking $u := (0,0,0)$ et $v \in \Xi_M(K,x)$ (or otherwise)
5. *division and migration* by taking $u \in \Xi_M(K,x)$ and $v \in \Xi_M(K,x)$

We can now introduce the equivalence relation on the directions

$$u \equiv_x v \text{ if and only if } x + u = x + v$$

which we denote by μ and ν the representatives, noting that by construction, for every pair (μ,ν) the equivalence class, for all $u \in \mu$ and $v \in \nu$, $\Psi(x,\mu,\nu) = \Psi(x,u,v)$ does not depend on the choice of directions belonging to equivalence classes.

Because two cells can not occupy the same position, just select at most one extensive direction in each class.

The correspondence of regulation is defined by the quotient set :

$$\Theta_M(K,x) := R_M(K,x)/\equiv_x \qquad (2)$$

The morphological dynamics Φ_M is always defined by

$$\begin{aligned} \Phi_M(K) := & \bigcup_{x\in K} \bigcup_{(\mu,\nu))\in\Theta_M(K,x)} \Psi(x,\mu,\nu) \\ = & \bigcup_{x\in K} \bigcup_{(u,v)\in\in R_M(K,x)} \Psi(x,u,v) \end{aligned} \qquad (3)$$

In the case of a discrete dynamics, it is defined by control sequences (u_n, v_n) associated to K_n to define $K_{(n+1)}$.

Implementation of the algorithm is equivalent to setting the viable directions.

TOOL FOR MODELLING MORPHOGENETIC BEHAVIOUR

This section presents the tool developed for modelling of morphogenetic phenomena.

We set as a basic principle that cells are autonomous agents. The cell perceive changes in the environment and can change its dynamic.

In addition, the cells are autonomous by ignorance of the whole system because the reductionist method does not predict

the evolution of the whole system. The principle that cells must be autonomous is set as a basic rule.

The platform was created to understand morphogenesis as the theoretical basis for morphological analysis. The program is implemented (applied) in C++ using the Workshop of Virtual Reality AREVI (Reignier et al., 1998), it is a simulation library of autonomous entities and 3D rendering.

The order of scheduling has a significant impact on the results of the simulation (Lawson and S.Park, 2000), (Bonneaud et al., 2009). Different behaviours can be observed in virtual models depending on the manner (type) of scheduling used. In nature morphogenesis shows us robust forms despite autonomous cells. To better appreciate and understand the mechanisms that are put into play in morphogenesis, we wanted to retain flexibility in the scheduling of cells. The program has two different modes of simulation, a stochastic mode and a controlled mode. The cells represented on screen by spheres can proliferate in a discrete environment (cellular automaton) or in a "continuous" one. In the latter case the movement of cells are more precisely described. Distance between the cells can vary. It represents the forces of attraction or repulsion between them.

The behaviour of cells is not the same in each type of simulation. A simple graphical interface has been implemented in order to select the features of the simulation. It allows dynamic change of parameters and selecting mechanisms (e.g. apoptosis, differentiation) that are active/inactive during the simulation. A number of parameters were taken into account to test their relative influence on the forms generated by populations of cells.

Options are available to allow choice between 2D/3D, discrete or continuous simulations. The size and shape of both the environment and the cells can also be defined and adjusted, as can cell behaviour such as apoptosis, the direction of mitosis etc.

In the case of continuous simulation, each cell can perceive its neighbors within a radius of attraction and evaluate the stresses:

- by the neighbors.
- by the membrane containment.

Constraints are crucial for evolution of the cell; if it is too strong, the cell is not viable as it can no longer divide. A maximal constraint parameter sets up a threshold below which the cell remains viable.

The notion of coercion has no place when cells are represented in a grid. To account for the influence of the environment, a parameter is defined as the maximum number of cells that a cell is able to force when it divides. When the current strain of the cell is greater than the maximum stress threshold, the cell can no longer divide. Two modes of mitosis have been considered; firstly the cell chooses to divide in the direction where the stress is less intense; secondly where the direction of cell division is predetermined. However in both cases, if the spatial constraints of the current cell exceed the maximum stress threshold, it cannot divide.

It is also possible to assign an amount of energy to each cell. The basic idea is: consider that a cell has a store of energy assimilate from its environment. A percentage of the store is used to maintain structure and growth. The remaining its reserve is used for maturation (e.g. maintenance of the immune system) and reproduction. In very simple terms initially we want to apply this principle. A level of energy is mapped and associated with virtual cell application. A small amount of energy representing cell maintainance of its structure is logged at each step of the simulation. We consider that a cell uses a lot of energy in reproduction - during mitosis this energy level is divided by two. The cell dies when the energy level becomes too low. The behaviour of a cell is directly related to the quantity of energy contained and it is possible to obtain forms of very different population of cells by modulating certain thresholds, as detailed below. A cell can recover energy if it is in contact with a relevant part of the environment.

An option of the application allows cells to differentiate. In this case, cells that are not the same type (represented by different colors) have different dynamics. A specific cell can differentiate when under stress. We wanted to demonstrate this in connection with the spatial constraints of the cells. Stress corresponds to a large differential spatial constraint between two consecutive measurements. A threshold defines the minimum value of the interval for which the cell differentiates. It is also possible to define a numerical value for stress necessary to induce differentiation.

Another control is the direction of cell division. It is possible to define the direction that mitotic cells take in advance and the order of selection. This parameter can also be chosen as a random option. The morphogenesis changes when varying the choice of these directions. Real time morphogenesis film has produced data that demonstrates the features of the direction of cell division.

Parameter values modulate cell activity. The steps of the

algorithm define cell behaviour and are the same for both discrete and continuous cases, as we have seen, the stress calculations are different.

TEST

As described in the preceding paragraph, by varying the parameters the application offers the possibility to make different types of simulation.

In collaboration with Nadine Peyrieras (Melani et al., 2007; Campana et al., 2008), we compared the behaviour of our model with the first segmentation of zebrafish cells. (See Figure 3).

The model shows the first segmentation of the small fish up to 1,000 cells, then the model cannot be used further because the dynamic of certain cells has changed. To enable biologists to continue to advance understanding on the establishment of the dorsoventral axis of the zebrafish, it is important to elucidate the cellular dynamics.

This question leads us to examine the outcome of differentiated cells. This motivated the development of morphological analysis to control cell dynamics and the creation of a simulation platform to visualize and compare with biological data.

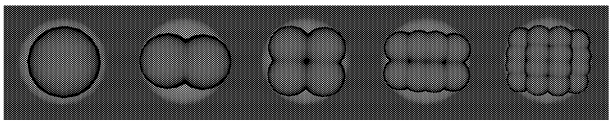


Figure 3: First simulated segmentation of the zebrafish

We also tried to observe spacial constraints when Vitellius is half covered (see Figure 4) to understand the changing dynamic of the cells that form the backbone of the fish.

To better understand morphogenesis, and to overcome obstacles in understanding the influence that the dynamics have on the shape of the organism, we voluntarily limit to discrete simulation by controlling the order of division and ordering executions. Here we have chosen to present a number of simulations by choosing modes of division and as a result of mitosis different directions to observe the impact of these parameters on the shapes of cell populations (see Figure 5). The study is still in its infancy but it is fundamental for undrestanding the mechanisms controlling morphogenesis.

CONCLUSION

The main contribution of the paper is the mathematical formalisation of cell dynamic on the principles of mor-

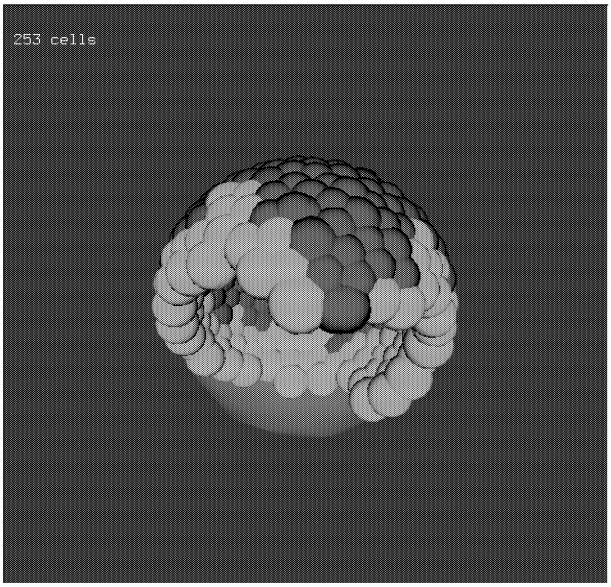


Figure 4: Spaces constraints for the Zebrafish

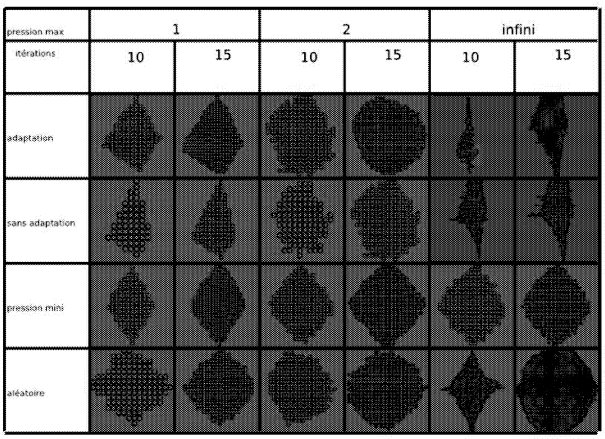


Figure 5: 2D-Shape dictionnary

phological analysis and the construction of a platform of virtual reality to experiment this dynamic.
This beginning of mathematical formalization of cellular dynamics using mutational analysis has guided this work and the development of this platform modeling. Similarly, the use of simulation tools has guided us in our questioning on morphogenesis.

Among the important issues that have emerged, two issues have attracted particular attention: the robustness of biological forms and the equilibrium of the shape. How cells whose dynamics is simple (mitosis, apoptosis, migration) can maintain form while continually renewing itself (homeostasis), and how despite environmental disruption during embryonic development, the shape stay stable.

Morphological analysis is an original way to formalize the

cell dynamic and to better understand the retro-actions implemented by the cells to maintain their viability and the organism's viability.

References

Aubin, J.-P., 1991. Viability theory. Birkhauser.

Aubin, J.-P., 2000. Mutational and morphological analysis: tools for shape regulation and morphogenesis. Birkhauser.

Ballet, P., Tripodi, S., Rodin, V., Sep. 2009. Morphoblock programming: a way to model and simulate morphogenesis of multicellular organisms. Journal of Biological Physics and Chemistry.

Bonneaud, S., Redou, P., Desmeulles, G., Chevaillier, P., 2009. Biais computationnels dans les modèles de peuplement d'agents. In: JFSMA 09.

Campana, M., Rizzi, B., Melani, C., Bourgine, P., Peyriéras, N., Sarti, A., 2008. A framework for 4-d biomedical image processing, visualization and analysis. In: Bobbitt, R., Connell, J. H., Flickner, M., Haas, N., Hampapur, A., Harris, D., Kurtz, C., Lloyd, B., Otto, C., Pankanti, S., Park, U., Payne, J. (Eds.), Retail Vision-Based Self-checkout - Exploring Real Time Real Purpose General Vision System. pp. 403–408.

Doursat, R., october 2007. Organically grown architectures: Creating decentralized, autonomous systems by embryomorphic engineering, organic computing.

Forest, L., 2005. Modèles de morphogénése tissulaire à partir de dynamiques cellulaires intégrées : Application principale à la croissance radiale secondaire des coniféres. Ph.D. thesis, Unversité Joseph Fourier.

Gentil, C., Tosan, E., Neveu, M., 2006. Geometric modelling with ifs. In: Third International Conference of Applied Mathematics. URL `http://liris.cnrs.fr/publis/?id=3181`

Graner, F., Glazier, J., 1992. Simulation of biological cell-sorting using two-dimensional extended potts model. Phys. Rev. Lett. 69, 2013–2016.

Hogeweg, P., Marée, S., 2001. How amoeboids self-organize into a fruiting body : multicellular coordination in dictyostelium discoideumm. Proc. Natl. Acad. Sci. U.S.A. 98 (7), 3879–3883.

Lawson, B., S.Park, 2000. Asynchronous time evolution in an artificial society mode. Journal of Artificial Society and Social Simulation 3 (1).

Lorenz, T., 2010. Mutational Analysis A Joint Framework for Cauchy Problems In and Beyond Vector Spaces. Springer.

Melani, C., Peyriéras, N., Mikula, K., Zanella, C., Campana, M., Rizzi, B., Veronesi, F., Sarti, A., Lombardot, B., Bourgine, P., 2007. Cells tracking in the live zebrafish embryo. In: 29th Annual International Conference of the IEEE.

Pena, A. C., Duthen, Y., 2007. An artificial development model for cell pattern geneation. Austrlian Conference on Artificial Life.

Prusinkiewicz, P., Lindenmayer, A., 1990. The algorithmic beauty of plants. The Virtual Laboratory. New York: Springer-Verlag. XII.

Reignier, P., Harrouet, F., Morvan, S., Tisseau, J., Duval, T., 1998. AReVi: A virtual reality multi-agent platform. Lecture Notes in Computer Science 1434, 229–240.

Stoma, S., Chopard, J., Godin, C., Traas, J., 2007. Using mechanics in the modelling of meristem morphogenesis. 5th International Workshop on functional-structural plant models, Napier, New-Zeland 52, 1–4.

Varela, F., 1979. Principles of biological autonomy.

AUTHOR BIOGRAPHY

ALEXANDRA FRONVILLE was born in Evian, France and went to the university Pierre et Marie Curie in Paris, where she studied mathematics and obtained her PhD in 1986. After a post-doctoral position at INRIA, she works for a couple of year for the university of Brest at Computer Sciences for Complex Systems Laboratory and in the European center for virtual reality. Her research focuses on viability theory and morphological analysis applied to complex systems.
E-mail: alexandra.fronville@univ-brest.fr

FABRICE HARROUET was born in Nantes, France, and went to the ENIB engeneering school in Brest, where he studied computer science and obtained his PhD en 2000. He works as a lecturer in this school and does his research at Computer Sciences for Complex Systems Laboratory and in the European Center for Virtual Reality in Brest. His research focuses on interactive multiagent simulations; this concerns parallel computation and 3D rendering.
E-mail: harrouet@enib.fr

ANNA DESILLES was born in Kharkov, Ukraine and went to the university Denis Diderot in Paris, where she obtained her PhD in 2000. After a teaching position at EISTI (International School of Information Processing, Cergy, FRANCE), she joinded a groop of researchers working on the viability theory. Actually, she works at the Applied Mathematics department of ENSTA. Her research focuses on on numercial algorithms for viability theory.
E-mail: anna.desilles@ensta-paristech.fr

PIERRE DE LOOR is Professor at ENIB (Ecole Nationale d'Ingénieurs de Brest, France). He's the responsible of the team ARéVi at the LISyC (Computer Science Laboratory for Complex System). He works on relations between artificial intelligence, virtual reality and cognitive sciences. He is particularly interessed in the enactive stance who considers biological and epigenetic conditions as the ground of autonomy and cognition.
E-mail: deloor@enib.fr

MATCHING HIDDEN NON-MARKOVIAN MODELS: DIAGNOSING ILLNESSES BASED ON RECORDED SYMPTOMS

Claudia Krull and Robert Buchholz and Graham Horton
Otto-von-Guericke University
P.O. Box 4120, 39016 Magdeburg, Germany
E-mail: {claudia.krull|robert.buchholz|graham.horton}@ovgu.de

KEYWORDS
Decision support systems, Markov-chain, Stochastic, Time series analysis, Health care

Abstract

Discrete stochastic models (DSM) can be used to accurately describe many natural and technical processes. The simulation algorithms usually require the system parts of interest to be completely observable in order to analyze the model. Hidden non-Markovian models (HnMM) have been applied successfully to the analysis of partially observable systems. They can determine the unobserved most likely system behavior that caused an observed output. The analysis can be done by the state space-based Proxel algorithm, which on-the-fly generates the reachable model state space at discrete points in time. In the current paper, we compute the unconditional probability of a given model having produced a given output. This can be used to find the most likely one of different possible system configurations to produce the given output. In our application we want to find the illness that most likely caused the recorded symptoms of a patient. Experiments are performed to determine the accuracy and limitations of the applicability of the approach. This paper increases the application area of HnMM analysis twofold. We can now perform model matching tasks for HnMM, and we have tested an application example from medical diagnosis.

INTRODUCTION

Discrete stochastic models (DSM) are widely used in industry today and can represent many manufacturing, natural, technical and other processes. Usually simulations are performed with fully parameterized models, which require a known and fully observable system. However, some real systems are not fully observable, only through their interaction with the environment. The internal state of the machine or process is not directly detectable, but the system generates observable output depending on the internal state. These models can be characterized as partially observable systems (Buchholz et al. 2010). Hidden Markov models (HMM) can model and analyze hidden systems, but they are restricted to discrete-time Markov chains (DTMCs) and thus to memoryless processes. This only allows for a very rough approximation of the runtime behavior of many real processes. The recently developed hidden non-Markovian models strive to relieve this problem by enabling the formal description of partially observable discrete stochastic systems with time-dependent processes (Krull and Horton 2009).

We have proposed using the Proxel algorithm (Horton 2002, Lazarova-Molnar 2005) for analyzing HnMM, which is based on the method of supplementary variables. The Proxel algorithm explores all possible system developments in given discrete time steps and quantifies them with their probability. Recent research (Buchholz et al. 2010) has shown that by using Proxels one can compute the most likely system behavior that produced a given output, when the specification of the HnMM is known. But what if several model configurations are possible? An example is a medical diagnosis system, which is described in detail later in this paper. The symptoms of a patient can be regarded as the output of a hidden model of the illness progression. However, it is not known which illness caused the symptoms, and thus the task is to find the most likely illness to have caused the observed symptoms. The question posed is now: Given a patient's symptoms and their time of occurrence or detection, what is the most likely illness he or she is suffering from? This can then help the physician to determine a promising treatment to apply or a medication to administer to the patient.

The abstract task is to determine the probability of a given model, given a specific sequence of output symbols. This corresponds to the model matching used in pattern recognition, a major application areas of HMM. Enabling the solution of this task for HnMM will broaden the number of applications that HnMM can be used for and increase their practical applicability.

STATE OF THE ART IN HNMM AND THEIR ANALYSIS

Hidden non-Markovian Models (HnMM) are an extension of Hidden Markov Models (HMM) (Fink 2008). The main enhancement of HnMM is the inclusion of time behavior and the shift of focus from the states to

the state transitions, since these are often the objects of interest in discrete stochastic systems. This also required shifting the symbol emissions from the states to the state transitions. The time-dependent transitions of a HnMM are described by continuous distribution functions, such as Normal or Weibull.
An HnMM can be described by a 6-tuple (S, C, V, A, B, Π), with the set of states S, the set of state transitions C, the set of output symbols V, the time-dependent transition matrix $A(t)$, the emission probability matrix B and the initial probability vector Π. This formal description is derived from HMM and has been adapted for HnMM, to allow for any kind of discrete stochastic models as hidden system description (Krull and Horton 2009).
The time dependence was also incorporated in the output symbol sequence O (trace) by attaching a time stamp to each symbol emission. The internal system behavior of a HnMM can be described by the sequence of state changes Q (path) with the corresponding time stamps. The path may be longer than the trace, because not every state change has to result in a symbol emission; however, every symbol emission is caused by a state change. This last condition has to be relaxed to reflect the medical diagnosis application example.

Proxel-based Analysis of HnMM

To analyze the newly developed paradigm of HnMM, we have adapted the original HMM algorithms to the new requirements Krull and Horton (2009). This was only possible for specific model properties, for example requiring the models to regenerate after every state transition. Since we developed a general modelling paradigm we are also interested in general analysis algorithms. One promising candidate is the state space-based Proxel algorithm (Horton 2002, Lazarova-Molnar 2005).
The algorithm tracks all possible system developments in discrete steps creating so-called probability elements (Proxels) and discovers possible development paths. Furthermore, it turns a model containing arbitrary continuous distribution functions into a DTMC (Bolch et al. 1998). This analogy to path analysis of HMM led us to apply Proxel-based analysis to HnMM.
The analysis of HnMM using Proxels has been described for example in (Buchholz et al. 2010). The task in that paper was to determine the most likely hidden behaviour that caused a given output trace, given a system with known specification. This is also known as the decoding task in HMM Fink (2008).

APPLICATION EXAMPLE: PATIENT DIAGNOSIS

In this paper we want to test the application of HnMM analysis to a field outside of engineering and manufacturing. The example used throughout the paper is the diagnosis of a patient's illness based on a sequence of recorded symptoms of that patient. We specified two different types of illness models: one describing the progression of an abstract illness and the other describing the progression of the common cold or an influenza infection. In this section we introduce the specification of the HnMM for that application. The basic time unit assumed in all of the models is one day.

Abstract Illness We assume that an abstract illness can progress in up to three successively severe stages, healing is possible in every stage. Each stage lasts for a random amount of time, described by an arbitrary continuous probability distribution. The time to healing depends on the total illness duration and is also random. Tests are performed every day: a blood test, urine tests and taking the patients temperature. Blood test and urine test can result in a negative, inconclusive or positive result and the patient may or may not have fever. Each stage has different probabilities for exhibiting each of the symptoms, the more probable the symptoms, the more severe the illness stage.

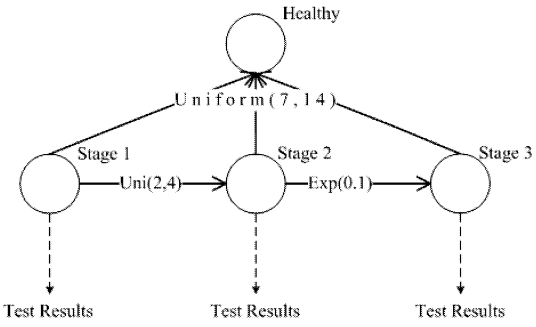


Figure 1: An HnMM Describing the Progression of an Abstract Illness and the Resulting Symptoms

Different abstract illnesses are characterized by different stage models, with individual stage durations and individual probabilities for causing certain symptoms in each stage. Figure 1 shows an HnMM describing the progression and symptom probabilites for one such abstract illness. Tables 1 and 2 show the stage durations and symptom probabilites for each of the three illnesses. While disease one has different symptom probabilities from the other two, disease two and three can only be distinguished by their stage durations.

Cold and Flu We assume, that the common cold and an influenza infection progress in two stages before healing, the first phase being the more severe of the two. We have chosen to consider fever, cough and body aches as the symptoms to be recorded, each of these will be recorded every day. An influenza infection has a more severe and a more prolonged first infection stage. The common cold has a milder progression indicated by

Table 1: Symptom Probabilities and Stage Duration of Disease One

		Stage 1	Stage 2	Stage 3
Fever	yes	0.1	0.5	0.8
	no	0.9	0.5	0.2
Urine	negative	0.8	0.6	0.5
	inconclusive	0.1	0.3	0.3
	positive	0.1	0.1	0.2
Blood	negative	0.7	0.5	0.2
	inconclusive	0.2	0.3	0.3
	positive	0.1	0.2	0.5
	Duration	U(2,4)	Exp(0.1)	U(7,14)

Table 2: Symptom Probabilities and Stage Duration of Diseases Two and Three

		Stage 1	Stage 2	Stage 3
Fever	yes	0.3	0.7	0.9
	no	0.7	0.3	0.1
Urine	negative	0.8	0.7	0.7
	inconclusive	0.2	0.2	0.2
	positive	0.0	0.1	0.1
Blood	negative	0.7	0.5	0.2
	inconclusive	0.2	0.3	0.3
	positive	0.1	0.2	0.5
Duration disease two		N(3,0.5)	Exp(0.3)	N(10,2)
Duration disease three		N(6,1)	Exp(0.3)	N(12,2)

smaller probabilities to develop the symptoms; cough, fever and aches. The two state models for influenza and common cold are shown in Figures 2 and 3.

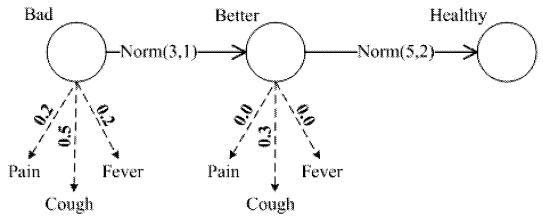


Figure 2: An HnMM Describing the Progression of the Common Cold and the Resulting Symptoms

We assume that the models describing the progression of these illnesses are known. Question: Based on the recorded symptoms of a specific patient, what is the illness that he is most likely suffering from? To answer this, the Proxel algorithm has to be adapted slightly.

Adaptions for Diagnosis Example - Decouple Emissions from State Changes

One difference of the above HnMMM to the definition from (Krull and Horton 2009) is that the state transitions of the illness progression do not cause symbol output. It is easy to reason that the progression from one stage of the illness to another does not cause a blood test. The symptom probabilities depend on the stage of the illness, which means that symbols are emitted based on the current state of the system, as in the original HMM paradigm (Fink 2008). The difference to HMM

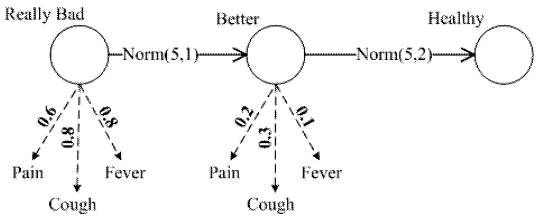


Figure 3: An HnMM Describing the Progression of an Influenza Infection and the Resulting Symptoms

is, that the emitted symbols in our case still have a time stamp, which can take on any value. Similarly, a detected symbol emission does not cause or indicate a state transition, effectively decoupling them.
The Proxel algorithm can be used in its original specification. The necessary adaption is that when a symbol is detected, the set of Proxels for the corresponding time step is modified as follows:

- Proxels that represent discrete states which could not have emitted the given symbol are discarded.
- The probabilities of the remaining Proxels are scaled by the particular symbol's emission probability in the discrete state of each Proxel.

The resulting reachability graph contains all reachable model states, now under the condition of having observed the given symbol sequence. When asking for the most likely model to have produced a given symbol sequence, we no longer need the information of the specific generating path. Therefore the path information can be omitted from the Proxel, speeding up the analysis algorithm considerably. The next section describes the use of the result of this Proxel algorithm in model matching.

MODEL MATCHING IN HNMM

Model matching for HnMM can be performed in the same way as speech recognition using the Forward algorithm in HMM (Fink 2008). In speech recognition, word models are assigned a probability to have produced a recorded piece of speech. The most probable word model is the most likely meaning of the audio recording. The modified Proxel algorithm can also compute the probability with which a given model has produced a given output sequence. The model with the largest probability should be the most likely model to have caused a given sequence.
In the application example, the Proxel algorithm can compute the probability with which a given sequence of symptoms is caused by a specific illness. This probability corresponds to the sum of the unconditional probabilities of all Proxels valid at the end of simulation time. These Proxels represent all possible end points of the patient's illness, under the condition of observing the given sequence of symptoms.

The unconditional probabilities of the different illness models to have produced the given symptom sequence can be compared and the illnesses ranked according to their likelihood. We assume that the largest probability represents the most likely illness to have caused these symptoms. This knowledge can be used to select a treatment or medication.
The procedure for HnMM matching is the following:

1. For each possible model or model configuration, the total probability of generating the given sequence is calculated as described in the previous section.

2. The models are ranked according to these unconditional probabilities.

3. The model with the largest probability of generating the given sequence represents the most likely system to have produced the output.

We applied this model matching approach in different experiments, described in the following section.
Knowledge of likely system behavior can be used to determine promising courses of action to take. The model that has been determined to be the most likely one for the system can be used to simulate the future system development. Knowledge about a likely system setup can also be compared to the manufacturer's specifications. When large deviations are detected, further steps can be taken, such as scheduling maintenance or supporting claims towards the manufacturer.

EXPERIMENTS WITH PATIENT DIAGNOSIS EXAMPLE

This section describes the validation and performance experiments performed for the Proxel-based HnMM matching algorithm and patient diagnosis application example. For test purposes we created models of the different illnesses and their progression in AnyLogic (Borshchev 2007). We then generated several different sequences of symptoms for each of the different models.
Due to inherent ambiguities within and similarities among the models, the illness model that was used to create a specific symptom sequence is not always the most likely one. The experiments are aimed at answering the following questions: How does less / more information (less, more frequent symptom data) influence the matching accuracy and algorithm performance? More specifically, how does matching accuracy develop when increasing the similarity of the models? If the matching errors are due to inherent ambiguities or algorithmic problems is currently very hard to determine, since it is still unclear how to determine inherent model ambiguities. The answers to these questions will lead to a statement of the overall feasibility of the proposed method for model matching in this particular application setting.

We will be using the following quantities as measures of our algorithms matching accuracy (Bramer 2008): accuracy , precision, recall and F1-measure. Symbol explanation: P positives (traces caused by the illness), N negatives (traces not caused by the illness), correctly classified traces: TP true positives and TN true negatives, incorrectly classified traces FP false positives and FN false negatives.

$$\begin{aligned} accuracy &= \frac{TP+TN}{P+N} \\ precision &= \frac{TP}{TP+FP} \\ recall &= \frac{TP}{P} \\ F1-measure &= \frac{2*precision*recall}{precision+recall} \end{aligned}$$

The algorithms performance will not be evaluated in a separate experiment. In all of the test cases, the algorithm runtime was within a few seconds, even with the smallest evaluation time step chosen ($\Delta t = 0.1 day$). Smaller evaluation time steps were not necessary, because the choice of evaluation time step had little effect on the matching accuracy. Furthermore, the traces are of limited length, since the symptoms are recorded only for the relatively short period of time, during which the patient is ill, resulting on traces of at most 100 symbols.

Measurement Frequency Experiment

The goal of this experiment was to determine the effect of the symbol frequency on the matching accuracy. The symbol frequency in the application example was determined by the measurement interval of the various symptoms. We used the illness example with three distinct abstract illnesses as our test set. We created two separate test sets, each containing five traces for each illness. These 15 traces were then tested against each of the three illnesses. We varied the interval between subsequent measurements (blood test, urine test, taking temperature) from 0.25 days (0.5 days in test set two) to 2.0 days. The illness with the highest absolute probability for generating the given trace was then assumed to be the one to have caused that trace.
With increasing measurement intervals, the number of correctly classified traces decreases, and the class assignment of a trace seems to become arbitrary. Table 3 shows precision, recall and F1-measure for the different measurement intervals. It shows that the matching accuracy decreases with increasing measurement intervals. Figure 4 shows the fraction of correctly classified traces when the measurement frequency decreases. For a measurement interval of 0.25, almost all traces are classified correctly. Doubling the time between two measurements decreases the matching accuracy considerably. However,

Table 3: Accuracy Measures for Different Measurement Intervals for First Test Set

Interval	Precision	Recall	F1-Measure
0.25	1.0	1.0	1.0
0.5	0.84	0.8	0.818
1.0	0.61	0.6	0.606
2.0	0.59	0.6	0.594

the size of this decrease depends on the test set. Matching accuracy of the first test set declines more steeply than that of the second one.

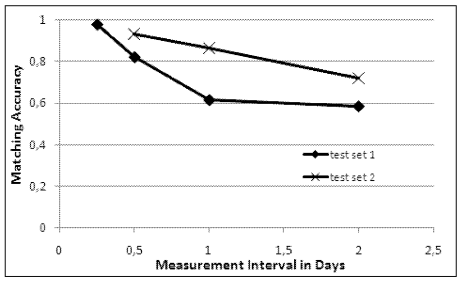


Figure 4: Matching Accuracy with Increasing Measurement Interval for Different Test Sets

The decrease in matching accuracy with decreasing symbol frequency demonstrated in this section is to be expected. It is due to a decrease in the amount of information available on the system behavior, thus also decreasing the ability to differentiate between traces created by different illness models. This decline is not due to shortcomings in the algorithm itself, but only to a decreasing amount of information available on the patient.

Symptom Reduction Experiment

The goal of this experiment was to determine the effect of the number of different symptoms on the illness matching accuracy. We used the illness example with three distinct abstract illnesses as our test set. We created two separate test sets, each containing five traces for each illness. These 15 traces were tested against each of the three illnesses. We varied the number of different measurements available by deleting one or two of the symptoms from the traces, fixing the measurement interval at 0.25 days. The illness with the highest absolute probability for generating the given trace was assumed to be the one to have caused the symptoms.
Figure 5 shows the development of the fraction of correctly classified traces when the number of available symptoms decreases. When all three symptoms are available in a trace, almost all traces are classified correctly. When the number of symptoms available is reduced to two, the matching accuracy decreases slightly. When only one of the three symptoms is available, the accuracy declines considerably, while not each of the three different symptoms leads to the same steep decline. Both test set showed almost the same behavior. The symptom fever distinguishes better between the illnesses, probably because it has only two possible outcomes, but the exact reason for this different ability to distinguish is a subject of further research.

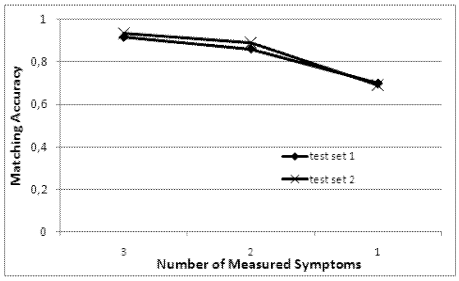


Figure 5: Matching Accuracy with Decreasing Number of Symptoms being Recorded for Different Test Sets

Table 4: Symptom Probabilities for Midpoint Model

		Stage 1	Stage 2
midpoint	cough	0.65	0.3
	fever	0.5	0.05
	pain	0.4	0.1

The reduction of matching accuracy observed in this experiment is again as expected. Decreasing the amount of information available also decreases the ability to differentiate between traces created by different models.

Ambiguity Experiment

The goal of this experiment was to determine the effect of model ambiguity on the matching accuracy. We used the models of the common cold and the influenza as baseline, because the algorithm was able to match these traces even with large measurement intervals with an accuracy of 0.9. We created an intermediate model by changing the stage durations and symptom probabilities of the common cold model towards those of the influenza. For each of the different levels of similarity, ten traces were created for the influenza model, the midpoint model and the common cold model.
The stage durations and symptom probabilities for the common cold (*Cold*) and the influenza infection (*Flu*) are given in Section . The stage durations for an intermediate point (*midpoint*) are $Stage1 \sim N(4,1)$ and $Stage2 \sim N(5,2)$. Table 4 shows the symptom probabilities in each stage for an intermediate point (*midpoint*).
Figure 6 shows the effect of increasing model similarity on the matching accuracy for the given example. The x-axis reflects the increasingly similar stage durations, where *Flu-Cold* having the largest difference and *Flu-Flu* the same stage durations for both models. The different colored bars similarly reflect the increasingly similar symptom probabilites. The small increase in matching accuracy when changing the symptoms to the mid-

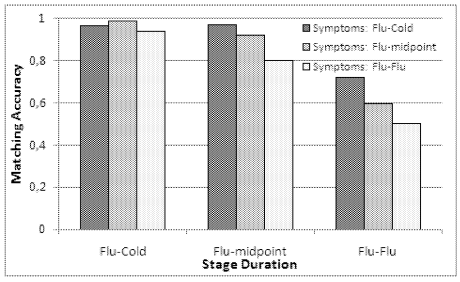


Figure 6: Accuracy for Increasing Model Similarity

point and keeping the original stage durations is probably due to random effects in the two different test sets. The diagram shows that with increasing similarity it becomes increasingly difficult to determine the correct illness. The matching accuracy seems to be more sensitive to a change in stage durations than to a change in symptom probabilities. However, that might be due to the test sets considered. The decline in matching accuracy in this experiment is due to increasing ambiguity between the different models.

CONCLUSION AND OUTLOOK

The paper presents a new application of HnMM from medicine. A patient's internal state of illness is unknown and regarded as source of the observed symptoms. The analysis is done by on-the-fly state space-based simulation only regarding the paths that could have resulted in the observed trace. The algorithm is successfully applied to finding a specific model's probability for a given symptom trace and finding the most likely model to have produced the given symptom trace.

The algorithm's matching accuracy is tested in the experiments section. The proposed method is able to accurately determine the correct illness for a given sequence of symptoms. The algorithm exhibits an expected decrease in matching accuracy, when the amount of information available is reduced or the model ambiguities are increased. The computational effort for the matching procedure is neglectable in this application example, due to the limited size of the traces and models.

Model matching is tested for the first time for HnMM, broadening the range of analysis tasks that can be solved for HnMM. The application example requires symbols being emitted depending on the current state at arbitrary points in time, not as before at state changes, thus further broadening the range of application areas.

Future Work

The application example presented in this paper is a diagnosis based on symptoms. We modeled an illness from an engineer's point of view. We do not know whether discrete stochastic models as we used them can represent illness progression in general or in specific cases. We still have to discuss the proposed approach with practitioners and build models based on field data or incorporate other important aspects that we have so far disregarded. Furthermore, the approach will most likely work for applications from engineering and manufacturing, because these systems can be represented accurately by discrete stochastic models.

We are also attempting to sompare the proposed approach with established data mining or classification techniques. However, it is not yet clear, which methods are applicable to the systems specified in this paper. We are interested in incorporating interventions from outside the model, which affect the future development of the system. This can be interesting in the medical domain, when incorporating the effect of treatments and medications. In general, we need to test the applicability of the approach to more complex systems, testing boundaries of applicability and extend the model matching to different application domains.

References

Bolch G.; Greiner S.; de Meer H.; and Trivedi K.S., 1998. *Queuing Networks and Markov Chains.* John Wiley & Sons, New York.

Borshchev A., 2007. *Multi-Method Simulation Modeling using AnyLogic.* INFORMS Roundtable Fall Meeting, Seattle. Www.anylogic.com XJ Technologies.

Bramer M., 2008. *Principles of Data Mining.* Springer-Verlag London ltd.

Buchholz R.; Krull C.; Strigl T.; and Horton G., 2010. *Using Hidden non-Markovian Models to Reconstruct System Behavior in Partially-Observable Systems.* In *3rd International ICST Conference on Simulation Tools and Techniques.*

Fink G.A., 2008. *Markov Models for Pattern Recognition.* Springer, Berlin, Heidelberg.

Horton G., 2002. *A New Paradigm for the Numerical Simulation of Stochastic Petri Nets with General Firing Times.* In *Proceedings of the European Simulation Symposium 2002.* SCS European Publishing House, 129–136.

Krull C. and Horton G., 2009. *HIDDEN NON-MARKOVIAN MODELS: FORMALIZATION AND SOLUTION APPROACHES.* In *Proceedings of 6th Vienna Conference on Mathematical Modelling, Vienna, Austria.* 682–693.

Lazarova-Molnar S., 2005. *The Proxel-Based Method: Formalisation, Analysis and Applications.* Ph.D. thesis, Otto-von-Guericke-University Magdeburg.

ARTIFICIAL NEURON WITH HOMEOSTATIC BEHAVIOUR

Martin Ruzek, Tomas Brandejsky
Czech Technical University in Prague
Konviktska 20
Praha 1, 100 00, Czech republic
Email: ruzekmar@fd.cvut.cz, brandejsky@fd.cvut.cz

KEYWORDS
model design, artificial neuron, homeostasis, network, model reduction

ABSTRACT

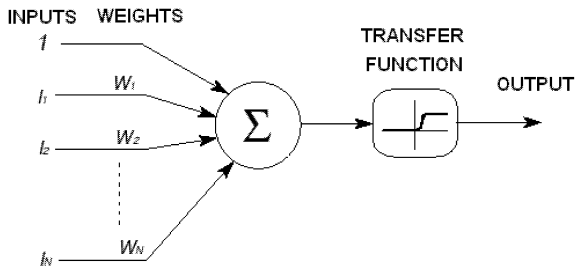
Homeostasis is a property of a system that regulates its internal environment in order to maintain stable conditions. It is typical for any for biological systems and therefore also for neural cell. This paper presents a way how to use the idea of homeostasis in the field of artificial neural networks. The artificial neuron is here considered as an information homeostat. The state of equilibrium means a situation when the level of computational utility reaches its maximum. This idea is based on the presumption that the neuron has two inputs: first, the output of the neurons in the previous layer through its dendrites, and secondly the part of its output signal that is returned from the folowing layer through its axon.
The presented idea is inspired by the fact that the biological neuron can know which part of its output energy is accepted by other neurons. Several methods of the learning are presented. Some qualities of the homeostatic neuron, such as stability, speed of learning and independence, are discussed.

INTRODUCTION

The artificial neural networks were inspired by the biological neuron. The biological neuron is, however, too complex to be accurately simulated. Therefore, several approximations are used in practical applications. The most commonly known is the MsCulloch-Pitts artificial neuron. The main advantage of this model is its simplicity. On the other hand, this simplification significantly reduces its functionality. Many other models have been used, some of them were closer to biological neuron. Also, various researches of homeostasis in neurons have been carried out. In (Batolozzi and Indiveri 2009) is presented a research of the homeostasis in the synapsis, especially focused on silicon implementations of neural network. Another remarkable facts, regarding mostly the biological neural systems, can be found in (Turrigiano and Nelson 2004). The authors suggest that the homeostasis can be either 'global' or 'local'. In the case of global homeostasis, all the neurons will cooperate and behave more or less similarly. This idea seems quite promising to solve out some problems, such as stability of the whole system, however, this paper focuses only on homeostasis in a single neuron. The global homeostasis will be the topic of future research. Another study of homeostasis in neural cells is described in (Downing and Miyan 2000), where relation of homeostasis to imnity system is presented.
This work is aimed to find such a compromise between the biological and artificial neuron that is simple enaugh to be simulated and that conserves the basic functions. The exact model is not achievable, so that the main intention was to design a neuron that is able of self–learning and with homeostatic behaviour.

METHODOLOGY

The proposed neuron is based on McCulloch-Pitts model that is illustrated on figure 1.



Fi

Figure 1: McCulloch-Pitts model of neuron

Formally, its function is described as

$$y = f(\sum_{i=1}^{n} x_i w_i + \delta) \quad (1)$$

The difference from the McCulloch-Pitts model is the fact that the the output (axon) serves also as an input for the learning stage. The learning process is divided into two steps: first, in the forward phase, the neuron computes its output according to (1). In the second step, the learning phase, the neuron updates its parameters (the input weights) according to the level of acceptance of its output from the previous step. The *level of acceptance* or the *utility* is the part of the output energy that is accepted by other neurons. This proccess reminds the back-propagation algorithm, however, the learning phase is different. In the proposed model, there is no desired function for the neuron. The neuron is only trying to find such vector of its input weights for which the level of acceptance is maximal. The neuron is trying to be as useful for other neurons as possible and to do so, it is trying to maximize the level of acceptance.

Homeostasis

Homeostasis, a typical property of biological systems, has been widely used in numerious technical applications,

such as control systems. Each homeostat has three basic units: receptor, control system and effector. In our model, the receptor is the axon, the control system is the neuron body and the effector are the dendrites. Therefore, the process of the learning has the opposite direction than the computational phase. Generally, the homeostat can use two types of feedback: the positive feedback and the negative feedback.
The *positive feedback* means that the system is trying to reach some optimal position instead of maintaining stable conditions.
The *negative feedback* is used to maintain stable environment and means that the output is subtracted from the input..
The positive feedback is used to reach some state, the negative to maintain the state. In the case of the homeostatic neuron. the task is more to teach the neuron than to maintain its function. Therefore, the positive feedback is used. Mathematicaly, the possitive feedback is described by:

$$Y = X \cdot A + C \cdot X \cdot Y \qquad (2)$$

where X is the input, Y is the output, A is the system matrix and C is the control matrix. However, it is not possible to use this principle directly. The neuron is seeking a state for which the utility for other neurons is maximal. Therefore, it can not observe simply the utility for given input vector, but it must consider the changes of the utility. Therefore, the neuron must be equiped by 'memory' that allows it to compare its current utility with the previous one.

Learning process

The basic idea is that the neuron is able to change its weights and by this change to increase its importance in the whole network. In further text, *input weight* w_i will denote the weight that is leading the signal into the reference neuron and *output weight* w_o will denote the weight that transfers the output of the reference neuron to others neurons. *Utility* is number from <0;1> interval that quantifies the importance of the reference neuron to other neurons.
The basic idea is that the neuron is willing to be useful for other neurons. If the other neurons are interested in its output, they will accept major part of its output energy. The neuron is programmed so that it tries to maximize the part of its output energy that is accepted by other neurons.
The simplest way how to calculate the utility is by the output weights. The weights, both output and input, are limited to interval <-1;1>. It is possible to use another limits, this interval was chosen because of simple analysis. With the respect to the negative values, it isn't possible to determine the importance of the connection directly by the sum of the output weights. For instance, a connection with weight –0.8 is more important than 0.3. Therefore, the absolute or square values of the weights are used instead.
The process of the learning of the neuron is described by the following algorithm:

1. neuron sets its input weights to random values
2. neuron performs the forward phase (the computation of the output values)
3. neuron computes the utility from the output weights and saves it to the memory
4. neuron adds *dw* to the weight of the first connection
5. neuron repeats the forward phase; computes the output of the same input with the changed weight
6. if the previous change made the utility greater, the neuron will confirm the change. In the contrary, it will substrack *dw* from the first connection
7. neuron repeats steps 3 to 6 with all the connections and all the inputs

To calculate the importance of the neuron, several methods can be used. The first extreme case is searching such setting of the input weights that maximizes the sum of absolute values of the output weights; so that the neuron is trying to be interesting to all of the neurons in the higher layer. Then, the utility is calculated as:

$$q = \sum_{j=1}^{n} \left| w_j^o \right| \qquad (3)$$

The other extreme case is neuron that is trying to be interesting for only one neuron in the higher layer. This neuron is searching a setting for which the maximum of the output weights is maximal.

$$q = \max \left| w_j^o \right|; \; j \in \{0,1,..n\} \qquad (4)$$

Between these extremes, there are many compromise variants. For example, the neuron may try to increase its importance to some given number of output neurons..

RESULTS

The neuron was programmed in MATLAB R2008b. The input is a vector of any length that is composed by ones and zeros. The output is a real number from <0;1> interval. This number determines the values of the output weights, which are the 'second input' of the neuron. Several functions may be used for the computation of the utility, for example sum of absolute values, of square values, maximum or other.
In this model it was crucial to define the *output layer*. This layer is necessary because neuron can be tested and improved only as a part of the whole environment. The model of the output layer simulates the connections between the reference neuron and the higher layer. The realization of this layer was the major difficulty of the whole model. The neuron's functions are well defined and therefore its code can be easily written, but the output layer is not preciously defined. In the output layer, there may be many different neurons with diverse functions.
During the first phase of the tests, simple and homogenous layers were considered. Neurons of the output layer were more or less identical. The great majority of the neurons was interested in simple transmission of one input of the reference neuron. In this case, the reference neuron tends to set one weight (the

desired connection) to 1 and all the others to 0. The speed of convergence for different number of dendrites shows Table 1.

Number of dendrites	3	5	7	9
Number of iterations	137	698	4891	3714

Table 1: Number of necessary iteration as function of number of the inputs to the neuron. The neuron was trained to a simple function of transmitting one input

In the next step, more complex functions were desired by the output layer. First, the output neurons were divided into two groups, each one was interested in another input signal In this case, the convergence process was significantly slower. Aftewards, the neuron was tested in an environment with diverse and complex desired functions. In this case, the convergence was very slow and sometimes the neuron didn't reach the homeostatic position at all.

Discussion

Compared to McCulloch-Pitts model from which the homeostatical neuron is derived, it has slower speed of learning. However, the basic disadvantage of this type of learning is the dynamics of learning. The proposed neuron changes its weights and expects that the change will be immediately reflected by the output layer. This presumption will be true only for networks with two layers. In the case of multi-layered network it will last several steps until the change appears again in the neuron. One of the possible solutions is setting the dynamics of the inputs to sufficiently low level, so that the change of the input signal will be significantly slower than the communication between the neurons.

Network of homeostatic neurons

The final neural network shall be composed only of neurons of this type. In other words, the neuron proposed in this paper should be also a part of the higher layer. The optimization of the neural network that is based on homeostatic neurons is quite complicated task, as there are too many unknowns (except the 'traditional' questions of the neural network such as topology and neurons' parameters, here also the learning method, e.g. sum of output weights, maximum, absolute vs. square values …).

CONCLUSIONS

The neuron proposed in this article is able to set its input weights in a waz that maximiyes its importance to other neurons. This process reminds the searching of homeostatical position in neural cells. The main advantage of this neuron is its ability of self-improvement in a way that can be expected in the biological neuron. The learning is indirect; there is no channel for the back propagation of the error. The neuron is improving itself in order to increase its importance to other neurons.

In comparison to the back-propagation algorithm, the proposed neuron has a slower learning, however, this disadvantage is compensatedb by another characteristics, such as the independence of the learning. During the process of learning, it is searching its homeostatic position. This learning process can be adapted also for other parameters of the neuron, such as the slope and the threshold; however, in this article only the weight adjustment is discussed. It is closer to the original biological inspiration, the neural cell because no explicit external function is needed. The experiments confirmed that this neuron converges to its homeostatic position, however in some more complicated environment it doesn't work. The main disadvantage of this model is that is applicable only to systems with first order delay. The future planned network composed of independent homeostatic neurons may be used for simulation of brain functions as suggested in (Rumelhart and McClelland 1986, Zarita and Ong 2007), as the principle meets the concept of homeostat that is proper to all living cells. Another possible improvement consists in considering the fuzzy characteristics of the neurons, as proposed in (Novoa et al., 2000). In the future, the question of signal delay should be resolved.

Acknowledgement: This work has been carried out with the help of CTU research and student grant number SGS10/217/OHK2/2T/16.

REFERENCES

Bartolozzi Chiara, Indiveri Giacomo, *Global scaling of synaptic efficacy: Homeostasis in silicon synapses,* Neurocomputing 72 (2009), pp. 726-731.

Downing J., Miyan J., *Neural immunoregulation:emerging roles for nerves in immune homeostasis and desease,* Immunology today, Vol. 21, No. 6, June 2000

Novoa, D. Pérez, A. Rivas, F., "Fault Detection scheme using Neo-fuzzy Neurons", *IASTED International Conference on Intelligent Systems and Control*, Honolulu, Hawaii, USA, 2000

Rumelhart D. E.,McClelland J., *Parallel Distributed Processing*, MIT Press, USA, 1986

Turrigiano G., Nelson S., *Homeostatcal plasticity in the developing nervous system,* Neuroscience, Nature Publishing group, Vol 5, February 2004

Zarita Zainuddin & Ong P., "Function Approximation Using Artificial Neural Networks", *INTERNATIONAL JOURNAL OF SYSTEMS APPLICATIONS, ENGINEERING & DEVELOPMENT,* Issue 4, Volume 1, 2007, pp. 173-178

PERFORMANCE ANALYSIS OF PARALLEL DEMOGRAPHIC SIMULATION

Bhakti S. S. Onggo
Department of Management Science
Lancaster University Management School
Lancaster LA1 4YX, UK

Cristina Montañola-Sales
Department d'Estadística i Investigació Operativa (DEIO)
Universitat Politècnica de Catalunya (UPC)
Computer Applications in Science & Engineering
Barcelona Supercomputing Center (BSC)
Barcelona, 08034, SPAIN

Josep Casanovas-Garcia
Department d'Estadística i Investigació Operativa (DEIO)
Universitat Politècnica de Catalunya (UPC)
Barcelona, 08034, SPAIN

KEYWORDS
Parallel simulation, demographic simulation, simulation tool, performance analysis

ABSTRACT

Today, we have seen an increase in the number of papers on parallel simulation applications outside the traditional military and network simulations areas, such as in the physical science and management science. One of the new areas in which parallel simulation could be used is demography, specifically for population projection. In this paper, we report the performance evaluation results of a parallel demographic simulation tool called Yades. We investigate the effect of three factors: unbalanced workload, heterogeneous processing speed and heterogeneous communication latency on performance measures such as: time spent in executing useful events, time spent for overhead and the number of rollbacks. The results are consistent with what has been reported in other application areas of parallel simulation. Since the application in demography is new, it is useful to quantify the effect of the three factors on performance.

INTRODUCTION

Parallel simulation deals with techniques that allow the use of multiple processors to run a single simulation. One of the techniques to achieve parallelization is through partitioning a simulation model into a set of smaller components called logical processes (LPs) and running the LPs concurrently. Research in parallel simulation has produced a number of synchronization protocols which can be classified into two main categories: conservative and optimistic. This classification is based on how the local causality constraint (lcc) is maintained. Lcc imposes that if event *a* happens before event *b* and both events happen in the same LP, then event *a* must be executed before event *b*. Parallel simulation must adhere to lcc to produce correct simulation results. This will guarantee that it will produce a result that matches the equivalent sequential simulation. Conservative protocols do not allow any lcc violation throughout the duration of the simulation. Optimistic protocols allow lcc violation, but provide mechanisms to rectify it. (Perumalla 2006) provides a good summary on the recent development in parallel simulation.

Traditionally, parallel simulation has been applied in military and network simulations. Today, we have seen an increase in the number of papers reporting on parallel simulation applications outside the traditional areas. (Tang et al. 2005) conducted an initial study in applying parallel simulation to a plasma physics application. In the realm of biological science, (Lobb et al. 2005) applied parallel simulation to a neuron model. (Lan and Pidd 2005) applied parallel simulation to simulate a quasi-continuous manufacturing process. (Yoginath and Perumalla 2008) applied parallel simulation to a traffic simulation model. (Bauer et al. 2009) conducted an experiment to evaluate the scalability of their parallel simulation tool using a Transmission Line Matrix model (for electromagnetic wave propagation). (Park and Fujimoto 2009) evaluated their Master/Worker parallel simulation tool using a particle physic model.

(Onggo 2008) developed a parallel simulation tool to run demographic simulation models. In the past few years, the use of simulation in demography at the micro level (such as household and individual) has become more common. A recent example includes (Zinn et al. 2009). The main advantage of this approach is that individual-specific explanatory variables can be included in the model. For example, we may include factors such as age, education level, salary group and ethnicity to model the number of children that an individual female will have. Demographic modelers often use a complex regression model to decide the change in the state of an individual whenever a specific life event occurs. There is also a general interest in using larger sample sizes in demographic simulation models. Therefore, parallel simulation could provide an alternative to speed up the execution of such compute-intensive demographic models. (Onggo 2010) demonstrated that parallel simulation could improve the execution time of a large-scale demographic model significantly. However, the experiment was conducted using an ideal model where the population is distributed equally across different processors. The objective of this paper is to measure the effect of unbalanced workload, heterogeneous processor speed, and heterogeneous communication latency on the performance of the parallel demographic simulation tool. Since the parallel simulation application on demographic models is new, it is important to understand the performance under some possible runtime configurations. The rest of this paper is organized as follows. Section 2 presents an overview of existing demographic simulation tools. Section 3

summarizes the design of our parallel demographic simulation tool. We explain the experiments and the performance analysis results in section 4. Section 5 presents our concluding remarks and highlight further works.

RELATED WORK

Demography is the study of human population in relation to changes brought about by the interplay of births, deaths and migration (Pressat 1985). One of the main applications of demography is population projection. The use of simulation for population projection has increased in recent years. As stated previously, this is because we may include individual-specific explanatory variables in the model. Apart from their application in population projection, demographic models are often used as the basis for policy modelling and analysis. To take two cases, (Walker et al. 2000) used simulation to analyze the effects of demographic changes on government expenditure on pharmaceutical benefits in Australia. Similarly, (Bonnet and Mahieu 2000) took into account three main components: demographic, labour market and income in the simulation model to analyze pension policy in France.

One of the commonly used simulation modeling paradigms in demography is *microsimulation*. The initial work in microsimulation goes back to the work of (Orcutt 1957). In this simulation paradigm, modellers have to specify a random sampling process for each individual at each simulation time point, to determine the state of each individual at the next simulation time point. At one extreme, the sampling process requires a simple random sampling. At another extreme, it may require a complex regression model. Many microsimulation tools have been built for certain public policies, for example LABORsim for policies related to labour supply in Italy (Leombruni and Richiardi 2006). SOCSIM (Hammel et al. 1990) is one among the few generic microsimulation tools for demography. Recently, (Zinn et al. 2009) developed another generic microsimulation tool called MIC-CORE.

System dynamics is another commonly used modeling paradigm in developing demographic simulation models. Unlike microsimulation, system dynamics does not keep track changes in the state of each individual but focuses more on the population of individuals and the rates of individuals moving from one state to another. System dynamics is commonly used to analyze the complex feedback systems and the mutual interactions in the system over time. Important works in this area include the World Dynamics (Jay Forrester 1971) and World3 population model (Meadows et al. 1972, 2004). A demographic model in system dynamics is often used as a component in a policy model. For example, (Ahmad and Billimek 2005) developed a system dynamics model to analyze policies to reduce the harmful effect of tobacco on population health. Key demographic components such as: fertility rate, mortality rate and net migration are included in the model. (Saysel et al. 2002) developed a system dynamics model to analyze policies on various environmental issues such as water distribution management and agricultural pollution. The model took into account the mutual interactions between the environmental issues and the demography in the region.

Similar to microsimulation, *discrete-event simulation* keeps track the individuals 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 each simulation time point. It inspects an individual only when the state of the individual changes. Most discrete-event demographic simulation models are used in applications such as healthcare and epidemiology. For example, (Rauner et al. 2005) developed a discrete-event simulation model to study the effectiveness of intervention programs to reduce the vertical HIV transmission. The model used demographic data to initialize and to project the population. The model took into account the demographic information for activities such as: being tested for HIV and receiving treatment. (Roderick et al. 2004) developed a discrete-event simulation model to estimate the future demand of renal replacement therapy in England which took into account the demographic population changes in England. A number of researchers have attempted to build large-scale epidemiological simulation models. The main objective is to understand the spread of global epidemics which may include analysis of a large number of individuals. (Eubank 2002) and (Rao and Chernyakhovsky 2008) showed that parallel discrete-event simulation was needed for large-scale epidemiological models. They developed specialized parallel simulation tools for epidemiological models. (Onggo 2008, 2010) developed a parallel demographic simulation tool which focused more on the life events that change the economic and social status of a large number of individuals.

THE PARALLEL DEMOGRAPHIC SIMULATION TOOL

Yades (*Yet Another DEmographic Simulator*) was implemented using μsik parallel simulation library (Onggo 2008). (Perumalla 2005) developed μsik parallel simulation library that supported multiple synchronization algorithms such as: lookahead-based conservative protocol and rollback-based optimistic protocol (time warp with state-saving and reverse-computation). 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 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). To implement a simulation model in μsik parallel simulation library, we must specify three main components: a physical process (must inherit from class Simulator) which is responsible for managing LPs, a set of logical processes (each must inherit from one of these classes: NormalSimProcess, PeriodicSimProcess, or ThreadedSimProcess) which contains the main simulation processes, and a set of events (each must inherit from class SimEvent) which is used by an LP to communicate with another LP. A detailed explanation on the structure of a simulation model written in μsik can be found in (Perumalla 2005). The library comes with a number of examples (PHOLD and space craft) which are very useful. At the time of writing, the μsik website is no longer available hence those who are interested in the library may contact the developer (Perumalla 2005).

Yades allows users to provide data for the initial population. The data follows the structure of the UK Family Resources Survey (FRS) data which includes the proportion of different age groups in the population, the proportion of different

types of families by age group, proportion of different economic status by age group, proportion of different marital status by age group and the proportion of the number of children in a family. 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.

There are two types of LPs in Yades: *family unit* and *administrative area*. In FRS data, a family unit is defined as a single independent individual or two independent individuals living together (as married, in civil-partnership, or in cohabitation) and any dependent individuals (children). Hence, in this definition, a family unit may represent an independent individual, a single parent, a childless couple or a nuclear family. For completeness, in Yades the definition is extended to include orphans, that is, a family unit of dependent children without any parents.

The main advantage of representing a family unit as an LP is that many public policies may apply to individuals as well as groups of related individuals, such as households and single parents. For example, the UK Department for Work and Pensions and HM Revenue & Customs manage a number of public funds that may apply to individuals (including jobseeker's allowance and incapacity benefit) or groups of related individuals (which could include child benefit and housing benefit). Therefore, it is easier for users to specify policies for different types of family unit. The decision to represent a family unit as an LP has another advantage. When there is a change in the marital status that affects couples (such as from married to divorced or from married to widowed), only one message needs to be sent to the affected couple. In the earlier work where an LP represents an individual (Onggo 2008), two messages had to be sent, one for each affected individual. Hence, representing a family unit as an LP reduces the number of sent messages in the simulation.

A family unit may receive events which are related to five demographic components that may change the system states. Modellers need to specify models for five demographic components: fertility, a change in economic status, a change in marital status, migration and mortality. The fertility component determines whether a female individual will give birth, based on the characteristic of the female individual and the current calendar time. The model returns the time when the baby is due. Similarly, modelers can use the characteristic of an individual and the current calendar time to determine a new economic status of that individual. A new marital status can be modeled based on the characteristics of the individual (or individuals for a couple) and the current calendar time. If the new status is either married or cohabitating, modellers need to define the criteria that will be used to match the individual to another individual from the list of prospective partners (i.e. we use a closed marriage model). If a suitable partner is found, then a 'family formation' event will be scheduled for both individuals. Otherwise, the individual will be added to the list for a fixed duration. If a partner still cannot be found at the end of the duration, an event will be sent to remove the individual from the list. Modellers also need to specify a model that is used to determine whether a family unit is going to migrate. If the destination is in another country (emigration), the family unit will simply be removed from the simulation. Finally, in the mortality component, modellers need to model the time when an individual will die based on the characteristics of the individual. Commonly used methods, such as life table and survival function can be used for the mortality component.

The second type of LP represents an administrative area where a number of families live. This LP will handle domestic migrations, immigration, changes in simulation parameters and periodic reports. Yades allows users to have administrative areas with different population characteristics. The main limitation of the current version is that it only allows one processing element to run one administrative area.

An administrative area may receive four types of events. The first event is used when a family unit is going to migrate to a new administrative area. The family unit will send an event to request a place at the destination area. The destination area will prepare an empty family unit and send the identification number to the migrating family. Subsequently, all members of the family will be sent to the new location. The second type of event is used to simulate the immigration events, i.e., the number of family units entering the country every month (in batches). This allows modellers to implement different models for immigration policies, such as the number and demographic characteristics of the immigrants. The third type of event can be used by modellers to specify periodical changes in simulation parameters such as life table and fertility rates. Finally, the report event can be used to produce periodical reports, for example, a report on the population structure (by gender, age group, marital status and economic status).

We use the competing risk model (Hosmer et al. 2008, Chapter 9) to ensure that every family unit will have exactly one future event. This approach will sample time-to-event for a number of competing events such as death, giving birth, change in marital status and change in economic status. The event with the shortest time-to-event will be chosen and executed. This process is repeated whenever a life event occurs (except for death and emigration). As a consequence of this approach, the model uses a continuous time where future events can happen almost immediately. Hence, the lookahead is relatively small that makes a conservative protocol less efficient. For this reason, the optimistic protocol is used.

PERFORMANCE ANALYSIS

In this section, we present the results of experiments to understand Yades performance under varying conditions: homogeneous environment, heterogeneous population size, heterogeneous processing speed and heterogeneous communication latencies. All experiments were run using μsik settings that gave a roll-back based optimistic parallel simulation execution with a state-saving mechanism and a time window of 12 months (to limit how far an LP can advance ahead of others). The program was compiled using gcc version 3.3.5 with the optimization flag O3 turned on and mpich version 1.2.5 was used. All experiments were run on a cluster of PCs connected via a dedicated gigabit Ethernet switch. Each node has two dual-core 2.4GHz Opteron CPUs and 8GB of memory. All results presented in this section are based on the average of five replications.

Because the standard deviations are very low, we did not need more than five replications for each experiment. The results of the parallel simulation have been checked against the sequential execution for correctness.

Homogeneous environment

The objective of this experiment is to understand the effect of migration activities on the performance of the tool, specifically the execution time and the number of rollbacks, under an ideal execution configuration. In this configuration, we ran the simulation for a period of 50 years with an initial population size of 640,000 family units (around 1.3 million individuals), divided equally among all administrative areas. This would produce a homogeneous workload to all processing elements. The simulation was run on one compute node containing four processing elements to minimize the effect of heterogeneous communication latency. The probability of migrations was varied between 0% and 60%. The probability of migrations determines the probability of a family unit to migrate when there is a change in the employment status of one of the parents. As explained earlier, migrations are responsible for all inter-processor communications in the simulation.

Table 1: Number of Migrations

Probability	0%	20%	40%	60%
Number of migrations (individuals)	0	172,231	344,094	516,462

The results are shown in Table 1 and Figure 1. As expected, the number of migrations is proportional to the migration probability (Table 1). Figure 1 shows that the increase in the number of migrations increases the execution time. The increase in the number of migrations increases the number of event that has to be executed by the simulator. As a result, it requires more time to execute all useful events. In this configuration (homogeneous environment), the average number of rollbacks is close to zero regardless of the migration probability. This indicates that the overhead costs are mainly due to the inter-processor communications.

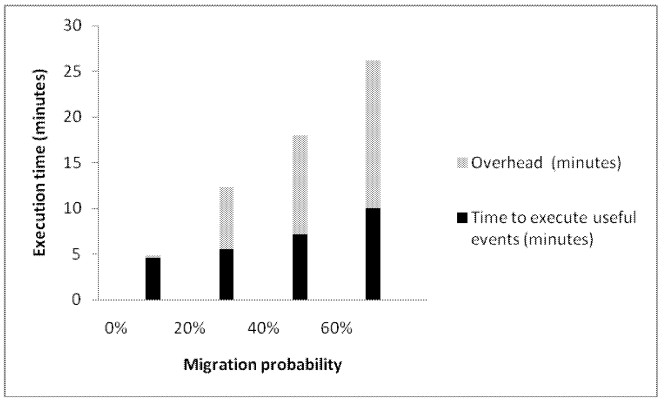


Figure 1: Execution Time for Homogeneous Workload

Unbalanced population size

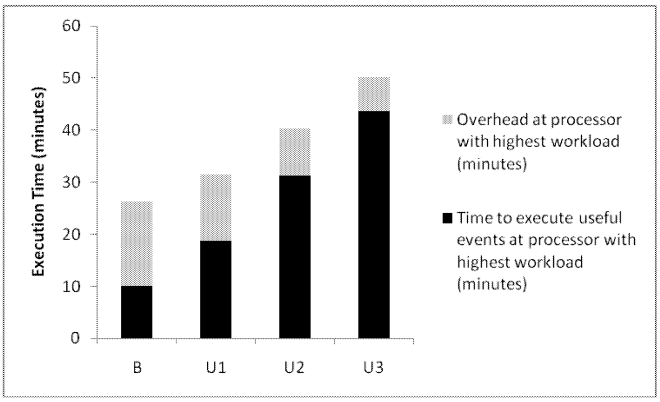
In practice, the number of family units may vary across administrative areas. Hence, it is important to measure the effect of an unbalanced distribution of family units on the performance of the tool. As in section 4.1, in the experiment, we set the simulation duration to 50 years but fixed the probability of migrations to 60%. We ran the simulation on four processing elements in one compute node with a total of 640,000 family units at the start of the simulation. We varied the distributions of the family units from a balanced distribution of 160,000 family units on each processing element to a rather unbalanced distribution of 385,000 family units on one processing element and 85,000 family units on each of the remaining processing elements.

The four configurations are arranged in different columns in Table 2 (B=160,000 on each processor, U1=235,000 on one processor and 135 on remaining processors, U2=310,000 on one processor and 110,000 on remaining processors, and U3=385,000 on one processor and 85,000 on remaining processors). Row 2 shows total number of migrations. As expected, the total number of migrations is roughly the same regardless of the distribution of the family units. Row 3 onwards shows the total number of rollbacks.

Table 2: Effect of Unbalanced Distribution of Family Units on Performance

Workload distribution	B	U1	U2	U3
Number of migrations (individuals)	516,462	512,160	517,034	513,971
Total rollbacks	0	496,166	564,267	698,158

Figure 2 shows that the equal distribution of family units across processing elements results in the best execution time. The worst execution time (almost two times slower) was given by the most unbalanced configuration in the experiments (385-85-85-85). This result is intuitive because an equal distribution of family units will result in an equally distributed workload across the processing elements. Consequently, the processing elements can advance their simulation clock at a similar pace, which reduces the number of rollbacks.



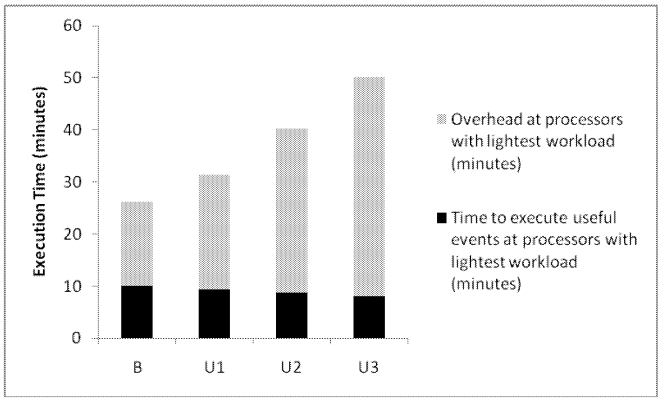


Figure 2: Effect of Unbalanced Distribution of Family Units on Performance

A processing element with the highest workload in the more unbalanced configuration has to execute more events. This explains the increase in the amount of time spent for executing useful events. In contrast, a processing element with the lightest workload in the more unbalanced configuration will execute fewer events. Consequently, it spends less time executing useful events.

In each of the unbalanced configurations, the processing element with the higher workload will advance its simulation clock slower than the other processing elements; hence it will not experience any significant number of rollbacks. Consequently, its overhead can be attributed mainly to the communication costs other than rollback, such as waiting for events from another processing element. The busier the processing element, the less time is spent on waiting, which explains the decrease in the time spent for overhead. On the other hand, processing elements with lighter workload execute fewer events so they may advance their simulation time ahead of the busier processing element. As a result, they have to rollback more often (see Table 2). This explains the increase in the time spent for overhead at the less busy processing elements.

Heterogeneous processing elements

In this section, we measure the effect of using heterogeneous processors on the performance of the tool. In the experiment, we ran the simulation for a period of 50 years with an initial population size of 640,000 family units, divided equally among all administrative areas. The probability of migrations was fixed at 60%. To emulate the difference in processor speed, we inserted a delay for every simulation year at one of the processors (1 second and 2 seconds for each experiment, respectively). This is done by adding a delay to the event that generates an annual report. The result is shown in Table 3. The result is consistent with what has been reported in literature on parallel simulation, i.e., the wider gap in processor speed will result in more rollbacks (see the last row).

Table 3: Effect of Unbalance in Processor Speed on Performance

Delay (second)	0	1	2
Number of migrations (individuals)	516,462	516,475	516,462
Time to complete simulation (minutes)	26.2	94.1	158.7
Total rollbacks	0	496,166	564,267

Heterogeneous communication latency

Finally, we are also interested in the effect of heterogeneous latency in the communication between processing elements. The event size used in Yades is 512 bytes. For this event size, we used the Intel MPI Benchmark Suite to measure the inter-node latency and intra-node latency and found that the inter-node latency was 16 times slower than the intra-node latency. In the experiment, we used the same configuration as in Section 4.3 but without any delay. We varied the locations of the four processing elements used in the experiment: using one compute node with four processing elements, using two compute nodes with two processing elements each, and using four compute nodes with one processing element each. The performance result is shown in Table 4. As expected, the number of migrations is about the same (row 2). The time spent in executing useful events is roughly the same because we expect similar number of useful events (row 4). The last two rows show that when the latency is homogeneous, the number of rollbacks is close to zero (row 6). As a result, it incurs some additional overhead cost (row 5). The overall performance (row 3) shows that a configuration with heterogeneous communication latencies (2×2) could perform worse than a configuration with higher but more homogeneous communication latencies (4×1) due to rollbacks.

Table 4: Effect of Unbalance in Communication Latency on Performance

Nodes × Processors	1×4	2×2	4×1
Total migrations (individuals)	516,462	516,129	516,462
Time to complete simulation (minutes)	26.2	29.6	28.2
Time to execute useful events (minutes)	10.0	10.3	10.3
Overhead (minutes)	16.2	19.3	18
Total rollbacks	0	15,167	13

CONCLUSIONS AND FUTURE WORK

We have presented the performance evaluation result of a parallel demographic simulation tool called Yades. The performance measures such as speed-up and scalability of the tool using up to 64 processing elements have been reported somewhere else (Onggo 2010). In this paper, we concentrate on the more fine grained performance measures such as: time spent in executing useful events, time spent for overhead and the number of rollbacks. Specifically, we have investigated the effect of three factors: unbalanced workload, heterogeneous processing speed and heterogeneous communication latency. The results are consistent with what has been reported in other application areas of parallel simulation. Since the application of parallel simulation in demography is new, it is useful to quantify the effect of the three factors on performance. The findings are useful because it is likely that the simulation users will run the tool using non-homogeneous configurations.

We plan to add new functionalities (such as allowing multiple administrative areas to be run on a processing element and introducing the concept of household which would allow one or more members of the same family unit to live in separate administrative areas), to implement a graphical user interface for the tool, and to validate the simulation results.

ACKNOWLEDGEMENT

This research is supported by the Royal Society International Joint Project 2009/R2 grant number JP090402.

REFERENCES

Ahmad S., and J. Billimek. 2005. “Estimating the health impacts of tobacco harm reduction policies: A simulation modeling approach”. *Risk Analysis* 25, No. 4, 801–812.

Bauer-Jr., D.W., C.D. Carothers, and A. Holder. 2009. “Scalable time warp on blue gene supercomputers”. In *Proceedings of the 23rd Workshop on Principles of Advanced and Distributed Simulation*. IEEE, Picataway, N.J., 35–44.

Bonnet, C., and R. Mahieu. 2000. Public pensions in a dynamic microanalytic framework: the case of France. In *Microsimulation modelling for policy analysis: challenges and innovations*, L. Mitton, H. Shuterland, and M. Weeks (Eds.). Cambridge University Press, Cambridge, UK, 175-199.

Eubank, S. 2002. “Scalable, efficient epidemiological simulation”. In *Proceedings of the 2002 ACM Symposium on Applied Computing*. ACM Press, N.Y., 139-145.

Forrester, J. 1971. *World Dynamics*. 2nd edition. Productivity Press, New York.

Hammel, E., C. Mason, and C. Wachter. 1990. SOCSIM II, a sociodemographic microsimulation program, rev. 1.0, operating manual. Graduate Group in Demography Working Paper No. 29, University of California, Berkeley.

Lan, C., and M. Pidd. 2005. “High performance simulation in quasi-continuous manufacturing plants”. In *Proceedings of the 2005 Winter Simulation Conference*, M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines (Eds.). IEEE, Picataway, N.J., 1367–1372.

Leombruni R., and M. Richiardi. 2006. “LABORsim: An agent-based microsimulation of labour supply: An application to Italy.” *Computational Economics* 27, No. 1, 63–88.

Lobb, C.J., Z. Chao, R.M. Fujimoto, and S.M. Potter. 2005. “Parallel event-driven neural network simulations using the Hodgkin-Huxley neuron model”. In *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*. ACM Press, N.Y., 16-25.

Meadows, D.H., D.L. Meadows, J. Randers, and W.B.-III William. 1972. *Limits to growth*. Universe Books, New York.

Meadows, D.L., D.H. Meadows, and J. Randers. 2004. *The limits to growth: The 30-year update*. Earthscan, London, UK.

Onggo, B.S.S. 2008. “Parallel discrete-event simulation of population dynamics”. In *Proceedings of the 2008 Winter Simulation Conference*, S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler (Eds.). IEEE, Picataway, N.J., 1047–1054.

Onggo, B.S.S. 2010. “Parallel discrete-event simulation tool for population analysis”. In *Proceedings of the 4th Operational Research Society Simulation Workshop*, M. Gunal, B. Tjahjono, S. Robinson, and S.J.E Taylor (Eds.). The Operational Research Society, Birmingham, UK, 167-175.

Orcutt, G. 1957. “A new type of socio-economic system”. *Review of Economic and Statistics* 58, 773–97.

Park A., and R.M. Fujimoto. 2009. “Efficient master/worker parallel discrete event simulation”. In *Proceedings of the 23rd Workshop on Principles of Advanced and Distributed Simulation*, IEEE, Picataway, N.J., 145–152.

Perumalla, K.S. 2005. “µsik – a micro-kernel for parallel/distributed simulation systems”. In *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*, ACM Press, N.Y., 59-68.

Perumalla, K. 2006. “Parallel and distributed simulation: Traditional techniques and recent advances”. In *Proceedings of the 2006 Winter Simulation Conference*, L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto (Eds.). IEEE, Picataway, N.J., 84–95.

Pressat, R. 1985. *Demographic analysis*. Aldine Atherton, Chicago.

Rao D.M., and A. Chernyakhovsky. 2008. “Parallel simulation of the global epidemiology of avian influenza”. In *Proceedings of the 22nd Workshop on Principles of Advanced and Distributed Simulation*, IEEE, Picataway, N.J., 1583–1591.

Rauner, M.S., S. C. Brailsford, and S. Flessa. 2005. “Use of discrete-event simulation to evaluate strategies for the prevention of mother-to-child transmission of HIV in developing countries.” *Journal of the Operational Research Society* 56, 222–233.

Roderick, P., R. Davies, C. Jones, T. Feest, S. Smith, and K. Farrington. 2004. “Simulation model of renal replacement therapy: predicting future demand in England.” *Nephrology Dialysis Transplantation* 19, 692–701.

Saysel A.K, Y. Barlas, and O. Yenigün. 2002. “Environmental sustainability in an agricultural development project: a system dynamics approach.” *Journal of Environmental Management* 64, 247–260.

Tang, Y., K.S. Perumalla, R.M. Fujimoto, H. Karimabadi, J. Driscoll, and Y. Omelchenko. 2005. “Optimistic parallel discrete event simulations of physical systems using reverse computation”. In *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*. ACM Press, N.Y., 26-35.

Walker, A., R. Percival, and A. Harding. 2000. The impact of demographic and other changes on expenditure on pharmaceutical benefits in 2020 in Australia. In *Microsimulation modelling for policy analysis*: *challenges and innovations*, L. Mitton, H. Shuterland, and M. Weeks (Eds.). Cambridge University Press, Cambridge, UK, 149-171.

Yoginath, S.B., and K.S. Perumalla. 2008. “Parallel vehicular traffic simulation using reverse computation-based optimistic execution”. In *Proceedings of the 22nd Workshop on Principles of Advanced and Distributed Simulation*, IEEE, Picataway, N.J., 145–152.

Zinn, S., J. Gampe, J. Himmelspach, and A.M. Uhrmarcher. 2009. “MIC-CORE: A toolCG Times; for microsimulation”. In *Proceedings of the 2009 Winter Simulation Conference*, IEEE, Picataway, N.J., 992–1002.

AUTHOR BIOGRAPHY

BHAKTI S. S. ONGGO is a lecturer at the Department of Management Science at the Lancaster University Management School. He received his MSc in Management Science from the Lancaster University and completed his PhD in Computer Science from the National University of Singapore. His research interests are in the areas of simulation methodology (conceptual model representations, modeling paradigms such as discrete-event and system dynamics), simulation technology (parallel and distributed simulation) and simulation applications. His current research includes the application of parallel simulation in policy analysis and simulation conceptual model representation. His email address is s.onggo@lancaster.ac.uk.

CRISTINA MONTAÑOLA SALES is a PhD student at the Statistics and Operations Research Department in Technical University of Catalonia (UPC). She is currently doing her research in the Department of Computer Applications in Science and Engineering from Barcelona Supercomputing Center (BSC). She holds an MSc in Computer Science from UPC. Her research interests include agent-based modelling, computer simulation, high-performance computing and computational social science. Her email address is cristina.montanola@bsc.es

JOSEP CASANOVAS GARCIA is a full professor in Operations Research, specializing in Simulation systems. He is one of the founders of the Barcelona School of Informatics (FIB) where he had acted as its Dean from 1998 to 2004. He also is the director of the LCFIB (Barcelona informatics school laboratory), an institution that has been very active in technology transfer to business. One of his recent projects has been the cooperation in the creation of simulation environments for people and vehicle flow in the new airport of Barcelona. He has led several EU funded projects in the area of simulation and operations research and is a strong advocate of the knowledge and technology transfer function between the university and society. His email address is josepk@fib.upc.edu.

NETWORK SIMULATION

A Framework for the Integration of Network Modeling and Simulation Tools

Eduardo M.D. Marques
Paulo N.M. Sampaio
Madeira Interactive Technologies Institute (M-ITI)
University of Madeira (UMa)
9000-390 Funchal
Madeira, Portugal
{emarques, psampaio}@uma.pt

KEYWORDS

Network description language, Modeling, simulation, XML

ABSTRACT

To provide the integration of different data network modeling and simulation tools is a complex task. Indeed, the underlying information structures used by these tools are, most of the times, very distinct and it is not simple to relate them. Moreover, the reutilization of network descriptions and other network related data among tools is not current. The use of a standard data structure, or language, to describe data networks would promote the interoperability among network tools, providing the users with the possibility of applying new platforms and tools to validate their network scenarios. This work proposes a framework to promote interoperability among network modeling and simulation tools which is based on a novel language, the Network Scenario Description Language to represent generic data networks scenarios.

INTRODUCTION

As many other kinds of applications, the network tools can also be developed as proprietary or open-source applications. The proprietary tools, usually are not free, and offer a broad possibility of options and functionalities, being supported by a more complete and accurate documentation. The open-source tools are free, normally being developed in order to provide the solution of some specific issues and problems, limited to some specific domains, and, most of the times, having a scarce and incomplete documentation.

It is also important to consider the functionalities provided by these network tools. Some tools are very limited in terms of number of functionalities available and have a very specific application. Others are based on a large set of models and provide a more complex analysis over a network. Thus, we can group these tools based on their functionalities.

A first group of tools provides the network modeling identifying the existing objects and their characteristics. These tools can be also automatic being able to scan a network infra-structure and to collect network information and configuration aspects in order to be able to build the network topology through a graphical user interface.

A second group of tools executes some algorithms to test or validate the represented data network. Examples of these tools are the network simulators or applications that evaluate a particular protocol.

A last group of tools allows monitoring the utilization of the network and collects data for further analysis. The data gathered can be visualized through simple statistics or more complex visualizations in order to provide a better understanding of the network behavior.

If all these tools were applied jointly in a coordinated way, they could provide a solid and helpful environment for optimizing the management of a network. Nevertheless, the data formats used in general by the tools are very distinct and, most of the times, incompatible.

In order to provide a generic solution for promoting interoperability among different network management tools, this paper presents a framework which relies on the Network Scenarios Description Language (NSDL), which has been proposed as a common solution that can be applied to assist network managers with the optimization of the network during its life cycle.

Fig. 1 illustrates the hierarchical organization of the several components of the NSDL framework. The top and bottom layers represent the existing networks tools. The top layer represents the management tools to provide the modeling, monitoring and visualization of networks (e.g., GUI's such as, topology generators, operation and failures monitoring, statistics and results, etc). The bottom layer represents the network analysis tools, such as network simulators, management tools, security evaluation, etc. Actually, some of the existing management platforms support both bottom and upper layers; however, most of them are purpose-oriented and are present only in one of the layers.

The NSDL representation represents a middleware layer to connect either layers or different networks tools. The GUI's may read and write the created network scenarios and,

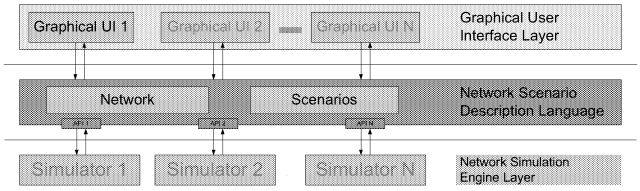


Figure 1: Composition of the NSDL Framework

through specialized Application Programming Interfaces (API's), the bottom layer tools are invoked to execute some operations over the network. In this paper we illustrate the utilization of the NSDL framework to provide the interoperability among network modeling and simulation tools.

The remaining of the paper is organized as follows: the section Related Works introduces some of the efforts to describe networks and to promote interoperability. In the section Network Scenario Description Language we present a new language to describe data networks. The section Case Study illustrates the integration of some tools for the modeling, simulation and analysis through the utilization of NSDL. The last section presents some conclusions and future works.

RELATED WORKS

An in-depth literature review has been carried out in order to identify the existing description languages for data networks, and the results demonstrated that the concern in finding a solution to represent this type of information is highly relevant. Some of these existing contributions are presented chronologically.

Network Description (NED) (Varga 1998) uses modules (simple and compound) and connections among them to allow the representation of any network topology. Its notation is very similar to a general programming language.

ANother Modeling Language (ANML) (Kiddle et al. 2001) is based on schemas, databases and models to represent, respectively, the components of the network, the repository of reusable components and specific simulation scenarios.

RElational Network Description Language (RENDL) (Estévez-Tapiador et al. 2003), implements the description of the network system through entities and relations established among them. It is used in the NSDF (Estévez-Tapiador et al. 2003) framework and it is focused in the network security domain.

Network Description Language (NDL) (van der Ham et al. 2006) is used to describe optical networks, and it uses several schemas to describe the network, such as: Topology, Layer, Capability and Domain.

Language for Network Meta-Description (LNMet-X) (Rahman et al. 2005) provides the network description in XML using the objects node, link, agent and traffic. Also supports the description of events in the context of network simulations.

Network Modeling Language (Netml) (Addie et al. 2006) is also based on XML, and it uses the objects node, link and traffic stream to support the description of a network. It also contains information about objects display properties.

Network Design Markup Language (NDML+) (Luntovskyy et al. 2007) contains a large set of categories to represent a network, such as project info, environment, constraints, topology, traffic, technology, test and cost. It was intended to be applied in 802.3 Ethernet, 802.11 WLAN and 802.16 WiMAX networks.

Some of these languages are already implemented in many of the available network tools, and, in particular in the domain of network simulation, such as in the network simulator software Omnet++, which applies NED.

The use of a normalized language could bring many advantages since, as stated by (Kiddle et al. 2001), "many tools can make use of a language that describes communication networks". In (Perrone et al. 2009), the authors, referring the network simulation domain, affirmed that "a standard language for scenario description would enable a number of additional improvements to the simulation workflow". Others authors reaffirmed the positive support that such a language could bring for users working in several network related domains (Estévez-Tapiador et al. 2003; Rahman et al. 2005; Addie et al. 2006; Canonico 2003).

All of these languages presented some similarities in the representation of a network topology, with its nodes, links and other network components. However, since each language focused on a particular network context, the description of a component in different languages resulted to be, most of the times, very distinct.

Also, by analyzing all the languages, some guidelines emerged as common to trace the root for a language that could be used as a standard to represent data network. So, the most referred guidelines are: (1) reusability (Varga 1998; Kiddle et al. 2001), to provide efficiency using the language; (2) independency of any tool (Rahman et al. 2005; Canonico 2003), and, (3) adaptability (Rahman et al. 2005; Canonico 2003), so it can represent current and new networks. Other terms were also proposed to characterize the language, such as extensibility and flexibility (Varga 1998; Rahman et al. 2005; Canonico 2003).

Analyzing the existing solutions, we concluded that it is relevant the existence of a generic description language able to support different network environments and network issues. This language should allow the representation of the existing networks with all their objects, relations and parameters, and it also should be extensible to accommodate new objects and future networks. Also, it should be possible to add information about one or several contexts of operation or analysis to complement the network description.

These requirements were considered during the definition of NSDL, which is presented in the next section.

NETWORK SCENARIOS DESCRIPTION LANGUAGE

The purpose of the Network Scenario Description Language (NSDL) is to provide a vocabulary and a set of rules, both able to support the description of wired and wireless data networks. More than just describing the network topology with its objects and characteristics, NSDL introduces an important concept: to separate the network description from the several perspectives that may exist over that network. For instance, one user could have been concerned with the network development; while another user could be at the moment responsible for the introduction of a new application in the, already, operating network. For these two intervenient and to describe many others different perspectives or contexts, we defined the element Scenario in order to include the description of the objects and parameters of a particular context. Each network scenario will have its particular functions and tools.

The principles applied to the design of NSDL were (1) simplicity, which means the language has to be simple and clear not only to be manipulated by an application or tool, but also possible to be edited by a human user with a simple text editor; (2) definition of multiple abstraction levels, allowing to specify not only simple high level descriptions of a network scenario, but also, if needed, to have the possibility to create a very detailed description of all network scenario objects and its parameters; and (3) extensibility, implying that new objects and parameters can always be incorporated in future descriptions.

The use of a single language description in several moments of a network life cycle could be very advantageous. If the users responsible for the network are familiar with that language, they can easily understand the current state of the network, thus management of the network is optimized. Further integration capabilities can also be achieved since developers of a network tool are able to add import and export capabilities from and to NSDL, making their work interoperable with NSDL networks.

Interoperability is indeed an important NSDL´s advantage. If a user deploys several NSDL compliant tools, he can easily analyze a network using each one of them, thus obtaining integrated results. For instance, NSDL can be useful when the user needs to execute two similar simulations over the same network, with two different simulators.

NSDL can be extended by adding new parameters and creating new objects over the already defined basic objects. For instance, NSDL can be extended to support the description of wireless sensor networks. From the basic object Node, we can define a new node, designated Wireless Sensor Node, which will receive all node characteristics, and will add its components, such as, special interfaces and sensor networks protocols.

The main goal of NSDL is to provide a rich description of the network objects and their parameters and also a description of the several network scenarios throughout the network life cycle. In this sense, the defined NSDL structure and parameters should also be rich enough to describe any type of data network and allow incorporating in its definition data to support future objects and new data networks.

The language which has been chosen to support NSDL is XML due its richness and flexibility. Indeed, XML provides the specification of clear definitions and has a set of available tools. Also,XML assures the validity of the NSDL principles: simplicity, abstraction and extensibility.

An NSDL representation is an XML file with two basic elements: *Network* and *Scenarios* (Fig. 2 and Fig. 3). The *Network* element contains the description of a network identifying its objects and its parameters. The *Scenarios* element may contain several descriptions, each one referring to a specific use, or context, to that network.

In the next section, the NSDL structure is presented. This section illustrates how a NSDL file is composed and which objects are applied to represent the networks.

Structure

A NSDL file is an XML file with two basic elements: *Network* and *Scenarios*. Fig. 2 and Fig. 3 are a simple NSDL example. The *Network* element contains the description of a network identifying its objects and its parameters. The *Scenarios* element may contain several descriptions, each one referring to a specific use, or context, to that network. A detailed description of the NSDL structure is presented on the sequence.

Network

The *Objects* element can be considered the core element of NSDL since it describes the network topology and the network characteristics. The description is done by the identification of all the network objects and their parameters. For this purpose three objects were defined: *Node*, *Link* and *Domain*, for the active equipment, for the links between nodes and domains and for networks abstract representations, respectively.

One unique characteristic of the *Node* object is the ability to incorporate other objects in it. Besides the *Node's* available parameters, these other objects that can be described in the *Node* are: *Interface*, *Protocol* and *Application*, each one to include the network objects they name.

```
<?xml version="1.0" encoding="UTF-8"?>
<nsdl>
    <network>
        <objects>
            <computer id="server">
                <http id="httpSr">
                    <dst.app>httpCl</dst.app>
                </http>
            </computer>
            <computer id="client">
                <http id="httpCl"/>
            </computer>
            <router id="router"/>
            <link id="ser_rou">
                <connection source="server"
                    destination="router"/>
                <bandwidth>10Mb</bandwidth>
            </link>
            <link id="rou_cli">
                <connection source="router"
                    destination="client"/>
                <bandwidth>100Mb</bandwidth>
            </link>
        </objects>
    </network>
    <scenarios>
    . . .
    </scenarios>
</nsdl>
```

Figure 2: Network description

```
. . .
<!-- extract of Visualization -->
<object id="computer01">
    <x.position>100</x.position>
    <y.position>100</y.position>
    <z.position>0</z.position>
    <image>computer.png</image>
</object>
. . .

<!-- extract of Simulation -->
<description>
    <general>
        <duration>100</duration>
        <simulator>ns2</simulator>
    </general>
    <events>
        <event objectid="appl01" time="10">
            <parameter name="action"
                value="start"/>
        </event>
    </events>
    <outputs>
        <output outputid="out01">
            <format>nstrace</format>
            <path>~/user/simulations</path>
            <filename>simul_01.tr</filename>
        </output>
    </outputs>
</description>
. . .
```

Figure 3: Visualization and Simulation Scenarios

Other two sub-elements also present in the *Network* element are *Templates* and *Views*. They are optional, and aim at assisting the user in the process of network description creation (modeling) and organization. The *Templates* help the user to create and to understand the NSDL file, minimizing long and redundant descriptions. The *Views* allows the creation of groups of objects in the NSDL networks, without any limitation. Although *Views* sub-element exists in the *Network* element to group network objects, its main objective is to collaborate with the *Scenarios* element.

Scenarios

The purpose of the element *Scenarios* is to bring additional and relevant information to the network, from several specific perspectives. The *Scenarios* element is composed of the sub-elements *Visualization* and *Simulation*.

The *Visualization* element allows the description of positioning and graphical information about a network scenario. In some network descriptions this kind of information might not be relevant and could eventually be ignored; however in others it is fundamental. One example relates to the need to add the precise positioning, using coordinates, to identify the correct place where all the objects are located.

The element *Simulation* scenario, as the name indicates, will describe all the relevant information to support simulations over the network. For instance, we can consider the identification of the tool/simulator; the general parameters of the simulation, such as duration, condition to finish (or to be interrupted); the events for the simulation; and, information about the statistics and expected results.

Other planned contexts to *Scenarios* include *Management*, *Security* and *Animation*, for network administration tasks, for verification and implementation of security in a network, and, for visual animation of scenarios, respectively.

In the next section we present a case study to illustrate the utilization of NSDL framework.

CASE STUDY

The structure of NSDL allows the representation of the network scenario and its elements (nodes, links, interfaces, protocols, application, etc.). Inside each one of these elements we still can describe many other parameters related to the topology, object characteristics, etc.. To illustrate the utilization of the NSDL framework for providing the interoperability of different network tools, we propose the utilization of some network tools according to the life cycle for a scenario (modeling, NSDL mapping and simulation/analysis), as presented in the following sections.

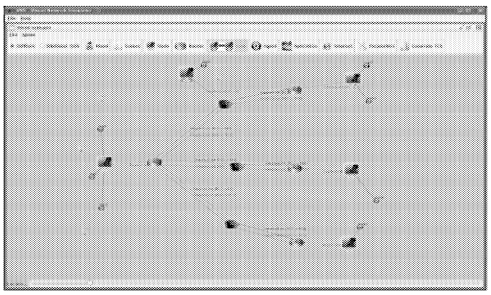


Figure 4: VNS Modeling Tool

Network Modeling

For the modeling of the network scenarios, we´ve adopted the network modeling tool called Visual Network Simulation (VNS) (Marques et al. 2008), which was developed at University of Madeira, and provides the broad representation of all the components of a network (nodes, links, protocol, traffic agents, etc.). Besides representing all the components of a scenario, VNS also provides the modeling of unicast/multicast scenarios, Differentiated Architceture (DiffServ) Scenarios, and further advanced object configuration. Fig. 4 illustrates a snapshot of the VNS modeling tool.

After having created and configured the scenario, VNS provides the automatic generation of NDSL based on the internal representation of the network scenario and its components.

The NSDL representation

The NSDL language provides a base object (called *Node*) to identify generically all the nodes in a network. Using the <node> element it is possible to specify some specialized abstraction for the objects, such as <computer>, <router>, among others. For each one of these objects, we can configure them according to their specific characteristics, also including their associated applications and protocols.

The <link> object can be applied to connect several nodes. As expected, some of its parameters are bandwidth and delay. A possible specialization for the object <link> is the object <fastethernet>.

In order to simplify the description of similar objects in the network, the Templates can be used. Another mechanism in NSDL to organize a network is the Views. With the views it is possible to create arbitrary groups of objects.

Fig. 2 depicts an excerpt of an NSDL *Network* representation. For simplicity reason we do not include templates and views in this code.

An important feature of NSDL is that it provides the integration of different network management tools. This integration can be achieved using the element Scenarios of this language. The Scenarios is useful to describe specific information related to a particular network tool, which offers a particular perspective over the described network. For the moment, two scenarios have been defined for the modeling and simulation tools, which are respectively the Visualizations and Simulations.

The Visualization scenario, for instance, allows the representation of positioning and graphical information associated with the network objects, as depicted in Fig. 3.

The simulation scenario allows the description of specific information needed to run the simulation of a particular network scenario. Fig. 3 also depicts an example of a simulation description.

Simulation/Analysis

As illustrated, the NSDL framework allows the integration of a network modeling tool with a network simulation tool

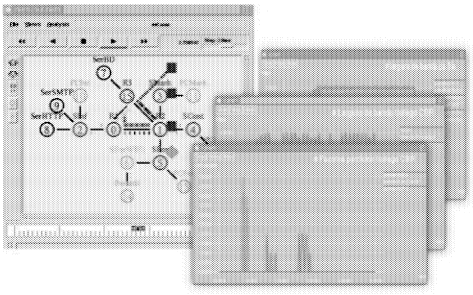


Figure 5: Simulation animation and analysis

for the verification of the correctness of the network topologies and their associate traffic distribution, eventually considering possible Quality of Service constraints.

In the case of the applied network tools, once VNS has generated automatically the NSDL representation, it is possible to apply XSL transformations in order to generate automatically compatible descriptions with any other tool. In this case, we generated OTCL, which is the format applied by Network Simulator 2 (ns-2) for the simulation of the network scenario and further utilization of its associated tools, such as Network Animator (NAM) (Estrim 1999) and xGraph, for the animation and analysis. Fig. 5 illustrates the utilization of NAM and xGraph for the animation and analysis of the simulation results.

The first experiments with the implementation of the NSDL framework allowed us to validate the expressivity of this language to describe completely all the components of a network, including their QoS features. For enabling the simulation of the network scenarios, the NSDL scenarios element was also improved to allow the complete description of the simulation characteristics related to ns-2. This experiment was relevant to realize the potential of NSDL as an integration medium able to provide the interoperability among different network management tools.

In order to evaluate NSDL, we applied this language for the modeling and simulation of network scenarios in an advanced network course at Madeira University. In this course, the students were asked to model their simulation scenarios using OTCL/ns-2 and NSDL. Among other objectives, the main purpose was to compare the development of the same scenario with both languages. Early results show that XML is a more common language to the students and that, although more verbose nature of XML, the NSDL structure was clearer in the description of the simulation environment.

In the last section we present some conclusions and future perspectives of this work.

CONCLUSION

The analysis of several languages to describe networks showed us that each language offers a particular structure to define a network, but, none of them, so far, emerged as standard to describe data networks and to provide the integration among different network tools. According to different authors, it would be important the existence of such a common language.

Our first contribution, the NSDL, is a proposal of that language. The most relevant difference of NSDL is separation of the network topology and objects from the several scenarios that may exist over that network.

The second contribution, the NSDL framework, was validated by integrating modeling to simulation network tools. Although, these are still some initial results, we intend to improve our framework with the development of new libraries to provide the integration of other tools in different domains.

As for future perspectives, we intend to improve NSDL to support different QoS solutions and also other types of networks such as Wireless Sensor Networks.

REFERENCES

Addie R.; S. Braithwaite; and A. Zareer. 2006. “Netml: a language and website for collaborative work on networks and their algorithms,” in *Australian Telecommunication Networks and Applications Conference*, 4-6 Dec 2006, Melbourne, Australia, 2006.

Canonico, R.; D. Emma; and G. Ventre. 2003. "An XML description language for Web-based network simulation," in *Proceedings of the Seventh IEEE International Symposium on Distributed Simulation and Real-Time Applications*. pp. 76-81, 23-25 Oct. 2003

Estévez-Tapiador, J. M.; P. García-Teodoro; and J. E. Díaz-Verdejo. 2003. “NSDF: a computer network system description framework and its application to network security,” In *Computer Networks*, vol. 43, pp. 573-600, Out. 2003.

Estrin D.; M. Handley; J. Heidemann; S. McCanne; Y. Xu;, and H. Yu. 1999. “Network Visualization with the VINT Network Animator Nam”, *tech. report 99-703*, Computer Science Dept., Univ. Southern California, Los Angeles, 1999.

Kiddle, C.; R. Simmonds; D. Wilson; and B. Unger. 2001. “ANML - A language for describing networks,” in *Proceedings of Ninth International Symposium Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 135-141.

Luntovskyy, A.; T. Trofimova; N. Trofimova; D. Gütter; and A. Schill. 2007. "To a Proposal towards Standardization of Network Design Markup Language," in *International Network Optimization Conference* (INOC'07), Spa, Belgium, April 2007.

Marques, E.M.D.; Ricardo A.S.A. Placido; and Paulo N.M. Sampaio, "Visual Network Simulator (VNS): A GUI to QoS simulation for the ns-2 simulator," in *Proceedings of the 2009 IEEE/ACS International Conference on Computer Systems and Applications*, pp.342-349.

Perrone, L. F.; C. Cicconetti; G. Stea; and B. Ward. 2009. “On the Automation of Computer Network Simulators”, in *Proceedings of SIMUTOOLS 2009*, Rome, 3-5 March 2009.

Rahman, A.; A. Pakstas; and F.Z. Wang. 2005. "An approach to integration of network design and simulation tools," In *Proceedings of the 8th International Conference on Telecommunications*, 2005. ConTEL 2005., vol.1, pp. 173-180, June 15-17, 2005.

van der Ham, J. J.; F. Dijkstra; F. Travostino; H. M. Andree; and C. T. de Laat. 2006. "Using rdf to describe networks," In *Future Generation Computer Systems*, vol. 22, no. 8, pp. 862-867, October 2006.

Varga, A. 1998. “Parameterized Topologies for Simulation Programs,” in *Proceedings of the Western Multiconference on Simulation (WMC'98)*, January 11-14, 1998, San Diego, CA.

MATHEMATICAL MODELLING OF THE HYDRAULIC LOAD OF COMMUNAL WASTEWATER NETWORKS[1]

Lidia Bartkiewicz*, Jan Studzinski**
*University of Technology Kielce
**Polish Academy of Sciences, Systems Research Institute
Newelska 6, 01-447 Warsaw, Poland
E-mail: studzins@ibspan.waw.pl

KEYWORDS: Mathematical modeling, neuronal nets, time series methods, wastewater networks.

ABSTRACT

In communal waterworks the whole water and sewage system consists usually of three basic objects: of water supply system, wastewater network and of sewage treatment plant. They are connected each other in series and the work quality of one of them affects the functioning of the following one. It means, that the water production for the waternet has an influence on the hydraulic load of the wastewater net and it decides of the raw sewage inflow entering the sewage treatment plant. This sewage inflow affects the quality of sewage purification and makes worse the treatment plant control in case of fast and big inflow changes. Because of that there is important to know in advance these inflow changes to have the opportunity to prepare the plant controllers on the oncoming events. A method to predict the sewage inflow changes is to model them. In the paper some mathematical models of raw sewage inflow using the neuronal nets and the time series methods are presented. For the modeling the real data from some Polish waterworks have been used.

INTRODUCTION

At the Systems Research Institute of Polish Academy of Sciences (IBS PAN) an intergrated information system for complex management of communal wterworks is under development. It consists of three subsystems for the water supply system, wastewater network and the sewage treatment plant. Each of the subsystems has the modular structure and the component modules are GIS, SCADA, optimization algorithms and mathematical models improving the management of the basic waterworks objects. To improve the control of the sewage treatment plant there is recommended to have the models with which the raw sewage inflow entering the treatment plant could be predicted. In the following such the models in form of the neuronal nets and the time series are shown and these models are components of the subsystem developed for the wastewater network. The data used to identify the models are coming from the waterworks in Rzeszow, a middle sized city in the south of Poland. The model calculations have been made using the software IDOL (developed at IBS PAN) in case of the time series models and using the software STATISTICA (developed by StatSoft) in case of neuronal nets.

SHORT DESRIPTION OF THE TIME SERIES METHODS

The main methods for the time series modelling are based on the classical least squares method whose advantage is the big simplicity and efficiency and also the clarity of its mathematical description. The calculation task of the time series methods consists in general in solving a system of linear algebraic equations, regarding the model parameters. The numerous existing variants of these methods differ one from another with regard to the statistical features of the parameter estimators calculated and to the accuracy and quickness of calculations. In the modelling calculations exercised three time series methods from the IDOL software have been used: the least squares Kalman method, the generalized least squares Clarke method and the maximum likelihood method. The first method is theoretically quickest and the least exact while the third one is relatively slow but the most exact. The Clarke method is more exact than this one of Kalman and quicker than the maximum likelihood method but less reliable of it, i.e. it can converge to the local solutions in case of complexer models.

The general descriptions of the process investigated and of the model are (Studzinski and Bartkiewicz, 2009):

$$y_n = -A(z^{-1})y_n - \sum_{i=1}^{M} B(z^{-1})x_{in} + v_n \tag{1}$$

$$\hat{y}_n = -\hat{A}(z^{-1})y_n - \sum_{i=1}^{M} \hat{B}(z^{-1})x_{in} \tag{2}$$

respectively, with n = 1, 2, ..., N, N - number of measurements data, M - number of model inputs, $A(z^{-1}), B(z^{-1})$ - difference operators for the output y_n and the inputs x_{in} of the process, $\hat{A}(z^{-1}), \hat{B}(z^{-1})$ - difference operators for the model signals. The process and model equations can be formulated in the matrix form:

$$y = \phi \cdot \gamma + v \tag{3}$$

$$\hat{y} = \phi \cdot c \tag{4}$$

where:

$y^T = [y_1, y_2, \ldots, y_N]$ process output

$\hat{y}^T = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_N]$ model output

$v^T = [v_1, v_2, \ldots, v_N]$ correlated noise

$\gamma^T = [\alpha_1, .., \alpha_R, \beta_{10}, .., \beta_{1P1}, .., \beta_{MPM}]$ process parameters

$c^T = [a_1, .., a_R, b_{10}, .., b_{1P1}, .., b_{MPM}]$ model parameters

[1] *The paper describes the results of the research project of Polish Ministry of Science and Higher Education No. NN 514 2977 33.*

$\varphi_n^T = [-y_{n-1},..,-y_{n-R}, x_{1n}, x_{1n-1},..,$ $x_{1n-P1},.., x_{Mn}, x_{Mn-1},.., x_{Mn-PM}]$ line n in matrix ϕ.

The estimator c of the process parameters γ is calculated by minimizing the following residual sum:

$$S_v(c) = \sum_{n=1}^{N} (y_n - \hat{y}_n)^2 = \sum_{n=1}^{N} \hat{v}_n^2 \quad (5)$$

with $\hat{v}$ - the estimator of correlated noise v.

The Kalman estimator resulted while minimizing (5) is:

$$c = (\phi^T \phi)^{-1} \phi^T y \quad (6)$$

and it is asymptotically biased. It would be asymptotically unbiased when

$$v = \varepsilon \quad (7)$$

i.e. for the uncorrelated noise.

The Clarke method is the least squares method applied to model a process described by (3) and with the additional description of the noise correlation in form of the relation:

$$(1 + D(z^{-1}))\, v_n = \varepsilon_n \quad (8)$$

Equation (8) in the matrix form is:

$$v = \theta \delta + \varepsilon \quad (9)$$

where:

$\varepsilon^T = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N]$ uncorrelated noise

$\delta^T = [\delta_1, \delta_2, \ldots, \delta_N]$ parameters of noise corelation

θ noise matrix with the line n in the form:

$$\chi_n^T = [-v_{n-1},.., -v_{n-S}] \quad (10)$$

The idea of this method is the transformation of (3) in such the way that the correlated noise v in it will be changed into the uncorrelated one like in (7). The parameters estimator (6) would be then asymptoticallky unbiased. It is to calculate from (5) after the process and model outputs have been transformated by multiplying (1) and (2) by the relation $(1 + \hat{D}(z^{-1}))$, where

$$\hat{D}(z^{-1}) = d_1 z^{-1} + ... + d_S z^{-S} \quad (11)$$

is the estimator of the correlated noise operator $D(z^{-1})$. This transformation means the filtering of the measurements data y_n and x_{in}. As a result the following process and model equations:

$$y^f = \phi^f y + \varepsilon \quad (12)$$

$$\hat{y}^f = \phi^f c \quad (13)$$

are obtained with:

$$y_n^f = (1 + D(z^{-1}))y_n \text{ and } x_{in}^f = (1 + D(z^{-1}))x_{in} \quad (14)$$

for n = 1,...,N, and with the transformed data matrix:

$$\phi^f = \phi^f(y^f, x_1^f, ..., x_M^f) \quad (15)$$

with the same structure as the primary matrix ϕ.

The residual sum for these equations is:

$$S_e(c) = \sum_{n=1}^{N} (y^f{}_n - \hat{y}^f{}_n) = \sum_{n=1}^{N} e_n^2 \quad (16)$$

with $\boldsymbol{e}$ – the estimator of the uncorrelated noise $\boldsymbol{\varepsilon}$.

The maximum likelihood method is applied to model processes described by equation (4) with the additional description of correlated noise v, which is different from this one in the Clarke method. To process equation (1) the following noise equation:

$$v_n = (1 + D(z^{-1}))\, \varepsilon_n \quad (17)$$

is added and the likelihood function:

$$L(\gamma) = f(y \,|\, x_1, \ldots, x_M, \gamma) \quad (18)$$

is formulated, where $y^T = [y_1, y_2, ..., y_N]$ is the process output, $x_i^T = [x_{i1}, ..., x_{iN}]$ for i=1,...,M are the process inputs, γ is the vector of process parameters and f is the function of likelihood density of random variable y.

The idea of this method consists in such the choice of the parameter values in γ, that function f will get the possibly greatest value for the measurements y and x_i. For the form of f is in general unknown, then the assumption is done of normal distribution $N(0,1)$ for the uncorrelated noise.

From (8) and (17) the following equation:

$$e_n = y_n + \hat{A}(z^{-1}) y_n - \sum_{i=1}^{M} \hat{B}(z^{-1}) x_{in} - \hat{D}(z^{-1}) e_n \quad (19)$$

results where $e^T = [e_1, e_2, ..., e_N]$ is the estimator of γ and it is a random variable of normal distribution $N(0,1)$, and $\hat{D}(z^{-1})$ is the estimator of difference operator $D(z^{-1})$. The likelihood density function for $e^T = [e_1, e_2, ..., e_N]$ is:

$$f_1(e_1,..,e_N \,|\, y, x_1,..,x_M, c, d) = \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{e_n^2(y, x_1,..,x_M, c, d)}{2}\right] \quad (20)$$

from the new likelihood function:

$$L_1(c) = \left(\frac{1}{2\pi}\right)^N \exp\left[-\frac{1}{2}\sum_{n=1}^{N} e_n^2(y, x_1, \ldots, x_M, c, d)\right] \quad (21)$$

results, where c and d are the estimators of process parameters γ and of correlated noise parameters δ, respectively.

The likelihood function L_1 for random variable e is equivalent to the likelihood function L for random variable y. Because of that the modeling task is formulated now as follows: For given data y and x_i such the value of parameters estimator c is to calculate that the possibly greatest value of function L_1 will be obtained. The task of maximizing L_1 can be replaced by the equivalent task of maximizing regarding c the following function resulted from L_1:

$$\ln L_1(c) = -\frac{1}{2}\sum_{n=1}^{N} e_n^2(y, x_1, \ldots, x_M, c, d) - \frac{1}{2} N \ln 2\pi \qquad (22)$$

The maximizing of $\ln L_1(c)$ is equivalent to minimizing the following loss function:

$$\mathrm{S(c)} = \frac{1}{2}\sum_{n=1}^{N} e_n^2(y, x_1, \ldots, x_M, c, d) \qquad (23)$$

The maximum likelihood method consists finally in finding the minimum of function (23) what is equivalent to minimizing the residual sum (16) in the Clarke method.

SHORT DESCRIPTION OF THE NEURONAL NETS

The artificial neuronal nets try to imitate in their assumption the operation of the biological neuronal nets of human beings (http://www.statsoft.pl/textbook/stneunet.html). This imaging is however very rough because of large quantitative limitations of artificial nets. The brain of human being consists of more than 10 billion neurons which are combined each other with more than several thousand connections. In case of an artificial net there are usually not more than several hundred neurons and not more than several dozen connections between two selected neurons. Another essential difference between a real and an artificial neuronal net is the division of this last one into layers on which the neurons are placed. Such the structure simplifiers the formulation of the mathematical net description. To each neuron on a layer the signals from the neurons located on the anterior layer are transmitted. The signals entered a neuron are multiplied by weight coefficients and accumulated and if the sum resulted is higher than the critical threshold value attached to the neuron than the neuron ignition succeeds and the totalled signal is converted by using a transition fuction related to the neuron. The signal value computed by the transition function means the output of the neuron under consideration. The mostly used transition function in neuronal nets is the nonlinear sigmoidal function.

The main layers of a neuronal net are input layer, output layer and some hidden leyers between them. The neuronal nets usually used have the feedforward structure, i.e. the signals are passing from the input layer via the successive hidden layers to the output layer what insures the net stability during the computations. The neurons on the neighbouring layers are connected “each to each” and a neuronal net of such the structure is called Multi Layer Perceptron, MLP.

By calculating neuronal nets a learning process is distinguished during which the end structure of the net is formulated. From the net the inter-neuronal connections are eliminated for which the values of the weight coefficients are stated as zero or close to zero. During the learning process the data from the learning set are used to model the network. The most known algorithm to learn the neuronal nets is the backpropagation algorithm, BP. With this algorithm the net parameters like the weight coefficients and critical threshold values are changed in the successive calculation steps in order to assure minimizing the error generated by the network during its modelling. This error is calculated using an error function and it is usually the squares residual sum. The BP algorithm is the algorithm of gradient optimization and it is coomonly used because of its simplicity, stability and high rate of computation.

An essential problem while learnig a neuronal net, i.e. while fitting it to the measurements data from the learning set, is gaining by it the ability for generalization, i.e. for making right forecasting with the use of other data sets either. Very often the correct neuronal net resulted from the learning set gains wrong results with other calculation data. This occurance is called the net overlearning. It means that the model resulted fits good to the small details represented by the single data and it is not able to imitate the main shape of the process modelled. A neuronal net with a big number of weight coefficients or hideen layers has usually a large tendency to an excessive adjustment to the data instead of ignoring their insignificant changes. The more complex neuronal nets reach almost at all times the smaller calculation errors than the simpler ones but it shows rather on the overlearning effect of them than on the good quality of the models. To avoid the overlearning of neuronal nets a validation approach is used. It consists in realizing a simulation run with the model resulted from the learning iteration and with another data set. The new data are used to check the quality of modelling done with the learning algorithm. If the quality of the model won by the learning process and the quality of the results won by the validation approach are similar then the assumption is made that the model constructed is correct. When the error resulted while learning the neuronal net is growing smaller in successive iterations and the error resulted from the validation runs is growing higher then it means that the neuronal net is going too far by matching the learning data and it loses gradually the ablity for generalization, i.e. it is going to be overlearned. In this case it has to be simplified by reducing the number of its hidden neurons or its hidden layers. The most important information by valuating the neuronal net is its validation error. In order to improve the valuation of the neuronal net a third data set, co called testing set, is isolated from the initial measurements data. Then the model calculated with the use of the learning set and verified using the validation set is tested additionally using this testing set. The testing calculation is done only once after the whole learning process is finished.

RESULTS OF MODELLING

The data used for the calculations has been got from the waterworks in the Polish city Rzeszow. They are daily measurements data series concerning the water production for the communal water net, raw sewage inflow reaching the sewage treatment plant, rainfalls data for the city Rzeszow and the water level values in the river flowing through the city. The number of measuremets in each of the data series is equal to 974.

On the first stage of modelling the time series methods of Kalman (K), Clarke (C) and of the maximum likelihood (ML) have been used. The models with three inputs, i.e. with the water production (WP), rainfalls data (R) and with the water level values in the river (WL); with two inputs, i.e. with WP and WL or with WP and R; and also with only one input, i.e. with WP or with WL or with R have been developed. As the single output of the models there the raw sewage inflow has been taken at all times. While developing different models

also different orders of their difference operators have been tested. To evaluate the models the following criteria have been used:

- *standard deviation of the model error* ($\sigma_{\hat{x}}$):

$$\sigma_{\hat{x}} = \sqrt{\frac{1}{N} \cdot \sum_{t=1}^{N} (\hat{x}_t - \bar{\hat{x}})^2} \text{ with } \bar{\hat{x}} = \frac{1}{N} \sum_{t=1}^{N} \hat{x}_t ,$$

- *standard deviations quotient* (ξ): $\xi = \frac{\sigma_{\hat{x}}}{\sigma_x}$

with $\sigma_x = \sqrt{\frac{1}{N} \cdot \sum_{t=1}^{N} (x_t - \bar{x})^2}$ and $\bar{x} = \frac{1}{N} \sum_{t=1}^{N} x_t$,

- *Mean Squared Error* (*MSE*):

$$MSE = \frac{1}{N} \sum (x_t - \hat{x}_t)^2 = \frac{1}{N} SSE ,$$

- *Pearson correlation criterion* (*R*):

$$R = \frac{\sum_{t=1}^{N} (x_t - \bar{x})(\hat{x}_t - \bar{\hat{x}})}{\sqrt{\sum_{t=1}^{N} (x_t - \bar{x})^2} \cdot \sqrt{\sum_{t=1}^{N} (\hat{x}_t - \bar{\hat{x}})^2}} ,$$

- *Acaike Information Criterion* (*AIC*):

$$AIC = \left[2{,}83788771 + \ln\left(\frac{SSE}{N} \right) \right] \cdot N + 2L_p$$

where x_t and $\hat{x}_t$ mean the measurements data and the model output at the time *t*, *N* is the number of data in the data series and L_p is the number of the model parameters.

Table 1. Time series models with three inputs.

Model/order/ inputs	*Evaluation criteria*				
	$\sigma_{\hat{x}}$	ξ	*MSE*	*R*	*AIC*
K/6/WP-WL-R	3.164	0,47	100,1	0,87	18,518
C/6/WP-WL-R	3.164	0,49	100,1	0,87	18,530
ML/3/WP-WL-R	3.822	0,60	146,1	0,80	18,858

Table 2. Time series models with two inputs.

Model/order/ inputs	*Evaluation criteria*				
	$\sigma_{\hat{x}}$	ξ	*MSE*	*R*	*AIC*
K/6/WP-WL	4.081	0,64	166,5	0,77	19,000
C/6/WP-WL	4.081	0,64	166,5	0,77	19,002
ML/3/WP-WL	4.354	0,68	189,6	0,74	19,106
K/6/WP-R	3.320	0,52	110,2	0,85	18,598
C/6/WP-R	3.320	0,52	110,2	0,85	18,600
ML/3/WP-R	3.902	0,61	152,3	0,79	18,893

In Tables 1, 2 and 3 the best time series models determined with different methods, for different numbers of the inputs and different orders of the difference operators are shown (Bartkiewicz, 2010). One can see that the best results of modeling have been got with the Kalman model of sixth order and with all three inputs considered, for which the minimal value of *AIC* criterion has been got. The percentage error of the sewage inflow forecast calculated for this model is equal to 4,53 %. The worst model of this family is this one calculated with the maximal likelihood method of third order for which the highest value of *AIC* and the inflow forecast error equal to 6,45 % have been received. The models that have been determined for the reduced numbers of the inputs are worse than the models with the full inputs set. In Figures 1, 2 and 3 the results of the modeling got for the best models of Kalman, Clarke and of the maximal likelihood are shown.

Table 3. Kalman models with one input.

Model/order/ inputs	*Evaluation criteria*				
	$\sigma_{\hat{x}}$	ξ	*MSE*	*R*	*AIC*
K/6/WP	4.272	0,67	182,5	0,74	19,075
K/6/R	3.624	0,57	131,3	0,83	18,754
K/6/WL	4.333	0,68	187,8	0,75	19,103

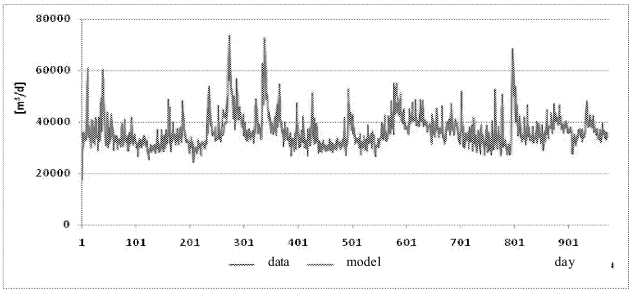


Figure 1. Modeling results for the best Kalman model.

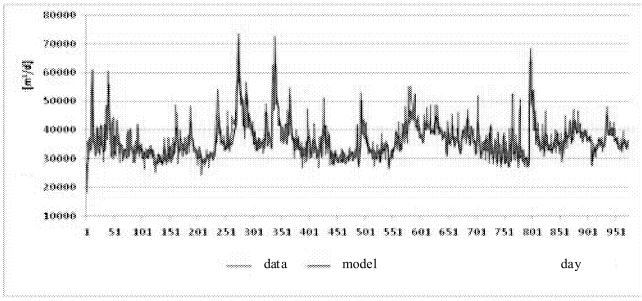


Figure 2. Modeling results for the best Clarke model.

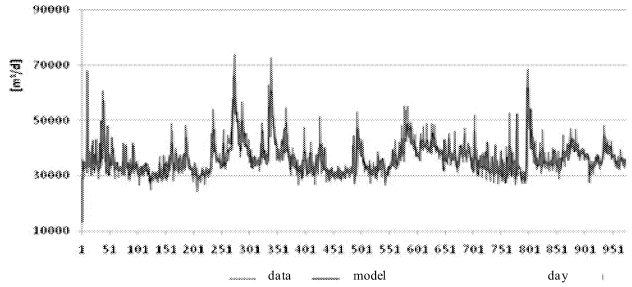


Figure 3. Modeling results for the best maximum likelihood model.

The results received with the time series methods show that although the simplest Kalman model is the best then the other ones are in general not much worse when they are compared each other quantitatively in view of the criteria values as well as qualitatively in view of their diagrams.

On the second stage of modeling the neuronal nets of type MLP have been used. In this case also the models with only

one input (WP or WL or R), with two inputs (WP and WL or WP and R) and with three inputs (WP and WL and R) have been tested and different time delays in the data series introduced onto the inputs have been either defined. The numbers of these time delays are equivalent to the orders of difference operators by the the time series methods. The output of the neuronal nets investigated is only one at all times and it means the raw sewage inflow to the sewage treatment plant. A neuronal net marked in the following for instance as MLP/1/3/3-6-1 means the MLP net with the delay (shift) in the data equal to 1 day, with 3 inputs, with 3 neurons on the input layer, 6 neurons on the hidden layer and with 1 neuron on the input layer.

In Table 4 the best neuronal net models determined for different numbers of the inputs and different delay numbers in the measurements data are shown (Bartkiewicz, 2010). Like the results of the time series methods also by modeling with neuronal nets the model with the full inputs set (MLP/4/3/12-5-1) turned out to be the best. It has got the smallest *AIC* value and the highest *R* values under all other models which are the best in their classes. Its *R* values are also quite similar for all modeling processes, i.e. for learning, validation and testing, what shows on its good features of forecasting. In this model the measurements data in the input series are shifted of four days what corresponds with the difference operator order equal to 4 in the time series methods.

Table 4. The best neuronal net models.

Model	*Criterion R*			*AIC*
	learning	test	validation	
1 input – daily water production WP				
MLP/1/1/1-2-1	0,10	0,15	0,17	30,1
1 input – daily water level in the river WL				
MLP/2/1/2-3-1	0,67	0,45	0,61	30,2
1 input – daily rainfalls R				
MLP/3/1/3-3-1	0,49	0,59	0,53	32,0
2 inputs (without R)				
MLP/1/2/2-3-1	0,76	0,73	0,78	33,1
2 inputs (without WL)				
MLP/2/2/4-3-1	0,51	0,62	0,55	37,6
2 inputs (without WP)				
MLP/2/2/4-4-1	0,68	0,46	0,64	43,9
3 inputs (WP and WL and R)				
MLP/4/3/12-5-1	0,83	0,76	0,79	17,5

In Figure 4 the output of the best neuronal net model and the measurements data are drawn and in Table 5 the best models determined with two different methods are compared in view on their main criterion values. There is to see that the model of Kalman is better than this one of neuronal net although the differences in their criteria values are rather slight. Also the diagrams drawn for the both models are alike (see Fig. 1 and Fig. 4).

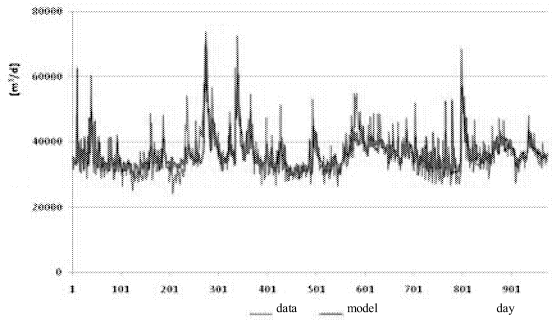


Figure 4. The best neuronal net model MLP/4/3/12-5-1.

Table 5. Comparison of the best time series and neuronal net models.

Model	*Evaluation criteria*		
	ξ	*R*	*AIC*
K/6/WP-WL-R	0,47	0,87	18,52
MLP/4/3/12-5-1	0,63	0,79	17,50

CONCLUSIONS

In the paper the results of modeling the raw sewage inflow into a sewage treatment plant have been presented. The models are developed with the time series methods and the neuronal nets. The results received show that the simpliest method of Kalman gets better models than the other more complicated time series methods. It is also better than the much more complex method of neuronal nets although the differences between the results are not essentially big. The sewage inflow models are meant for the forecast goals and they are to be included into an information system improving the control of the sewage treatment plant and the management of the communal waterworks.

REFERENCES

Bartkiewicz L.: „*Modelling of the sewage inflow entering the sewage treatment plant”*. Dissertation Thesis, University of Technology Kielce, 2010 (*in Polish*).

Bogdan L., Studzinski J.: „*Modelling and forecasting of the waternet load using the neuronal nets”*. Conference Proceedings, BOS’2010, Bydgoszcz (*in Polish*).

Studzinski J., Bartkiewicz L.: „*Methods and programs supporting the solution of system modeling and system identification tasks”*. INSTAL, 4A, 2009, 59-64 (*in Polish*).

Tadeusiewicz R.: „*Neuronal nets”*. Warszawa: Akademicka Oficyna Wydawnicza, 1993 (*in Polish*).

http://www.statsoft.pl/textbook/stneunet.html.

A Combined Traffic and Radio Network Simulation based on Predictive Scenarios

Sebastian Šubik, Christian Lewandowski and Christian Wietfeld
Communication Networks Institute (CNI)
TU Dortmund University, Germany
email: {sebastian.subik|christian.lewandowski|christian.wietfeld}@tu-dortmund.de

Daniel Weber and Michael Schreckenberg
Physics of Transport and Traffic (PTT)
University Duisburg-Essen, Germany
email: {weber|schreckenberg}@ptt.uni-due.de

KEYWORDS
Simulator Coupling, Network Planing, Infrastructure, Traffic Simulation

ABSTRACT

Modern network planning relies mainly on the use of simulation tools. They could either be used to optimize the radio coverage or to calculate the dimensions of the connection between different nodes (e.g. traffic load of mobile radio cells or required bandwidth for the core network). Because of the growing mobility of the users, the simulation of the traffic can not be neglected. The problem is that traffic simulations should be aware of relying only on assumptions of the behavior. To increase the impact of the simulation results, provable facts of the behavior of the simulated objects (cars or people) are used as start values.
In this paper, we present a combination of different simulation tools which enable the traffic simulation to use static anonymous data from the mobile networks to compensate for unavailable data in large-scale traffic networks.

INTRODUCTION

Simulations provide an established tooling for many problems in nowadays fields of research. Multiscale simulations (Lewandowski et al. (2008)) integrate many aspects into one simulation to enhance the quality of the results. In actual research, many simulations for only one aspect like traffic, network protocols or mobility exists. Every tool has its own advantages and disadvantages, some only rely on theoretical models, other use statistical data as starting values or live-date for a actual prediction. In this paper we demonstrate a way to increase the quality. Therefore different simulation tools as well as life-data should be combined in one simulation.
The paper is structured in the following way: At first, an existing traffic simulation is described, which is used for a traffic forecast in North-Rhine Westphalia. In the following chapter, a new model for enhanced data models is outlined and a possible source for the additional needed data is identified. The last chapter describes an example simulation setup, in which different simulation tools are combined to gain better overall simulation results.

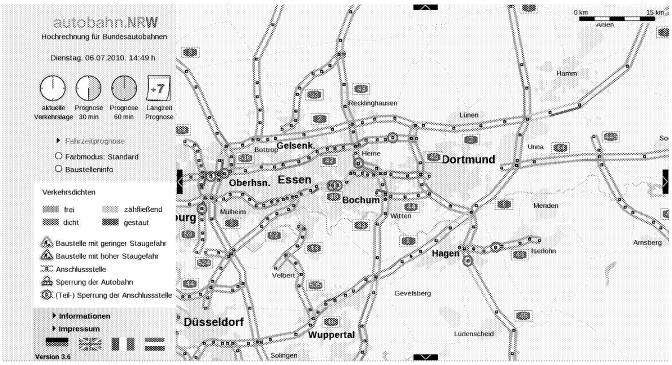


Figure 1: Traffic Information System

TRAFFIC SIMULATION

The traffic information system OLSIM (**O**n**L**ine Traffic **SIM**ulation) was established to provide the user with precise traffic information for the highway network of North Rhine - Westphalia. The current traffic state as well as several prognosis are calculated. To generate the traffic state for the whole network the system uses locally measured traffic data of about 4500 loop detectors and feeds them into a microscopic simulation model of traffic flow. The data contain the flow and the velocity for passenger cars and trucks and the occupation of the detectors and are sent every minute to the simulation server.

Simulation Model

The microscopic traffic simulation model used in OLSIM is derived from the Nagel-Schreckenberg (NaSch) cellular automata model (Nagel and Schreckenberg (1992)). In the cellular automata approach time as well as the position, speed and acceleration of the cars are treated as discrete variables and the street is represented as a lattice of cells which are occupied by at most one vehicle at a given time. At each time step $t \rightarrow t+1$, the state of the system is updated according to a set of rules. In the original NaSch model a cell size of 7.5 m is used and the update scheme contains a stochastic noise term (the dawdling parameter p), which produces spontaneous fluctuations resulting in the emergence of traffic jams at higher densities.
The model currently used in OLSIM uses smaller cells, different classes of vehicles, multiple lanes and a greatly modified update scheme with a slow-to-start rule and anticipation, to reproduce empirically observed phenomena like meta-stable traffic states. A smaller cell size of 1.5 m leads to a more realistic acceleration and more speed bins, vehicles then occupy a number of consecutive cells depending on their length. Two classes of vehicles are used, passenger cars and trucks, where trucks have greater length, lower maximum velocity and different lane changing behaviour.
To achieve a more realistic driving behavior velocity dependent randomization and anticipation was introduced in the modeling.
Consider a vehicle n with position $x_n(t)$, velocitiy $v_n(t)$ and distance to the next car $d_{n,m}(t)$. The brake light variable $b_n(t)$ signals if it is braking or not. In the original NaSch model wether a car should brake or not is determined solely by the distane $d_{n,m}(t)$ to the next car ahead. Taking into account the movement of car m leads to the effective gap

$$d^{eff}_{n,m}(t) = d_{n,m}(t) + \max\left(v^{min}_m(t) - d_S, 0\right) \qquad (1)$$

with the fixed safety distance d_S and a lower bound for the movement of vehicle m in this time step

$$v^{min}_m(t) = \min\left(d_{m,l}(t), v_m(t)\right) - 1. \qquad (2)$$

A vehicle reacts to the brake light of the next car ahead, if they are inside a velocity-dependent temporal interaction horizon

$$t^h_{n,m}(t) := \frac{d_{n,m}(t)}{v_n(t)} < \min\left(v_n(t), h\right) =: t^S_n(t). \qquad (3)$$

The constant randomization of the NaSch model is repolaced with a velocity dependent probability function

$$\begin{aligned} p_n &= p\left(v_n(t), b_m(t)\right) \\ &= \begin{cases} p_b, \text{if } b_m(t) = on \text{ and } t^h_{n,m}(t) < t^S_n(t) \\ p_0, \text{if } v_n(t) = 0 \\ p_d, \text{else} \end{cases} \end{aligned} \qquad (4)$$

The update scheme then consists of the following steps:

1. Initialization
   Set $p_n := p(v_n(t), b_m(t))$ and $b_n(t+1) = off$
2. Acceleration
   If $b_n(t) = on$ or ($b_m(t) = on$ and $t^h_{n,m}(t) < t^S_n(t)$)

$$v_n\left(t + \frac{1}{3}\right) = v_n(t) \qquad (5)$$

   else

$$v_n\left(t + \frac{1}{3}\right) = \min(v_n(t) + 1, v_{max}) \qquad (6)$$

3. Braking

$$v_n\left(t + \frac{2}{3}\right) = \min\left(v_n\left(t + \frac{1}{3}\right), d^{eff}_{n,m}(t)\right) \qquad (7)$$

4. Randomization
   with probability p_n:

$$v_n(t+1) = \max\left(v_n(t) + \frac{2}{3} - 1, 0\right) \qquad (8)$$

   else

$$v_n(t+1) = v_n\left(t + \frac{2}{3}\right) \qquad (9)$$

5. Move

$$x_n(t+1) = x_n(t) + v_n(t+1) \qquad (10)$$

Note that these steps are executed for all vehicles in parallel. There are additional rules to model lane changing, for a full account see (Hafstein et al. (2004))

RADIO NETWORK ENHANCED TRAFFIC SIMULATION

The model described in the last chapter accounts for the dynamics of the vehicles in a single network link, but it does not provide any information for the distribution of traffic in a large-scale network. This is no big problem when considering a freeway network where junctions are scarce and the sources and sinks where vehicles can enter and leave the network well defined. In the online simulation of the freeways in North Rhine - Westphalia the lack of information on the distribution of traffic is compensated by employing the data of the many loop detectors to adjust the simulation results to measured values by inserting and deleting simulated vehicles. But this approach will not work when extending the simulation to urban road networks, where the density of junctions gets higher and the coverage by traffic detectors (usually) scarcer. As a dense network of traffic detectors is quite expensive another source of information is needed. One possible set of data could be traffic

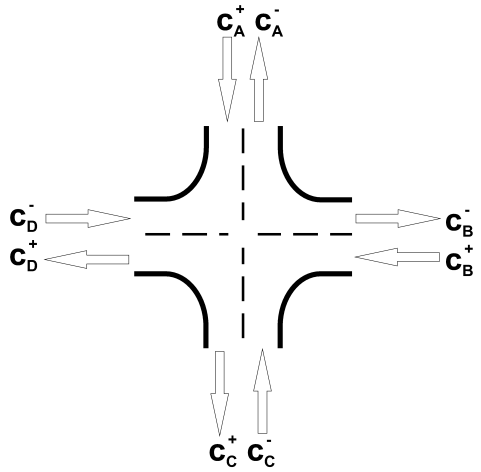


Figure 2: Elementary junction representation

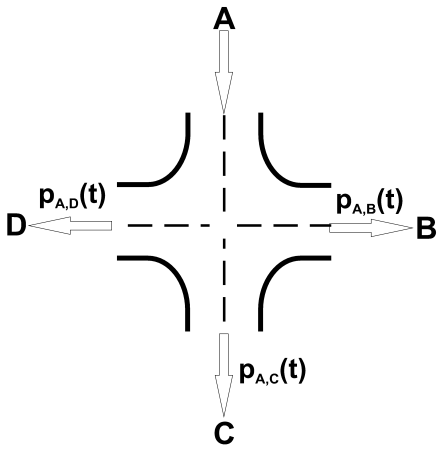


Figure 3: Extended junction representation

assignment, though in practice these kind of data are often hard to obtain, especially for areas comprising different cities or planning authorities. In the following we present an alternative approach to enhance the traffic simulation with data from mobile radio networks.

Junction Enhanced Traffic Model

In the simplest case the simulated links are connected with each other by elementary junctions as depicted in Figure 2 . In this model, junctions can be described as input and output flows of cars. As there are no other sinks or sources the number of vehicles is conserved and the sum over all entering and exiting cars has always to be zero.

$$\sum_{i=A\dots D} c_i^+ - c_i^- = 0 \tag{11}$$

But this is only the case for complete knowledge of all the flows. In reality we have a network with imcomplete measurements of the flows to and from the junction and possibly additional unknown sources and sinks in between so that equation 11 changes to

$$\sum_{i=A\dots D} c_i^+ - c_i^- \neq 0 \tag{12}$$

This leads to a systematic error in the simulation which will increase the difference between the predicted traffic and the real traffic on the streets.

To prepare the model for additional data, we slightly change our picture of the junction. Instead of the description of a junction as the sum of input and output flows we use a set of turning probabilities, because in our stochastic traffic flow model we are interested in the probability for a car entering the junction on road A at time t to turn to road B, C, D or return on the same road A. Therefore, a representation as depicted in Figure 3 could be used and all probabilities together set up a matrix $P_{junction}$, which describes the junction:

$$P_{junction}(t) = \begin{vmatrix} p_{A,A}(t) & p_{A,B}(t) & p_{A,C}(t) & p_{A,D}(t) \\ p_{B,A}(t) & p_{B,B}(t) & p_{B,C}(t) & p_{B,D}(t) \\ p_{C,A}(t) & p_{C,B}(t) & p_{C,C}(t) & p_{C,D}(t) \\ p_{D,A}(t) & p_{D,B}(t) & p_{D,C}(t) & p_{D,D}(t) \end{vmatrix} \tag{13}$$

$$\sum_{i,j=A\cdots D} p_{i,j}(t) = 1 \tag{14}$$

Each element of the matrix contains the probability $p_{i,j}$, where i and j are the streets for entering and exiting the junction (for $i \neq j$). For the case of $i = j$ the probability of a new occurrence of a car at the exiting street i is given. This enables the use of the extended junction model for a non completed modeling of the streets. In this case, Equation 14 changes to

$$\sum_{i,j=A\cdots D} p_{i,j}(t) \neq 1 \tag{15}$$

It could be argued, that the same effect could be seen in Equation 12. The problem with this approach is that for the NaSch model, the speed as well as the flashing of the braking-lights is taken into consideration (see Equation 1). In the elementary model of the junctions, only the overall changes of streets could be modeled, but not the actual distribution. For example, if the entering flows on two streets have the same size, it is not possible to differentiate between $A \to C, B \to D$ and $A \to D, B \to C$. The second version has a much higher traffic-jam potential, because the maximum capacity of cars passing the junction is lower while the cars need to reduce their maximum speed to turn around.
Thus, in the extended junction model, this effects can be taken into consideration as well, which can lead to

more exact simulation results compared to real world traffic.
The probability for a car to enter a junction and to exit it on a given street depends on the actual time t, which represents the different traffic condition allover one day (or even for different days like weekends). It also depends on the position x(t), the actual position and older positions x(t-n). So it is a function like

$$p_{A,B}(t, v(t), x(t-1), \cdots, x(t-n)) \tag{16}$$

To enrich the simulation with the extended junction model, all this values need to be extracted from different real world sources. As seen so far, the solely use of loop detectors could not provide the simulation with enough information.
To face this problem, an alternating source for the data is described in the following chapter.

Radio Network Generated Data

Regarding to the nowadays research, mobile radio network provider have a huge set of data, which can be used as the foundation of future mobility simulations. In some projects, the data of the mobility of the subscribers is used to generate live-traffic announcements (Schäfer (2009)). Most of these projects rely on an active upload channel over which the mobiles send their actual position to the system or can only give a up-to-date traffic information without prediction of future developments.
The problem for all systems is that most of the data collected by network providers is strictly personalized and therefore protected by law to ensure data security. Another disadvantage is, that the provider do not have the exact position of their mobile subscriber, but only the actual serving cell with the receive level of the mobile station is logged.
Considering the last fact, an isolated analysis of the radio coverage of the cells could not lead to the expected results, because in modern mobile radio systems (e.g. LTE, UMTS, GSM), the best serving cell is not automatically used for communication. If signal-strength would be used as the only criteria for a handover between two cells, a well known effect (ping-pong Handover) would accrue, in which the mobile permanently switches from one cell to another, if both cells have nearly the same signal level.
Instead of using the overall best cell, the progress of the receive level of all neighboring cells is monitored. As depicted in Figure 4, the handover process is only started, if a given handover margin is reached. The effects of the fast- (multipath-) fading overlay the long term fading and the handover margin has to be set to a value which is great enough to handle the fading effects. This avoids a frequently switching between different base stations.

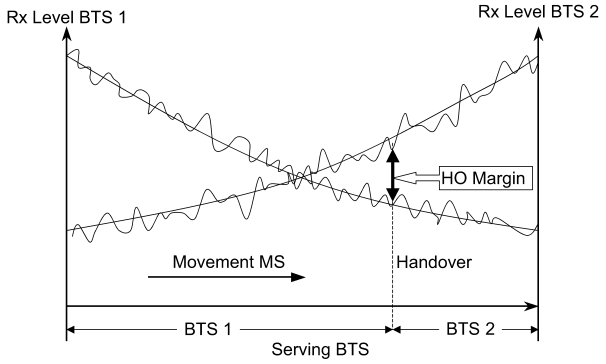


Figure 4: Handover margin in Mobile Radio Networks

Table 1: Dataset from Radio Network Provider

Datafield	Description
TIMSI	temporary id for a mobile station
serving cell id	base station connected with moblie
time	a timestamp

Enhanced Traffic Model with Mobile Radio Network enriched Data

The goal for the simulation described in this paper is to utilize the radio network generated data to enrich the traffic simulation to gain better results. The data should be used for the enhanced junction model as described in the preceding section. From now on we assume, that we have access to a defined static set of anonymized data from a radio network provider.
To extract all needed information, an area for the analysis has to be defined. In the center of the area, the junction for which we want to generate our probability matrix should be located. Additionally we need the exact position of all base station from the network provider.
The real world data should have a format like shown in Table 1. It is important, that this data is anonymized, it is not possible to deduct to the original personal data of the subscriber. From this dataset, traces of connected cells by a specific subscriber over the time could be extracted and ordered as depicted in Table 2.
Now this data need to be converted to a probability matrix P. For this, a agent based simulation can be utilized (as described in the following chapter). In contrast to the Nagel-Schreckenberg model which is used for the prediction of traffic flows and jams, the agent based simulation is used to identify the matching combinations of time cell-id combinations for the different routes. This enables the construction of the probability matrix.

Included into this process is a plausibility check to exclude unwanted combination of serving cells (e.g. turnarounds of cars). Also, the average speed could be extracted from the data and be used as an input for

Table 2: Extracted Data Set

TIMSI:		time	cell id		time	cell id		...
⋮		⋮	⋮		⋮	⋮		⋮

the following simulations. To increase the accuracy, the among of past cells could be analyzed. This is necessary to determine the size of the the dataset which needs to be requested from the radio network provider.

COMBINED SIMULATION

To show the functionality of the traffic jam forecast proposal, a multiscale simulation environment based on the event-based network simulation engine OMNeT++ 4 (Varga and Hornig (2008)) and Simulation of Urban Mobility (SUMO) microscopic road traffic simulation package will be introduced. Figure 5 shows the interaction between the simulation platforms. OMNeT++ and SUMO are connected via a TCP socket connection which allows bidirectional communication between both applications (Wegener et al. (2008)).
For realistic traffic scenarios geographical material is obtained from Openstreetmap (OSM), a free world map. The XML based maps are converted with the integrated *netconvert* tool to be used in SUMO. For displaying the road network within OMNeT++ and to locate the exact position of each car, a background image will be rendered from the converted SUMO network file.
The OMNeT++ simulation model consists of two main entities: the vehicle and the base station (BS). Both are based upon the same layer 1-4 configuration shwon in Figure 6. Layers 1 and 2 are based on the IEEE 802.11 protocol stack, as wifi is adequate for showing the functionality of the concept described in this paper. For other mobile communication technologies like GSM and UMTS the functionality will be the same as the base station will track all mobile stations connecting to the network cell.

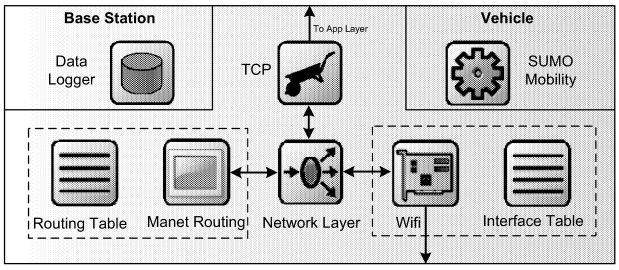


Figure 6: Simulation Model Layer Setup

Layer 3 and 4 utilize a standard TCP/IP stack with mobile ad-hoc network (MANET) routing functionalities and TCP on the transport layer. In order to simulate mobility scenarios a mobility option based on (Wegener et al. (2008)) is included. As MANET algortihm OLSR is used for generating routes between vehicles an base stations. Every time OLSR adds a route to the routing table, the routing table module increments a counter in the data logger which provides the information to the network operator later on.

Simulation scenario

The highway network of North Rhine-Westphalia, especially in the Ruhr Area is very interesting for traffic jam investigations, as the heavy commuter traffic leads to problems on the highways. Therefore the scenario in Figure 7 is used for the validation of the proposal.

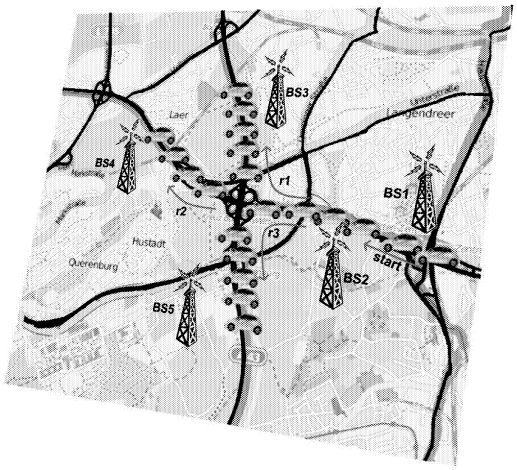


Figure 7: Simulation Scenario

It shows a highway junction of two highways with the specialty that one highway ends and leads to the center of a big city. A screenshot of the simulation is laid on top of an OSM map to show the environmental conditions. All vehicles will start the routes in the east of the highway and are connected to BS1. They are driving towards the highway junction, connect to BS2 and will then take different routes.

- Route r1: vehicles go north
- Route r2: vehicles go west to city center
- Route r3: vehicles go south

Vehicles that take route r1 will connect to BS3 on their way north, vehicles on route r2 will connect to BS4 and vehicles on route r3 will connect to BS5 in the south of the highway junction. With this connection patterns we can calculate a propability p that cars take the assumed routes so that the traffic jam forecast can be andvanced and the mobile phone providers can switch the capacities of considered cells in case of heavy traffic. In our scenario we assume: $p(r1) = p(r2) = p(r3) = \frac{1}{3}$ In this scenario setup, the combinations for the three different routes could be identified. The inclusion of the channel

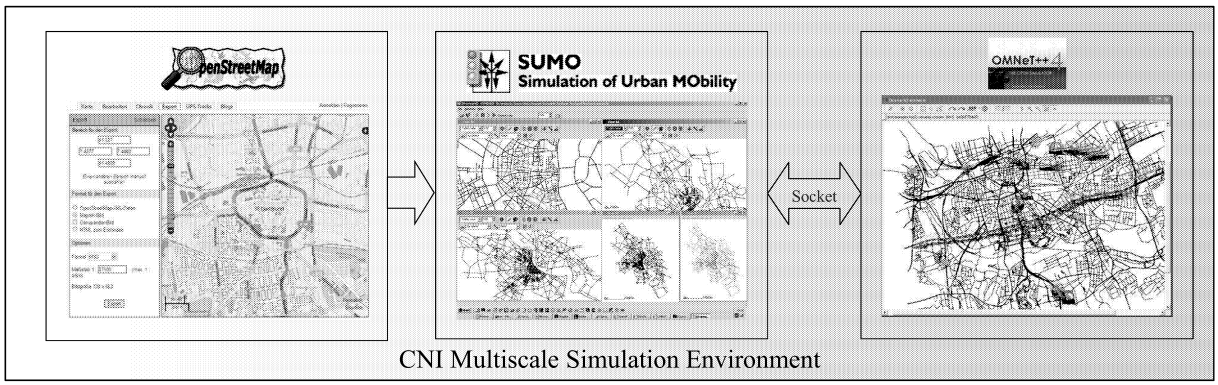


Figure 5: CNI Multicale Simulation Environment

characteristics and the movement of the mobiles lead to better results as an loop-detector only based simulation.

CONCLUSION AND OUTLOOK

As described in this paper, the combination of different simulation has the ability to increase the result's quality and accuracy. At first, a linear model is described as the beginning of the development. As a first improvement, two different kinds of junctions are designed.
These new elements in the simulation lead to the identification of additional information, which are necessary as starting values for the simulation (probability matrix). The actual information provider (loop-detectors) in the street are not suitable to provide the information. In a next step, radio network providers' data is identified a a possible source of the simulation input. It should be underlined, that only a static set of the data is needed from the provider is needed (and no active channel from the mobiles). After a short discussion of the problems with this kind of data, a new kind of simulation is introduced to generate the desired information out of the available data sets. In the end, a sample simulation setup is depicted to underline the powerful approach.
It could be shown that with the inclusion of static data in the simulation setup (probability matrix), the inaccurate live data (loop-detectors) could be enriched with details which lead to better overall simulation results.
As a next step it is planned to extend the working live simulation[1] with the statistical data to gain better results.

ACKNOWLEDGEMENT

Our work has been partially conducted within the SPIDER-project, which is part of the nationwide security research program funded by the German Federal Ministry of Education and Research (BMBF) (FKZ 13N10238, FKZ 13N10236). We thank all project members for their work and contributions to the SPIDER project.

[1] http://www.autobahn.nrw.de/

REFERENCES

Hafstein S.; Chrobok R.; Pottmeier A.; Schreckenberg M.; and Mazur F., 2004. *A High-Resolution Cellular Automata Traffic Simulation Model with Applicationin a Freeway Traffic Information System. Computer-Aided Civil and Infrastructure Engineering*, 19, 338–350.

Lewandowski A.; Burda R.; Subik S.; and Wietfeld C., 2008. *A Multiscale Simulation Environment for Performance Evaluation of high reliable heterogeneous Communication Networks.* In *European Simulation and Modelling Conference (ESM), Le Havre, France.* Eurosis, 131–136.

Nagel K. and Schreckenberg M., 1992. *A Cellular Automaton Model For Freeway Traffic. JPhysI France 2*, 2221–2229.

Schäfer R.P., 2009. *IQ routes and HD traffic: technology insights about tomtom's time-dynamic navigation concept.* In *ESEC/FSE '09: Proceedings of the the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT.* ACM, New York, NY, USA, 171–172.

Varga A. and Hornig R., 2008. *An overview of the OMNeT++ simulation environment.* In *Proc. of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (ICST).* Brussels, Belgium, 1–10.

Wegener A.; Piórkowski M.; Raya M.; Hellbrück H.; Fischer S.; and Hubaux J.P., 2008. *TraCI: An Interface for Coupling Road Traffic and Network Simulators.* In *Proc. of the 11th communications and networking simulation symposium (CNS).* New York, USA, 155–163.

A SIMULATION OF LOAD VARIABILITY OF A SYSTEM POWER STATION CAUSED BY A MICROGRID FED BY RENEWABLES

Eugeniusz M. Sroczan
State Higher Vocational School in Gniezno
ul. Ks. Kard. S. Wyszynskiego 38,
62-200 Gniezno, Poland
E-mail: eugeniusz.sroczan@put.poznan.pl

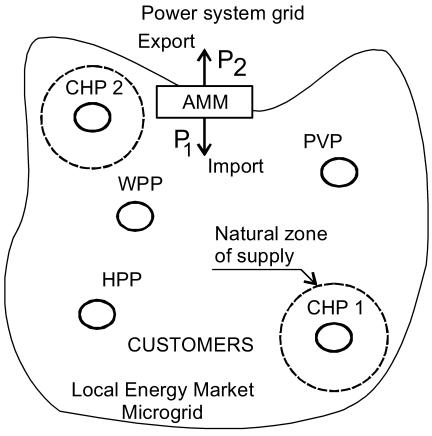
KEYWORDS
Energy, electrical engineering, decision support system, interactive simulation, optimization

ABSTRACT

Conventional grid systems require significant reserve capacity to manage the variability of demand, and the failure. A large power station as well as the microgrids supplied by renewable energy sources are committing to meet the power balance between generated and demanded power in the power system. In case of insufficient sun irradiation or wind speed, the electricity generation in photovoltaic and wind farms should be supported by thermal (classical and/or nuclear) power plants committing in the power system, which means that a microgrid is importing the energy. In the other case the energy is exported or stored.
The possibilities of covering the varied load by system power plants are curtailed. The presented paper tackles the technical and economical effects on the cost of electricity generation in mentioned manner by system power plants and power operation with the use of characteristics described for committing power plants. Inside of a microgrid demand electricity generation management (EGM) and demand side management (DSM) should be applied.
The results of simulation support the decision connected with the strategy of balancing the power flow among some microgrid with regard to plant operation in order to meet the power sytstem (PS) balance – between demanded load P_{dt} and generated power P_{gt} , in each *t* moment of time.

INTRODUCTION

The work of power plants, which generate electricity and are connected to microgrid is specific because of naturally limited resources of power and energy which results from renewable kinds of primary energy (PES) denoted to the given plant. The second characteristic is the limited number of group of energy consumers supplied by the microgrid and additionally the voltage level of the grid – usually low or medium at best. This kind of electricity genereation is also known as dispersed (or decentralized) generation (DG). The main idea of DG is to convert the primary energy in the place where it exists into electricity demanded by local energy market (LEM) as close to the final energy user as possible. Microgrid, in other words – LEM – is a part of power system (PS) and consists of a set of power plants - energy sources, grid and energy consumers.
The discussed local PS consist of the hydropower plants (HPP), photovoltaic power plants (PVPP) and wind power plants (WPP) committing, as well as standard thermal power plants (TPP) and combined heat and power plants (C-HPP) fired with gas, coal or biomass.
The connection of microgrids to the electricity utility (PS) network can be either synchronous or asynchronous. The synchronous is performed through an electrical connection and the asynchronous connection is carried out by a direct current coupled with a controlled power converter.
The essential problem of the local energy market (LEM) is to balance the power and energy production with consumers' demands, with regard to the technical and economic boundaries, as well as legal restrictions outlined by EC law in the area of the environment preservation. But the possibility of generating the electricity on demand may be and usually is strong curtailed by sun operation, wind velocity, weather conditions. Therefore inside of a microgrid the demand electricity generation management (EGM) and demand side management (DSM) must be applied.
The wide area of investigations include works describing power reliability assessment of transmission and distribution systems (Paska et al. 2010), as well as coordination the operation of wind power plant with conventional generating units (Karki et al. 2010).
A demand management system (DMS) operates as an interface between the microgrid and the outside energy network. The interaction between microgrid and the regional and national electricity system may be managed by an energy market (Fig. 1.) with the use of AMM (advanced metering management).



Figures 1: A Structure of Relations between PS and Microgrid as a Subsystem of PS.

The cost of electricity generation, in the given power system covering the demanded load during the defined time horizon, depends on the structure of committing power

plants connecting to the power network (LEM) and in the presented case the microgrid is the subsystem of PS.

SIMULATION OF COST OF ELECTRICITY GENERATION

Assuming that the structure of power plants in LEM is fixed and fulfills the requirements given by technical possibilities of managing of the renewable energy sources, the cost of electricity will depend on the load profile of energy customers and available options of converting the renewable energy into electricity.
The cost of electricity, defined for its end user include (Schweppe et al. 1988):

$$c = c_g + c_{TL} + c_{TC} \tag{1}$$

where: c_g –cost of generation, C_{TL} – marginal cost of transmissions losses, c_{TC} – cost of network constraints.
The decision of load proper for a particular plant is optimal if the costs will fulfill, in each time *t,* the following relationship (Sroczan 2009):

$$C_t \rightarrow \min\left\{\sum_{i=1}^{n} C_i\left(P_{gi}\right)\right\} \tag{2}$$

and

$$\sum_i P_{git} = P_{dt}$$

$$P_{i\min} \le P_{ig} \le P_{i\max} \qquad i \in \Omega$$

where: P_{gi}, P_{min}, P_{max}, P_d – respectively: generated, minimal and maximal power for i-th committing power plant, P_d – demanded power t – time range.
The simulated costs of electricity are calculated by using some methods depending on the primary energy used in the concerned power plants.
The value of power generated in hydropower plant is defined as:

$$P_{HPP} = 9{,}81 \cdot Q_t \cdot H_t \cdot \eta(Q_t, H_t) \quad [\text{kW}] \tag{3}$$

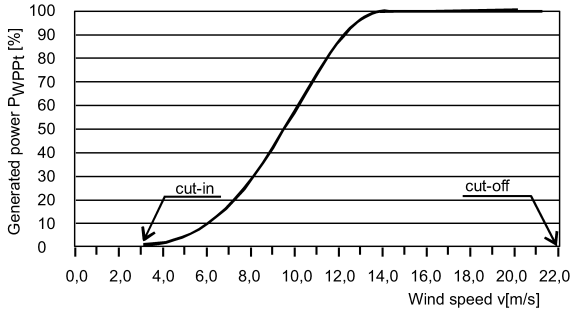
where: Q [m^3/s] – water flow, H [m] – head of water, $\eta(Q,H)$ – turbine efficiency coefficient.
Generated value of power of wind power plant depends on wind speed as follow:

$$P_{WTt} = \eta_T \cdot C_1 \cdot C_p\left(v_t\right) \cdot v_t^{\,3} \quad [\text{kW}] \tag{4}$$

where:

$$C_1 = \frac{1}{2}\rho \cdot \pi \frac{D^2}{4} \cdot 10^{-3}$$

and: η_T – total efficiency of generator and AC/AC converter, C_p [-] – efficiency coefficient, ρ_o – air mass [kg/m^3], D [m] – rotor diameter, v_t [m/s] – instantaneous wind speed.



Figures 2: Input-output Characteristic of Wind Power Plant

The value of generated power depends on input-output characteristic of wind turbine – imported from SCADA for each turbine and processedand value of efficiency coefficient depending on wind speed (fig. 3.)
Energetic characteristics of photovoltaic plant are updated with the use of correction coefficients for crucial parameters, which affects the efficiency factor of the power unit. The main are: sun irradiation and temperature of PV cells and cloud of sky. For purposes of load dispatch the corrected and guaranteed characteristics are calculated as:

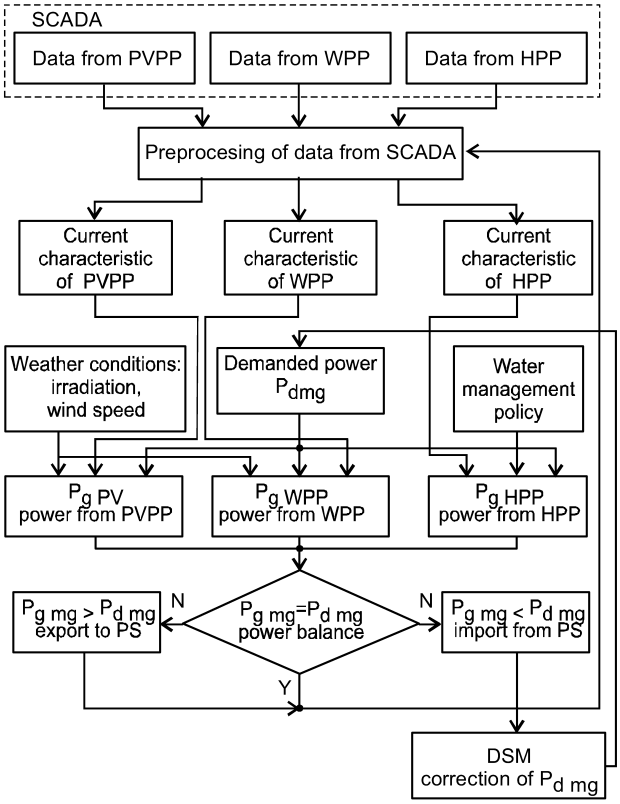
$$P_{PVt} = f\left[I_{IR}\left(dt, cl, \vartheta\right)\right] \quad [\text{kW}] \tag{5}$$

where: I_{IR} – irradiation of PV panels, coefficients: dt – daytime, cl – sky cloudy, ϑ – temperature of panels.

The optimization of the electric energy cost by using some procedures that fix the load of sources committed in the discussed PS are described also in the papers (Jasiński and Kaproń 2008) for CHP committing wit PS. Operation of power grid and wind farms is described in the paper (Kacejko and Pijarski 2009) and optimization processes of operation the hydropower and wind power plants are described, e.g., in papers (Karki et al. 2010, Sroczan 2010).

STRUCTURE OF SIMULATOR

Applied procedures simulate the costs of the generation and verify from the economic point of view the quality of balance of the generated and demanded electric power and the energy, inside of the microgrid, with regard to the input-output characteristics of considered kinds of power units.



Figures 3: The Structure of Simulator

In a case of shortage of power generated by photovoltaic and wind power plants, the power is balanced using power imported from PS. The relations between PS and microgrid as subsystem of PS are presented in the figure 1.

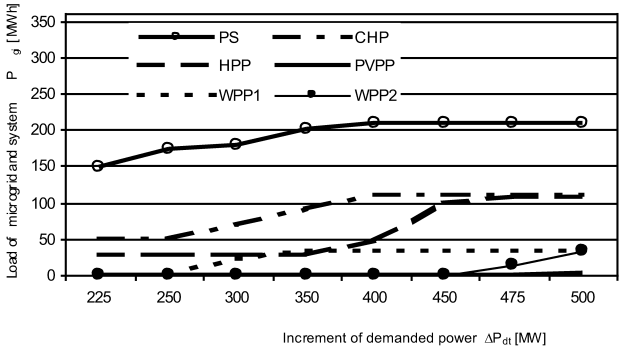
The structure of the developed simulator (fig. 3.) consists of:

- preprocessor of the data obtained from wind farm, with the use of SCADA (supervisory control and data acquisition), enabling the calculation of the range of generated power for each of wind turbines and PV cells;
- procedures for preparing the energetic characteristic of committing power plants with regard to simulation the dynamic properties of concerned power plants, supplied from RES and set of PS power plants;
- procedures applying the classical attempt to optimization as well as the fuzzy technique especially for assessment the possibilities of changing the covered load in the mode of cut-in and cut-off or shut down the renewable sources.

In a case when the power of the given source depends on wind speed or water inflow, the time schedule program of load must take into consideration some uncertainty of disposed volume of power or energy. The lack of power in that source node is compensated by additional flow from adjacent nodes or PS (fig. 1). This kind of possibilities depends on the structure of generation and allocation of power system reserve and flexibility of power grid.
The equations (1) and (2) are used to solve the problem of optimal scheduling the load of power plants described by characteristics (3), (4) and (5).
The results of simulation support the decision connected with the strategy of the power plant operation in order to meet the PS balance – between demanded load P_{dt} and generated power P_{gt}, in each *t* - moment of time.

RESULTS OF SIMULATION THE VARIABILITY OF PS LOAD

The main aim of this paper is to simulate the affect of the microgrid, as a structure of the electric power sources and subset of PS, and flexibility of system power plants in order to minimize the costs of energy with respect to renewable energy ratio (Paska et al 2010, Sroczan 2010) and possibilities of distributed operation of committing power plants.
The proposed attempt is based on the algorithm considering the real costs of energy in the given conditions, calculated with regard to demand and the updated characteristics of the renewable energy sources WPP, PVPP and HPP.



Figures 4: Load Schedule for Committing Power Plants

Finally the results define the policy for operating the set of power plants connected to microgrid and the given PS, with regard to data processed by SCADA systems (Sroczan 2010).
In the presented attempt the algorithm realized by AMM – the interface between microgrid and PS grid takes into account the disposed volume of renewable sources in discussed PS.
Analysis of cost of generated energy, realized with the use of developed simulator allows drawing of the following conclusion. The participation of power plants supplied with renewable energy sources causes the increase of the cost of generation in the discussed PS, due to additional costs stated by the rules of energy managing and these costs are transferred to the end users of energy.

CONCLUSIONS

The developed simulator supports the AMM in the optimal load dispatch of committing units with regard to the microgrid and dynamic properties of considered power plants, as well as varied in the time resources of sun irradiation, wind and water energy.
The process of optimisation of the work of committing power plants considers time period T in which all of the discussed units change the value of generated power with accordance to power demanded by end users. The goal is defined as minimization of generation cost with respect to the PES and PS constraints.

REFERENCES

Hunt V.D. 1982. "*Handbook of conservation and solar energy. Trends and perspectives*". Van Nostrand Reinhold Cop. Inc. New York 1982.

Jasiński P., Kaproń H. 2008. „Optimization of heat and power plant work in limit competition conditions". *Rynek Energii* No 2(75), 24-30.

Kacejko P., Pijarski P. 2009. „Connecting of wind farms – limitations instead of oversized investment". *Rynek Energii* No 1(80), 10-15.

Karki R., Po Hu, Billinton R. 2010. „Reliability evaluation considering wind and hydro power coordination". *IEEE Transaction on Power Systems*, Vol. 25 No 2, 685-693.

Paska J., Sałek M., Surma T. 2005 "Next Step in Application of Transmission and Distribution System Reliability Analysis". *Power Tech, 2005*, 27-30 June, IEEE Russia, p.1-7.

Sroczan E. M. 2009, The simulation of energetic characteristics of hydropower plants and wind farms applied to the cost analysis of electricity generation in the power system. In: Marwan Al-Akaidi (Ed.) *Modelling and Simulation '2009. The European Simulation and Modelling Conference 2009*. Publ. EUROSIS-ETI, Ghent Belgium, p. 181-183

Sroczan E. 2010. "Problems of integration of IT systems in dispersed systems of electricity generation." *Electrical Revue*, No 4, Year LXXXVI, 233-236.

Schweppe F.C, Caramanis M.C., Tabors R.D., Bohn R.E. 1998. *"Spot pricing of Electricity"*. Kluwer Academic Publishers, Norwell 1998,

EUGENIUSZ SROCZAN is employed as an assistant professor at the Poznan University of Technology (PUT) and professor of State Higher Vocational School in Gniezno. He obtained from PUT a M.Sc. and Engineering Degree in area of Industry Automatic and a Ph.D. in area of Electric Power System Engineering from PUT. Author of the book on contemporary electrical installations of the home. Since 2005 he has been the head of The Institute of Computing Science in State Higher Vocational School in Gniezno. and since 1984 he has been the President of Branch of Polish Electricians Society at the PUT.

LOAD DIFFUSION AND BROWNIAN MODELS FOR CLOUD BALANCING: BETWEEN C-S AND P2P

Vasil Georgiev

Faculty on Mathematics and Informatics
University of Sofia "St. Kliment Ochridski"
1164 Sofia, Bulgaria
v.georgiev@fmi.uni-sofia.bg

KEYWORDS

Cloud systems, load balancing, Markov chains, numerical approximation

ABSTRACT

In this paper we study several classes of dynamic load balancing schemes for cloud clusters (cloud balancing or CB for short). Taking client-server (C-S) and peer-to-peer (p2p) as the two boundary cases of the possible CB control schemes we present taxonomy of the intermediate ("diffusion" and "Brownian") schemes and develop two types of multi parameter models of these schemes. The first type consists of simulation models based on the queuing systems and the second type includes analytical models based on Markov chains. These two types of models allow studying of the various properties of the CB schemes and also can serve as a mean of mutual verification. Further in this paper a case study is presented comparing the analytical results for diffusion and Brownian CB scheme. Our plan for further extensive investigations on the full scale of CB concludes this paper.

I. CLOUD BALANCING FEATURES

The recent advances in service-centric architectures leaded to the adoption of the cloud paradigm for distributed server infrastructure. Beside the rest system functions like compound service interface non-functional subsystems are of great importance for keeping high level QoS in clouds. Their tasks are to improve application performance, ensure application availability, and implement a strategic disaster recovery plan. Cloud balancing (CB) extends those advantages by enabling organizations to leverage cloud deployments along with their local application delivery deployments (McVitte 2010, Amazon 2010). Thus the two major tasks of the load transfer between the cloud nodes are – performance improvement and fault tolerance.

Typical features of CB are the following.

The clouds typically work in the mode of ***data parallelism***. Always business-oriented, cloud computing is a cost-effective alternative to building additional data centers. Data centers perform asynchronous tasks. Cloud systems typically consist of one or more intentionally built datacenters, physically spread in strategic geographical locations over the world. For instance one of the biggest Cloud owner companies – Amazon, has exactly three datacenters at the time of present (Amazon 2010).

Cloud datacenters are usually built with the purpose of business servicing, and contain farms of hardware and equipment assembled internally in an optimized environment. These circumstances lead us to the idea to place one ***clusterization*** of Management Servers in each datacenter to handle the management of resources there. The cluster handles the organization of resources in the whole datacenter, sharing internally the load of management activities. Since we potentially have more than one datacenter, we would need additional inter-cluster layer to consolidate the results from all clusters into a centralized view to the Cloud system administrators (Zhelev and Georgiev 2010).

The clouds are mostly homogenous frameworks. ***Homogeneity*** here is in regard to the various levels of system protocols or services – IAAS, PAAS and SAAS – and it

- helps to apply compatible CB policies in each infrastructure entity incl. monitoring information and reallocation decisions;
- helps in the job transfer between the nodes with less concerns of portability.

An important feature of the cloud service infrastructure is ***dynamicity***. The cloud architecture is dynamically scaling up to accommodate growth or decline of the business oriented service process (Bahson 2010). A good example is the "elasticity" (i.e. other term for dynamicity reflecting "the increase or decrease capacity within minutes, not hours or days") of the Amazon cloud services (Amazon 2010).

Clouds operate on several (normally three) levels of manageable computing resources (e.g., networks, servers – for IAAS; storage, applications – for PAAS, and services – for SAAS-clouds). Thus ***layering*** of the system functions like CB help in building complex models for CB reflecting the load conditions and also helps in monitoring of the resource availability on these three levels.

Summarizing these features of CB we see that it is a type of load balancing allowing application of a widest range of balancing techniques and approaches in order to achieve a

[1] This research is part of the work on Projects 156&163/2010 of University of Sofia "St. Kliment Ochridski"

competitive QoS. As a result, the model space of CB is vast and its extensive study and comparison of the balancing schemes is of vital importance.

II. CLOUD BALANCING MODEL SPACE

According to the description in the previous section we assume that CB is performed in a cluster of homogeneous servers. The services' invocation process consists of asynchronous independent tasks and the actual node or resource to perform a task is specified dynamically and concurrently to the rest of the cloud servicing process.

Hence our CB model is ***clustered***. The balancing is performed in a intra-cluster manner. If an inter-cluster level is needed then the model may be upgraded ***hierarchically***. However intercloud (and thus an inter-cluster model level) seems to be not into the immediate agenda of technological standardization: "The intercloud is a lot like the set of all clouds connected via standards-based mechanisms. What those mechanisms are may be up for discussion and there are certainly groups devoted to defining those mechanisms but suffice to say that right now the "intercloud" does not exist. It (probably) will but we're a ways off from that" (McVitte 2009).

An important component of the model is the ***workload model*** and the complementary to it ***resource model***. For our considerations we adopt ***scalar models***: each task represents a statistically identical computing load entity whose complexity is presented by a scalar value of its abstract execution time by a dedicated server identical to those which are deployed in the cloud cluster. During the service period the various arriving tasks form the stochastic arrival process while their scalar complexities (i.e. service times) form the stochastic service (or departure) process. Correspondingly, the server's capacity to perform tasks is represented just and only by the time needed to execute a single task. We cannot (and need not to) prove that *most* of the load balancing models are based on scalar workload but surely lots of them share this feature (McVitte 2010, Triebes 2010, SwiftWater 2010, Cybenko 1994, Nathuji et al. 2010, Bahsoon 2010, Zhelev and Georgiev 2010). There are also ***vector*** workload/resource ***models***. In contrast to the scalar workload here the tasks (and consequently the performance capacity of the cloud cluster nodes) are associated with a [small] set of performance attributes (e.g., for tasks these are CPU consumption, memory consumption, and disk requirements, and for nodes the availability (or unavailability) of the CPU[-cores] in percents, free memory, disk space (Gong et al. 2009). At this stage we do not consider vector presentation of the workload model.

The whole CB process is split in three related but independent subprocesses, traditionally called ***control strategies***. These are the ***monitoring***, ***location*** and ***transfer*** strategies which are in certain relation – the output of each strategy is precondition or input for the next one in this order (i.e. M → L→ T). Their independence however reveals in the fact that each of the three strategies might be based on different principles. E.g. a centralized monitoring might be implemented on e server that gathers the local load and resource information and then this information is used by a distributed location strategy performed in each cluster node. Transfer strategies differ also in the ***invocation initiative*** which might be ***source-initiative***, ***receiver-initiative*** and ***symmetric initiative***. In (Eager et al. 1986) it is proven that transfer invocation initiative may have big impact on the overall performance and on the generated system communication overload. Additionally the receiver (and symmetric) initiative requires transfer of partially executed tasks. This requirement makes the receiver initiative not practically suitable especially for tasks with a large size runtime context. In CB we consider the sender-initiative transfers. One important specific feature of the monitoring strategy is the ***level of precision*** of collected system information. In CB a monitor (and consequently the locator[s] which decisions are based on this monitor) may only distinguish two states of each cluster node (or service): underloaded and overloaded – having one threshold parameter as a boundary between these two states. Less precision would be just the awareness if given node is in operation or down. More precise level of information would distinguish three states adding a neutral state between the under- and overloaded ones and having two thresholds. Obviously the level of precision can go further allowing more precise allocation decisions for the price of incurring bigger system overload. Furthermore the transfer strategy can be ***iterative*** or ***direct*** – according to the possibility for tasks re-transmission; direct transfers are appropriate for systems with centralized monitoring (Xu et al. 1994) because of the need of optimal or quasi-optimal matchmaking.

Major design issue in CB is the pattern of the ***control scheme***. It might be centralized, globally distributed or locally distributed. Note that each of these three control schemes can be applied independently to the three control strategies as mentioned above. By a ***centralized control scheme*** a cluster controller or server performs the CB system activity – monitoring and/or location and/or transfer. In fact it is a client-server model of distribution (***C-S***) by which the server monitors and manages the health and load distribution of the cloud-based deployment while the rest of processes and nodes are its clients regarding the CB system function. By ***globally distributed control scheme*** (which we associate with the peer-to-peer distribution model, ***p2p***) each cloud node monitors the global (cluster-wide) condition and/or negotiate possible task transfers with its peers. Again C-S CB normally associates with direct transfer schemes and p2p tends to favor iterative transfers. An utterly simplified form of p2p load negotiation is the balancing tasks transfer to a random peer node in case of local overload. Although such a location policy may seem too primitive we consider it as a boundary case of p2p CB control and refer to it as "***Brownian***" for its resemblance to the chaotic molecular movements in high-temperature fluids. With its lack of any monitoring and location considerations – and thus with the extreme reduction of the system overload – Brownian CB might well be an effective way for load sharing and we include it into our modeling considerations. ***Locally distributed control scheme*** (***Lp2p***) is again a p2p one but its scope is only focused on the limited set of nodes in the cloud. Each node has a limited number of neighboring peer nodes like if the system architecture is not a broadcasting one. We reuse the term "***diffusion***" for describing such CB schemes again analogously to the molecular dynamics by which the diffusion molecular movements are performed in local vicinity.

The inventive term "diffusion" in the context of the computation load management was introduced actually in (Cybenko 1994) where it denotes the load management in non-broadcasting distributed systems of given network topology (i.e. hypercube). In (Karageorgos et al. 2004) the same concept is generalized for an arbitrary (but again non-broadcasting) network topology presented by its adjacency matrix. However the concept of diffusion is well suitable for present time cloud clusters where the physical connectivity allows system-wide broadcasting but the logical connectivity might be limited for performance reasons. Thus the locality, the monitoring and decision scope of each node is limited to a degree of neighborhood peers according given pattern – line (1 neighboring peer), ring (2 neighbors), three n-dimensional lattice ($2n$ neighbors), hypercube (2^n neighbors) and so on, with a major parameter the number of logically connected peers v. We consider C-S and p2p as the two boundaries of a space of various locally distributed load (Lp2p) control schemes. By Lp2p each service has a limited number v of neighboring peers and performs the monitoring and task transfer functions in this limited scope. Obviously for p2p $v=p$-1 (where p is the cloud cluster size in number of nodes) and for a C-S v=1 (except of the controller itself).

The goal of this paper is to develop models for performance evaluation of the whole range of Lp2p cloud balancing schemes – from C-S to p2p – models which can be evaluated by means of computer simulations and also by analysis. By varying the workload conditions of these models not only the advantages of the two boundary control schemes could be compared but also the intermediate diffusion schemes of different level of logical connectivity v can be placed in a complete performance scale. Because of the complexity and price of the cloud infrastructures it is difficult to carry out experimentally such a study – at this stage we just want to compare two different types of models of described system processes. The plan of our research is to compare simulation models based on queuing systems and analytical study of Markov chains representing the behavior of each cloud cluster node in the cloud cluster. Here we describe a case study of a simple Lp2p CB scheme just to demonstrate how our model can be used to analyze the general class of CB problems.

III. GENERAL MODEL FOR CLOUD BALANCING

We consider a cloud cluster unifying several (or many) service multicore[1] nodes connected by a broadcasting network. The workload of this cluster consists of a stream of independent tasks which can be executed on any core and node. The ***cloud balancing task*** is to achieve shorter service time for the individual tasks by eliminating or reducing occurrences of temporarily idle cores (let alone whole nodes) while there are waiting tasks in other nodes. In p2p models tasks arrive randomly at any of the nodes. A variation of these models would be if tasks arrive at several interface nodes while the rest of the nodes work in background. In C-S models all the tasks arrive in a dedicated balancing server which is performing CB process. The balancing transfer is iterative (except for the C-S balancing) with a scalar presentation of the workload and always sender-initiation is presumed. By sender initiation, at the moment of arrival to an overloaded node, tasks are transferred to an underloaded neighboring node if available.. Our model is not limited to the scalar workloads and scalar resource parameterization but at this stage we exclude vector models just for simplicity.

By these conditions the ***input modeling parameters*** are as follows:

- cluster size: n nodes with c cores each; cloud homogeneity does not exclude the possibility of having nodes that differ in performance because of the different number of cores or because of different external load condition (e.g. interface nodes and background workers). In such case our model is still applicable but the analysis presented here in the case study should be carried out for each of the classes of similar nodes
- task arrival process with global (i.e. cluster-wide) rate Λ and local rate at the nodes λ – workload model presented by
  - centralized load scheme (i.e. C-S) by which a cluster controller monitors and manages the health of the cloud-based deployment
  - globally distributed load scheme – p2p, including Brownian and diffusion schemes
  - locally distributed load scheme with load interfacing nodes and background working nodes – p2p, including Brownian and diffusion schemes
- task execution/departure process with local rate of each core ω – i.e. the distribution of the time complexity of the consecutive tasks
- global load condition γ – the ratio between the rates of tasks arrival and departure process
- local load condition presented by the number l of tasks being executed in the node
- locality of the CB strategies – presented by the number of neighboring peers v and by the graph of their logical connectivity (or. the adjacency matrix)
- monitoring precision level – presented by the number of distinguishable load states and thresholds values. Again for simplicity we consider three distinguishable load states of the nodes – receiver ($\mathcal{R}$), neutral ($\mathcal{N}$) and sender ($\mathcal{S}$) of tasks - and thus two thresholds: T_1 is the upper boundary of the underloaded (receiver) state and T_2 is the lower boundary of the overloaded (sender) state. In order to define current load state, l is to be compared to T_i. Normally, $c \le T_1 \le T_2$. However in (Georgiev and Iliev 1997) we have studied and proved partially efficient one scheme by which $T_1 > T_2$; this inversion we called ***hysteresis***. Without hysteresis the intermediate state between the two thresholds is neither receiver nor sender of tasks and does not participate in balancing transfer – i.e. neutral. By hysteresis inversion of the thresholds, a node in the intermediate state (i.e. when $T_2 < l < T_1$) is both receiver and sender of tasks – "re-sender" ($\mathcal{RS}$). Although seemingly odd, the re-sender working mode may well prove to be effective for faster diffusion of the tasks (e.g. for schemes with a low degree of logical connectivity v). Each node informs its neighbors (v in count for diffusion, n for Brownian and 1 for C-S) for any transition in its state –

[1] i.e. the workstations are of the type of symmetric UMA multiprocessors as it is usual nowadays

$\mathcal{R}$, $\mathcal{S}$, $\mathcal{N}$ or alternatively $\mathcal{RS}$. As a result each node is informed for the current state of its neighbors and uses this information to allocate possible receiver of tasks.

The ***internal modeling parameters*** (***intermediate results***) are:

- local rate of receiving tasks from overloaded nodes ρ
- local rate of sending tasks to the receiving nodes σ
- effective local rate of tasks arrival in the node λ_ε
- effective local rate of tasks departure in the node μ_ε
- local load ratio ϕ
- mean number of tasks in a node £
- mean number of senders, receivers, neutrals or re-senders in the cluster – N_S, $N_\mathcal{R}$, $N_\mathcal{N}$ and $N_\mathcal{RS}$ respectively
- the associated probabilities for the nodes to be in that condition P_S, $P_\mathcal{R}$, $P_\mathcal{N}$ and $P_\mathcal{RS}$ additionally indexed by the class index if the cluster does not consist of homogeneously equipped and load nodes

The ***output parameters (results)*** of the evaluation of our model are:

- mean service time in cloud T
- mean rate of task transfer messages θ for each node
- mean rate of monitoring messages exchange process δ for a node
  - mean rate of system communication overload χ (task migration + monitoring messages for the cluster)
- mean local μ and global rate M of system departure
- mean idle time of nodes (including time of idle cores)
- optimal system size i.e. the bound of the speedup linearity/scalability

The CB modeling and analysis aims a better performance of the cloud infrastructure. From the user viewpoint this means shorter service time; for the viewpoint of a service provider (which in clouds includes infrastructure-providers too) effectiveness and resource utilization might not be less of concern. And from the viewpoint of the general community there might be other optimization goals which can be measured as "service price" – not necessarily in financial context: e.g. green (ecological) clouds (Bahsoon 2010). In case more than one parameter is of concern a vector workload/resource model is needed for the complex assessment of the service process. For example such vector may combine the service time with one or two types of "prices".

This CB model depicts a vast space of combinations and CB options. Extensive experimental study of this model hardly can be accomplished especially on a cloud scale. However it is straightforward to translate each of the presented model options into simulations based on queuing models which can give a complete picture of the performance vs. the load conditions and the allocated resources. We leave over such an exhaustive study and comparison between the options for the future work. Here we present a case study of analytical comparison of the diffusion and Brownian schemes and the possible effect of hysteresis.

IV. CASE STUDY: ANALYSIS OF DIFFUSION AND BROWNIAN CB SCHEMES

Let us consider the case of a homogeneous cloud cluster consisting of n nodes with c working cores each and connected in a high-speed broadcasting network. The task service process is a stochastic random Poisson process with mean rate of service in a single core server ω; the task arrival process is again a Poisson random process which is spread randomly between all the cluster nodes in sub-processes with a mean rate λ.

Case DF (***diffusion***). Each node has v ($v << n$) neighboring peers to which it sends the monitoring information about any distinguishable change of the local state (e.g. on transitions between – $\mathcal{R}$, $\mathcal{S}$, etc.) and to which it sends tasks when there are any receivers among the neighbors. In sender-initiated balancing the event of sending attempt is triggered by any arrival of a new task in a $\mathcal{S}$-node (and just for completeness only the newly arrived task are transferred to the potential receiver; the previously arrived tasks might have been already partially serviced which makes their migration without losses of the work done at least problematical if not impossible). Here we have distinguishing states' thresholds T_1 between $\mathcal{S}$ and $\mathcal{N}$ and T_2 between $\mathcal{N}$ and $\mathcal{R}$; $c \le T_1 < T_2$.

Case QB (***quasi-Brownian***). As case DF but $v = n$. It is *quasi*-Brownian because the selection is done only between $\mathcal{R}$-nodes.

Case DH (***diffusion with hysteresis***). As case DF but $c \le T_2 < T_1$ and $\mathcal{N}$ becomes $\mathcal{RS}$.

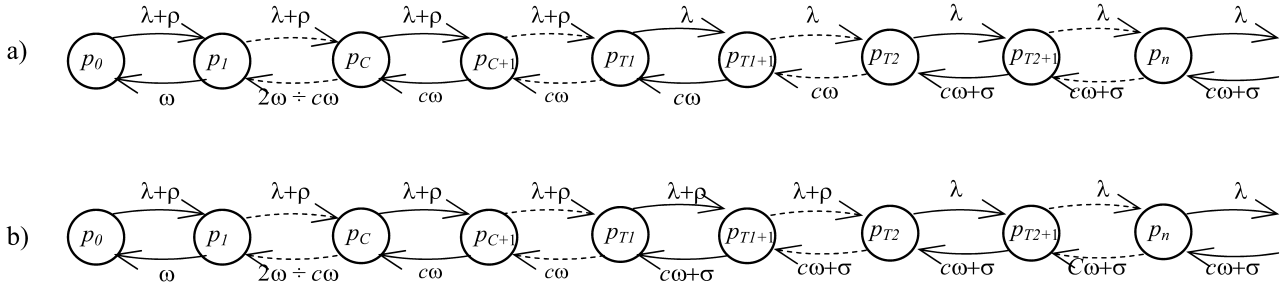


Figure 1. State-Transition-Rate Diagrams of a Node for Diffusion and Brownian CB – (a); and for Diffusion CB with Hysteresis – (b).

These cases as the rest of the model space can be easily translated to a queuing simulation model for CB performance evaluation. Here we describe ***Markov-chain based analysis*** of this model which is another possibility for its evaluation. The possible states of each node are presented by the number of tasks in service l that are further reduced to the three distinguishable and monitored states as l passes through the values of T_1 and T_2. The service process is described by the probabilities p_l of the node being in state l. The transitions between the node's states occur with a rate which is derived from the values λ, ω, ρ and σ. ***Figure 1.a)*** shows the state-transition-rate diagram of a random cloud cluster node for the cases DF and QB and ***1.b)*** presents the the state-transition-rate diagram for the case DH. Given the input parameters we have to find the output parameters (eventually passing thorough the intermediate results). Analyzing case DF we have the following system of equations (e1) ÷ (e10) for the steady-state probabilities p_l of each node:

$$(\lambda+\rho)p_i = (i+1)\omega p_{i+1}, \quad i \in [0, c] \quad \text{(e1)}$$
$$(\lambda+\rho)p_i = c\omega p_{i+1}, \quad i \in [c+1, T_1-1] \quad \text{(e2)}$$
$$\lambda p_i = c\omega p_{i+1}, \quad i \in [T_1, T_2-1] \quad \text{(e3)}$$
$$\lambda p_i = (c\omega+\sigma)p_{i+1}, \quad i \in [T_2, \infty) \quad \text{(e4)}$$
$$\Sigma p_i = 1, \quad i \in [0, \infty) \quad \text{(e5)}$$
$$\rho P_{\mathcal{R}} = \sigma P_{\mathcal{S}}, \quad \text{where} \quad \text{(e6)}$$
$$P_{\mathcal{R}} = \Sigma p_i, \quad i \in [0, T_1-1] \quad \text{(e7)}$$
$$P_{\mathcal{S}} = \Sigma p_i, \quad i \in [T_2+1, \infty) \quad \text{(e8)}$$
$$\sigma = \lambda P_{\Sigma\mathcal{R}}, \quad \text{where} \quad \text{(e9)}$$
$$P_{\Sigma\mathcal{R}} = 1-(1-P_{\mathcal{R}})^v. \quad \text{(e10)}$$

This system is complete to p_l from (e1) ÷ (e5) which are the equilibrium conditions of the state-transition-rate diagram in Figure 1.a); (e5) is the low for complete probability. The equations (e6) and (e9) are added in order to define the intermediate rates ρ and σ. The rest parameters are taken from the input block of the model. Particularly, (e6) with (e7) and (e8) reflect the fact that during the balancing process tasks neither enter nor leave the cloud cluster – they get only redistributed. Thus the overall rate of sent tasks is always equal to that of received tasks. In (e6) the factor n is removed from both sides of the equation; $P_{\mathcal{R}}$ is the probability for a node to be $\mathcal{R}$ and $P_{\mathcal{S}}$ is the probability for a node to be $\mathcal{S}$. The equation (e9) calculates the rate σ taking into account that the task sending occurs whenever a new task arrives with the rate λ at an $\mathcal{S}$-node and there is at least one $\mathcal{R}$-node among the neighbors the probability for which is $P_{\Sigma\mathcal{R}}$ in (e10). Here the calculation is based on v Bernoulli trials with failure probability $(1-P_{\mathcal{R}})$ for each.

The corrections of this system of equations for Brownian CB is straightforward and effect only (e10) where the factor v is to be replaced by n. Note that for these equations the system size n is not an input parameter except for the case QB where it replaces the factor of vicinity of the nodes. This is so because in our analysis up to now we consider just an isolated node which condition represents the homogeneous system as a whole. The derivation of the equations for case DH is straightforward in the same manner.

Having solved the system of equations, we have to derive the rest of the intermediate and output results. The mean number of tasks in a node is

$$£ = \Sigma i p_i, \quad i \in [0, \infty) \quad \text{(e11)}$$

The effective local rate of tasks arrival in the node is

$$\lambda_\varepsilon = (\lambda+\rho)P_{\mathcal{R}} + \lambda(1-P_{\mathcal{R}}). \quad \text{(e12)}$$

The effective local rate of tasks departure in the node μ_ε

$$\mu_\varepsilon = \Sigma(i+1)\omega p_{i+1} + \Sigma c\omega p_{j+1} + c\omega P_{\mathcal{N}} + (c\omega+\sigma)P_{\mathcal{S}} \text{ where}$$
$$i \in [0, c]; j \in [c+1, T_1-1]; P_{\mathcal{N}} = 1-P_{\mathcal{R}}-P_{\mathcal{S}}. \quad \text{(e13)}$$

The mean service time in cloud T (according Little's result) is

$$T = £/\lambda_\varepsilon. \quad \text{(e14)}$$

The local load ratio ϕ is

$$\phi = \lambda_\varepsilon / \mu_\varepsilon. \quad \text{(e15)}$$

The mean rate of task transfer messages θ for each node is

$$\theta = \sigma P_{\mathcal{S}}. \quad \text{(e16)}$$

The mean rate of monitoring messages' exchange process δ for a node is

$$\delta = (\lambda+\rho)p_{T1-1} + \lambda p_{T2} + c\omega p_{T1} + c\omega p_{T2}. \quad \text{(e17)}$$

The local mean rate of system communication overload is

$$\chi = \delta + \theta. \quad \text{(e18)}$$

The factor of idle state of nodes (including state of idle cores)

$$\Theta = \Sigma (1-i/c)p_i, \quad i \in [0, c] \quad \text{(e19)}$$

The mean local rate μ of system departure from the node is

$$\mu = \Sigma i\omega p_i + c\omega P_{\mathcal{R}} \text{where}$$
$$i \in [1, c]; \quad \text{(e20)}$$

and the mean global rate M of tasks departure from the cluster is

$$M = n\mu. \quad \text{(e21)}$$

Thus the global load ratio for the cluster is

$$\Phi = \Lambda/M. \quad \text{(e22)}$$

Now we can build the dependency of various output parameters of interest (e.g. T, χ, Θ) as a function of Φ and other model parameters like the thresholds, system size n, etc. Note that the system size which seemingly does not participate in the above formulas is nevertheless important system parameter as it is the factor by which the global load Λ is reduced to the local load rate λ and the factor by which the global load ratio is derived from the rates of the local departure processes.

Finally the important question for the optimal system size by given Φ i.e. the bound of the speedup linearity/scalability can be answered by evaluation of the system speedup

$$S_n = T(n)/T(1), \quad \text{(e23)}$$

where $T(n)$ and $T(1)$ are the mean service times for n- and one-node cloud cluster.

V. CASE STUDY RESULTS

We have done some preliminary evaluations of our cases DF model based on a numerical solution of the equations (e1) ÷ (e10). This system is not a linear one and the direct solution might be problematic. We propose a simplified numerical way for semi-approximation solution. First we derive from (e1) ÷ (e10) a sub-system of (T_2+1) linear equations for the unknown probabilities p_0 ÷ p_{T2} based on (e1) ÷ (e5). Note that the probabilities p_{T2+1} ÷ p_∞ are not needed for the solution of this

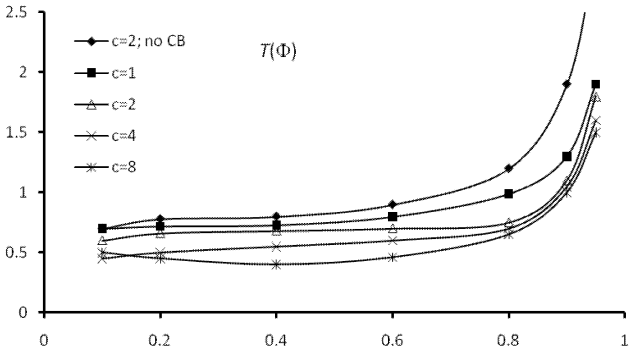


Figure 2. $T(\Phi)$ by Different Number of c.

system and their sum which participates in (e5) can be easily derived from the value of p_{T2}. The rest two unknowns σ and ρ receive approximation values iteratively until they satisfy the equilibrium in the additional conditions (e6) and (e9).

Our ***iteration procedure*** is defined as follows.

Step 0. Input values: n, c, Λ, $T_{1,2}$, Φ, step $k := 1$;
Derivates: $\lambda := \Lambda/n$, $\omega := \Lambda/nc\Phi$;
Approximations: $\rho^{[0]} := c\omega\Phi - \lambda$, $\sigma^{[0]} := \lambda/\Phi - c\omega$;
goto Step k.

Step k. Solve (e1) ÷ (e5) for $p_0 \div p_{T2}$ using $\sigma^{[k-1]}$ and $\rho^{[k-1]}$;
Calculate $\sigma^{[k]}$ and $\rho^{[k]}$ using (e6) and (e9);
If ($|\sigma^{[k]} - \sigma^{[k-1]}| < 1\%$ and $|\rho^{[k]} - \rho^{[k-1]}| < 1\%$) then goto End;
else k++ and goto Step k.

End.

By our numerical experiments this simple procedure puts $\sigma^{[k]}$ and $\rho^{[k]}$ within the 1% equilibrium boundaries in less than 10 iterations – usually less than 5. The staring approximations for $\rho^{[0]}$ and $\sigma^{[0]}$ we took from the ergodic conditions for the states $p_c \div p_{T1}$ and $p_{T2} \div p_\infty$ respectively. Yet by our experiments up to now we didn't find the iterative procedure to be very sensitive to the way initial values of $\rho^{[0]}$ and $\sigma^{[0]}$ are defined. Note also that during the consecutive iterations most of the intermediate results are probabilities having narrow numerical boundaries (0, 1) so they may serve as a warning if their iterative values do not fit in that interval.

As mentioned up to now we have done just a few numerical experiments with our model. First the basic dependence of the mean system service time T of tasks is presented as a function

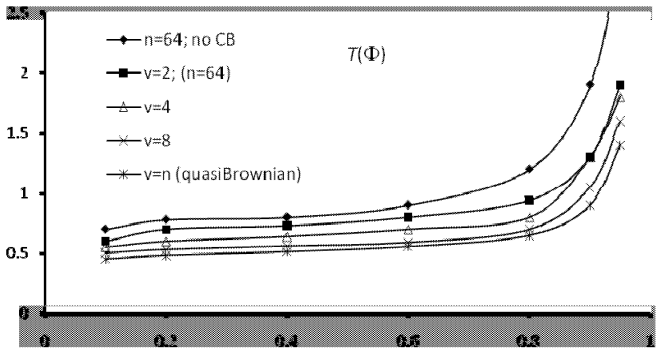


Figure 4. $T(\Phi)$ by Different Number of v.

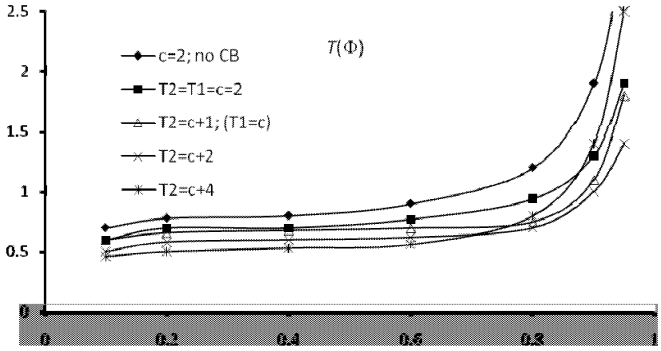


Figure 3. $T(\Phi)$ by Different $T_{1,2}$.

of the global load factor Φ. The advantage of the proposed CB scheme is obvious especially in the cases when Φ rises over 80%. The model favors systems with bigger number c of cores in the nodes (at least by conditions we have studied) – ***Figure 2***.

The subsequent numerical experiments show the impact of the values of balancing thresholds T_1 and T_2. We have fixed T_1=c in order to keep the nodes in $\mathcal{R}$-state just while there is at least one idle core. Normally T_1 shouldn't be much bigger than c. As ***Figure 3***. shows higher T_2-values are better for clusters with small to moderate load factor but do not perform well for heavily loaded clusters.

Further we analyzed the impact of the vicinity factor v on the performance starting from just two "neighboring" nodes and finishing to so called quasi-Brownian case with v=n on ***Figure 4***. Note that we haven't yet incorporated the possible dependence that might exist between the nodes' capacity to perform tasks locally and to do balancing transfers meanwhile. We have commented this subject later in the Conclusion.

Finally we have calculated the rate χ at which both task transferring and local load information messages are generated from a node to its neighbors. Obviously the bigger number of neighbors incurs higher exchange rate – ***Figure 5***. This dependence is even deeper for moderate to higher (but not highest) level of system load when predictably the node passes between the load states $\mathcal{R}$, $\mathcal{N}$ and $\mathcal{S}$ more frequently. We haven't checked yet how a hysteresis threshold policy will impact on χ. For this purpose we have to analyze the model case DH from Figure 1b).

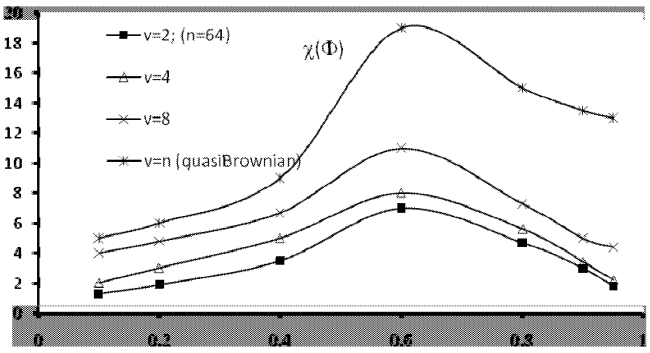


Figure 5. χ(Φ) by Different Number of v.

VI. CONCLUSION

We have introduced a model of cloud balancing and a method for numerical solution of this model for a couple of abstract cases of cloud clusters. In fact these cases only illustrate the way of analysis of our model and are far from complete study of the system behavior for various types of cloud clusters and workload conditions. For now we have skipped the intriguing question of how the hysteresis will impact the performance. The problem space which can be represented and analyzed by this model is far more large. The modeling results are to be validated by comparison to simulations and experiments in real systems.

In the context of the presented models we plan to investigate further the service qualities for several other workload and system parameters such as granularity and efficiency. To study ***granularity*** we can vary the mean task complexity by keeping Φ and rising Λ and ω. This way we model a workload of finer granularity, and vice versa a coarse granularity corresponds to lower values of Λ and ω for same Φ. ***Speedup*** and efficiency of given architecture can be modeled on the basis of (e23) by analysis of the impact that the system size n, the node "size" c and n:c ratio have on the performance for fixed system load ratio Φ. The impact $T_{1,2}$ and v have on the system performance rises the question of developing ***adaptation*** policies for these and others system parameters so that they can reflect dynamically the local load conditions.

Furthermore depending on the validation results we plan to extend this model by introducing vectorization and by correlating communication and computation performance in the cloud cluster. ***Vectorization*** will allow analyzing full scale QoS parameters like service price, "green price" (e.g. energy consumption), etc. Considering the performance impact of the system communication overload and the possible correlation between ω and the balancing-related communication rates ρ, σ and χ is another way to increase the adequacy of our model. This possible ***correlation*** can be introduced in two aspects. On local level we have to investigate how in a real service node the local service performance ω depends on the local communications represented by communication rates ρ, σ and χ. On a cluster scale we have to investigate the correlation between these communication rates having in mind that the internodes connections are implemented over a broadcasting network instead in a p2p one. The results in Figure 5. show the big variance of one of these rates which has to be taken into account for more realistic modeling of the system communication overload. Introducing ***heterogeneity*** and ***"topology"*** into the model will extend its scope to cloud systems in which there are several classes of nodes with various service parameters and also nodes that are specialized in different functions (e.g. interface nodes, worker nodes, balancing servers, etc.). For this purpose instead of the presented herewith modeling via definition of single node parameters for the whole system, we have to perform analysis on several classes of similar nodes and elaborate on their model parameters jointly.

REFERENCES

Bahsoon R. 2010. Green Cloud: Towards a Framework for Dynamic Self-Optimization of Power and Dependability Requirements in Cloud Architectures. *In Proceedings of the ACM/IEEE 32nd International Conference on Software Engineering*, 2-8. May 2010, Cape Town, South Africa.

Bahsoon, R. 2010. Towards a Framework for Dynamic Self-Optimization of Power and Dependability Requirements in Cloud Architectures. To appear in *Proceedings of the 4th European Conference on Software Architecture (ECSA 2010)*, Copenhagen, Denmark. LNCS, Springer.

Cybenko G., 1994. Dynamic Load Balancing for Distributed Memory Multiprocessors. *Journal of Paralel and Distributed Computing,* 7, 1989, pp. 279 – 301.

Eager, D., E. Lazovska, J. Zahorian. 1986. A Comparison of Receiver-Initiated and Sender-Initiated Adaptive Load Sharing. *Performance Evaluation*, Vol. 6., pp. 53-68, 1986.

Georgiev, V., M. Iliev. 1997. A Hybrid Scheme for Load Balancing in Distributed Systems. *Proceedings of the 11th European Simulation Multiconference ESM'97*. Istanbul, Turkey, June 1-4, 1997. pp. 521 - 528.

Karagiorgos, G., N. Missirli and F. Tzaferis. 2004. The generalized diffusion method for the load balancing problem. *Advances in Parallel Computing, Volume 13, Parallel Computing - Software Technology, Algorithms, Architectures and Applications.* Elsevier B.V. 2004, Pages 225-232

Nathuji R., A. Kansal and A. Ghaffarkhah. 2010. Q-Clouds: Managing Performance Interference Effects for QoS-Aware Clouds. *Proceedings of the 5th European conference on Computer systems*, April 13–16, 2010, Paris, France. pp. 237-250.

Xu C.-Z., F. C.M. Lau. 1994. Iterative Dynamic Load Balancing in Multicomputers. *Journal of Operational Research Society, v. 45, No 7*, 1994, pp. 786 – 796.

Zhelev, R., V. Georgiev. 2010. Generic Resource Framework for Cloud Systems. To appear in *the Proceedings of 5th International Conference Distributed Computing and Grid Technologies in Science and Education, GRID'2010*, 27. June - 04. July 2010, Dubna, Russia.

Zhelev, R., V. Georgiev. 2010. Load Balanced Resource Management for Cloud Systems. To appear in the *Proceedings of 4th International Conference on Information Systems and Grid Technologies, ISGT'2010*, 28-29. May 2010, Sofia, Bulgaria.

WEB REFERENCES

Adler, B. 2010. Load Balancing in the Cloud: Tools, Tips, and Techniques. *A technical white paper for Right Scale*. (www.rightscale.com/pdf/Load-Balancing-in-the-Cloud.pdf)

Amazon. 2010. Amazon Elastic Compute Cloud (Amazon EC2) (http://aws.amazon.com/ec2/)

Citrix Systems. 2010. NetScaler. Is your load balancer cloud ready? *A white paper for Citrix Systems, Inc*. (http://www.citrix.com/English/ps2/products/documents_onecat.asp?contentid=21679&cid=White+Papers)

Gong, Z.; P.Ramaswamy; X. Gu; and X. Ma. 2009. SigLM: Signature-Driven Load Management for Cloud Computing Infrastructures. *17th International Workshop on Quality of Service, IWQoS 13-15 July 2009*. (http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp? punumber=5201378)

MacVittie, L. 2009. Cloud Balancing, Cloud Bursting, and Intercloud. (http://devcentral.f5.com/weblogs/macvittie/archive/2009/07/09/cloud-balancing-cloud-bursting-and-intercloud.aspx)

MacVittie, L. 2010. Cloud Balancing: The Evolution of Global Server Load Balancing. *F5 White Paper*, 2010. (www.f5.com/pdf/white-papers/cloud-balancing-wp.pdf)

SwiftWater. 2010. Cloud computing, load balancing, and extending the data center into a cloud. (http://vburke.wordpress.com/2010/03 /26/cloud-computing-load-balancing-and-extending-the-data-center-into-a-cloud/)

Triebes K. 2010. Cloud Balancing: The Next Generation of Global Server Load Balancing. *Virtualization Review* (http://virtualization review.com/blogs/app-delivery-ondemand/2010/06/cloud-balancing.aspx)

ELECTRONICS SIMULATION

FIRST-PRINCIPLES MODELING OF BIPOLAR RESISTIVE SWITCHING IN METAL-OXIDE BASED MEMORY

Alexander Makarov, Josef Weinbub, Viktor Sverdlov, and Siegfried Selberherr

Institute for Microelectronics
Technische Universität Wien
Gusshausstrasse 27–29
1040 Vienna, Austria
Email: {makarov|weinbub|sverdlov|selberherr}@iue.tuwien.ac.at

KEYWORDS
RRAM, resistive switching mechanism, stochastic model, Monte Carlo method

ABSTRACT

A microscopic model of the resistive switching mechanism in bipolar metal-oxide based resistive random access memory (RRAM) is presented. The distribution of electron occupation probabilities obtained is in agreement with previous work. In particular, a low occupation region is formed near the cathode. A hysteresis cycle of RRAM switching simulated with the model including the ion dynamics is in good agreement with experimental results.

INTRODUCTION

The resistive switching phenomenon is observed in different types of insulators, such as metal oxides, perovskite oxides, and chalcogenide materials. Because the electrical conductance of the insulator can be set at different levels by the application of an electric field, this phenomenon becomes attractive for advanced memory concepts. Indeed, a state with high resistance can be interpreted as logical 1 and a state with low resistance as logical 0, or vice versa, depending on the technology. The concepts of memory using the resistive switching phenomenon can be conveniently divided into the following three categories: Conductive Bridge RAM (CBRAM), Phase Change RAM (PCRAM), and Resistive RAM (RRAM). CBRAM is based on a solid-state electrolyte in which mobile metal ions create a conductive bridge between the two electrodes under the influence of an electric field. PCRAM employs the difference in resistivity between the crystalline and amorphous phases of a chalcogenide compound. RRAM is based on metal oxides, such as TiO_x (Kugeler et al. 2008), HfO_2 (Chen et al. 2009), Cu_xO (Dong et al. 2007), NiO (Seo et al. 2005), ZnO (Lee et al. 2009) and perovskite oxides, such as doped $SrTiO_3$ (Watanabe et al. 2001), doped $SrZrO_3$ (Lin et al. 2007), $Pr_{1-x}Ca_xMnO_3$ (Sawa et al. 2004), and employs the electric field induced difference in resistivity between the high and low current carrying states.

The increasing demand for miniaturization of microelectronic devices has significantly accelerated the search for new concepts of nonvolatile memory during the last few years. Memory based on charge storage (such as flash memory, and others) is gradually approaching the physical limits of scalability, and conceptually new types of memories based on a different storage principle are gaining momentum. Apart from good scalability, a new type of memory must also exhibit low operating voltages, low power consumption, high operation speed, long retention time, high endurance, and simple structure (Lee et al. 2010 ; Kryder and Kim 2009).

In addition to RRAM, PCRAM, and CBRAM there exist several other concepts as potential replacements of the charge memory. Some of the technologies are already available in prototype form (such as carbon nanotube RAM (NRAM)), others as product (magnetoresistive RAM (MRAM), ferroelectric RAM (FRAM)), while the technologies of spin-torque transfer RAM (STTRAM) and racetrack memory are under intensive research.

From these concepts the CBRAM, PCRAM, and RRAM possess the simplest metal-insulator-metal (MIM) structure and, as a consequence, have good scalability. This fact gives advantages to RRAM, PCRAM, and CBRAM over other advanced memory concepts.

In addition to its simple structure RRAM is characterized by a low operating voltage (< 2 V), fast switching time (< 10 ns), high density, and long retention time.

Several physical mechanisms based on either electron or ion switching have been recently suggested in the literature: a model based on trapping of charge carriers (Fujii et al. 2005), electrochemical migration of oxygen vacancies (Nian et al. 2007; Wu et al. 2008), electrochemical migration of oxygen ions (Szot et al. 2006; Nishi et al. 2008), a unified physical model (Gao et al. 2009), a domain model (Rozenberg et al. 2004), a filament anodization model (Kinoshita et al. 2006), a thermal dissolution model (Russo et al. 2007), a two-variable

resistor model (Kim and Choi 2009), and others. Despite these efforts, however, a proper fundamental understanding of the RRAM switching mechanism is still missing, hindering further development of this type of memory.
We propose a stochastic model of the resistive switching mechanism based on electron hopping between the oxygen vacancies along the conductive filament in an oxide-layer, where a redox reaction plays a crucial role in the resistive switching from the state with low resistance to the state with high resistance and back.

MODEL DESCRIPTION

We associate the resistive switching behavior in the oxide-based memory with the formation and rupture of a conductive filament (CF). The CF is formed by localized oxygen vacancies (V_o) (Gao et al. 2009) or domains of V_o. Formation and rupture of a CF is due to a redox reaction in the oxide layer under a voltage bias. The conduction is due to electron hopping between these V_o. (Fig. 1)
For modeling the resistive switching in oxide-based memory by Monte Carlo techniques, we describe the dynamics of oxygen ions (O^{2-}) and electrons in an oxide layer as follows:

- formation of V_o by O^{2-} moving to an interstitial position;
- annihilation of V_o by moving O^{2-} to V_o;
- an electron hop on V_o from an electrode;
- an electron hop out from V_o to an electrode;
- an electron hop between two V_o.

In order to model the dependence of transport on the applied voltage and temperature we choose the hopping rates as (Sverdlov et al. 2001):

$$\Gamma_{nm} = A_e \cdot \frac{dE}{1-\exp(-dE/T)} \cdot \exp(-R_{nm}/a). \quad (1)$$

Here, A_e is a coefficient, $dE = E_n - E_m$ is the difference between the energies of an electron positioned at the sites n and m, R_{nm} is the hopping distance, a is the localization radius. The hopping rates between an electrode (0 or $N+1$) and an oxygen vacancy m are described by:

$$\Gamma_m^{iC} = 2 \cdot \alpha \cdot f \cdot \Gamma_{0m}, \Gamma_m^{oC} = 2 \cdot \alpha \cdot (1-f) \cdot \Gamma_{m0}, \quad (2)$$

$$\Gamma_m^{iA} = 2 \cdot \beta \cdot f \cdot \Gamma_{(N+1)m}, \Gamma_m^{oA} = 2 \cdot \beta \cdot (1-f) \cdot \Gamma_{m(N+1)}. \quad (3)$$

Here, f is the electrode occupation probability, α and β are the coefficients of the boundary conditions on the cathode and anode, respectively, N is the number of sites, A and C stand for cathode and anode, and i and o for hopping on the site and out from the site, respectively.

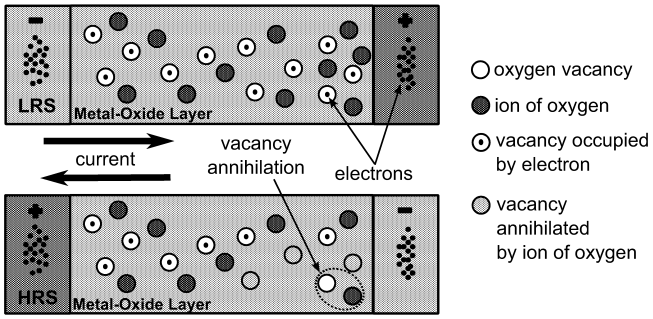


Figure 1: A schematic illustration of the conducting filament in the low resistance state (top) and the high resistance state (bottom).

To describe the motion of ions we have chosen the ion rates similar to (1):

$$\Gamma'_n = A_i \cdot \frac{dE}{1-\exp(-dE/T)}. \quad (4)$$

Here we assume that O^{2-} can only move to the nearest interstitial, and a distance-dependent term is thus included in A_i. dE includes the formation energy for the n^{th} V_o /annihilation energy of the n^{th} V_o, when O^{2-} is moving to an interstitial or back to V_o, respectively.
The electron current generated by hopping is calculated as:

$$I = q_e \cdot \sum dx / \sum \left(1 / \sum_m \Gamma_m \right). \quad (5)$$

SIMULATION TOOL

For modeling the RRAM switching behavior a simulation tool was developed which allows simulating 1D/2D/3D model systems. C++ was chosen as programming language. Fig. 2 shows a flow chart of the simulation process performed by the tool.
The module "starter" is a basic module which allows choosing different modes of simulation to produce results for a particular experiment of interest.
In the module "random generator" a random number generator using the l'Ecuyer algorithm (Press et al. 1992) was implemented. This algorithm allows random number generation with a practically infinite period ($\sim 2\text{x}10^{18}$).
The dimension (1D/2D/3D), size, site location, site energies, and other parameters describing the structure under the simulation are defined in the module "description of elements".
The input parameters are set in an initialization file as shown:

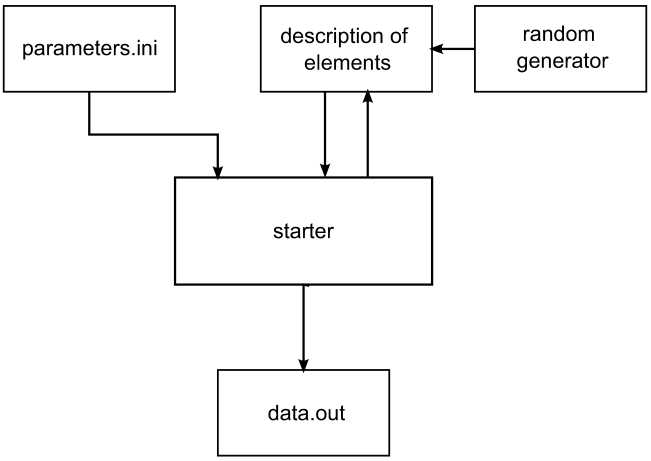


Figure 2: Basic schema of modules of the simulation tool.

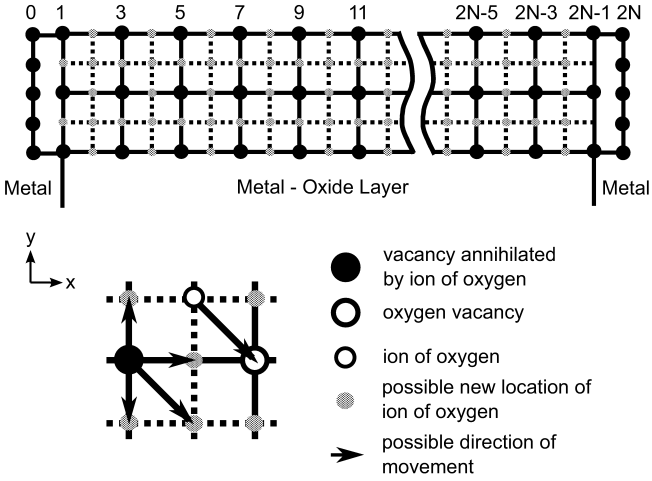


Figure 3: A schematic picture of the unit cell of the model system.

```
[Experiment]
type=2
n=1
[Lattice]
x=30
y=10
z=1
def=0
[RedoxEnergy]
E0=0.036
E1=5.0
E2=0.018
E3=0.036
E4=5.0
E5=0.018
[Stochastic]
Ae=100000
a=2
Ai=100
eR=25
iR=1
[Environmental]
T=0.025
[Set]
U=1.3
t=0.0
alpha=0.1
beta=0.1
f=0.5
[Reset]
U=-1.0
t=0.0
alpha=0.1
beta=0.1
f=0.5
[Change]
dU=0.01
dt1=0.01
dt2=0.001
dt3=0.01
dt4=0.001
```

Fig.3 demonstrates an example of a unit cell for a two-dimensional array of sites. The columns with number 0 and 2N are reserved for the electrodes (anode and cathode). By moving O^{2-} from a lattice site to an interstitial a vacancy V_o at the lattice site is formed. In the first moment of time we assume that there are no vacancies V_o. Each O^{2-} has a probability Γ'_n of moving to the nearest interstitial position (if this position is empty) making a formation of a new V_o possible; moreover, each O^{2-} has a probability Γ'_n of annihilation with the nearest V_o if this V_o is not occupied by an electron. In addition, the electron dynamics according to (1-3) on the vacancies V_o already formed must also be taken into account giving rise to the electron current in the system.

MODEL VERIFICATION

All calculations are made on one or/and two-dimensional lattices, the distances between two nearest neighboring V_o in all directions are equal. All V_o are at the same energy level, if no voltage or temperature is applied. Despite the fact that in the binary metal oxides V_o can have three different charge states with charge 0, +1, +2 (Schmidt-Mende and MacManus-Driscoll 2007), to simplify the calculations, we assume that the V_o is either empty or occupied by one electron. This assumption is not a limitation, however, due to the energy separation between the three charge states only two of them will be relevant for hopping and significantly contribute to transport.

Calculation of electron occupation probability

In order to verify the proposed model and our implementation in a simulation tool, we first evaluate the average electron occupations of hopping sites under different conditions. For comparison with previous works all calculations in this subsection are made on a one-dimensional lattice consisting of thirty equivalent, equidistantly positioned hopping sites V_o.

To implement all the above conditions, we used a model system with the artificially inflated value for annihilation energy of V_o, low value of formation energy for V_o, and large A_i. The low value of formation energy for V_o is used to guarantee that all O^{2-} are moved to the interstitial positions. An increased value for the annihilation energy of V_o is used guaranteeing that the already formed V_o will not be annihilated by O^{2-}.

Large values A_i are needed for accelerating the process of the CF formation. In order to compare our results with (Derrida 1998), we use for the occupation probability $f = 0.5$.

Following (Derrida 1998), we first allow hopping in one direction and only to/from the closest V_o (asymmetric single exclusion process).

Each site i of a one-dimensional lattice of N sites is either occupied by an electron or empty; during a time interval dt, each electron has a probability Γ_{nm} of hopping to the right, provided the target site is empty; moreover, during the time interval dt, an electron may enter the lattice at the Site 1 with the probability $\alpha \cdot \Gamma_{01}$ (if this site is empty) and an electron at the Site N may leave the lattice with the probability $\beta \cdot \Gamma_{N(N+1)}$ (if this site is occupied). The occupation probability p_c of a central V_o (p_c) is described, depending on the boundary conditions, as follows: 1) for $\alpha > 0.5$ and $\beta > 0.5$, $p_c = 0.5$; 2) for $\alpha < 0.5$ and $\alpha < \beta$, $p_c = \alpha$ 3) for $\beta < 0.5$ and $\beta < \alpha$, $p_c = 1 - \beta$. Fig.4 shows simulation results of the stochastic model, which fully obey the theoretical calculations (Derrida 1998).

To move from a model system of an asymmetric single exclusion process (Derrida 1998) to a more realistic structure, we have demonstrated the dependence of electron occupation probabilities from the position of the Fermi level in the electrodes relative to the energy level of the sites, determined by the value of f. Fig. 5 shows the result of our simulations.

We have further calibrated the model in a manner to reproduce the results reported in (Gao et al. 2009), for $V = 0.6$V to $V = 1.4$V. Fig.6 shows a case, when the hopping rate between the two V_o is larger than the rate between the electrodes and V_o (i.e. $\alpha, \beta < 1$). In this case a low occupation region is formed near the cathode (bipolar behavior).

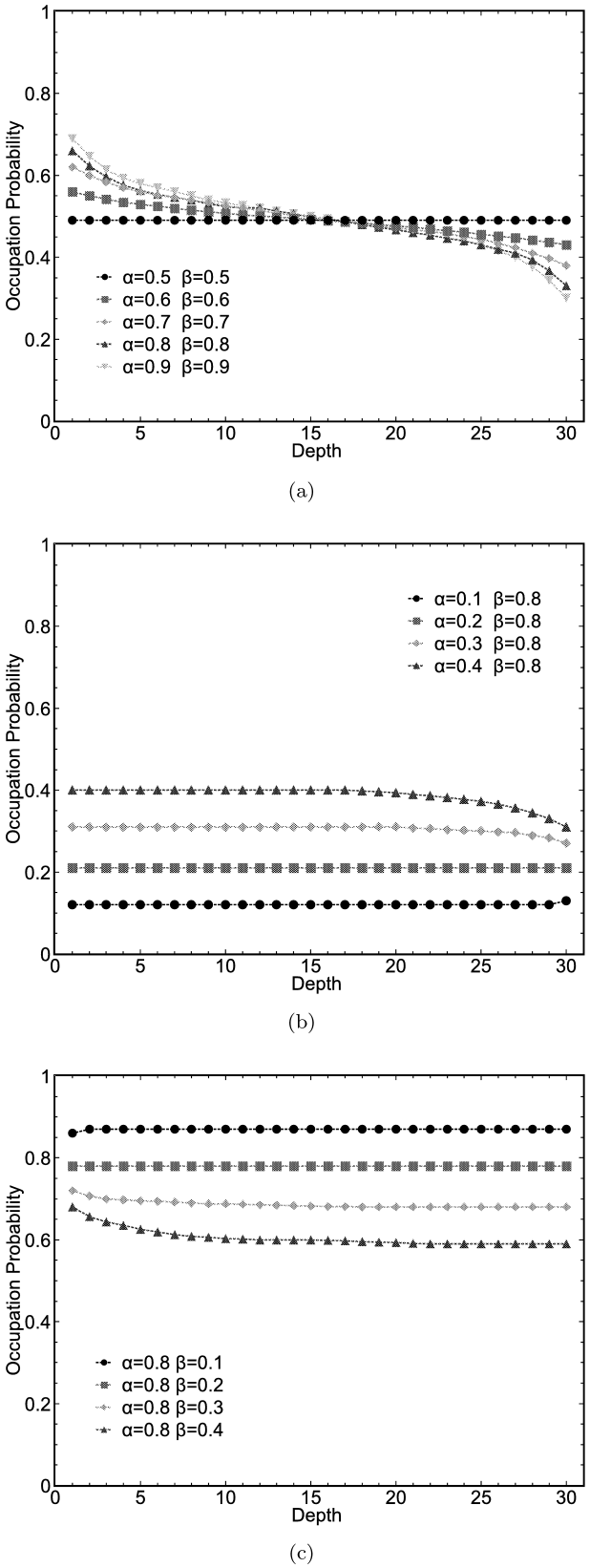
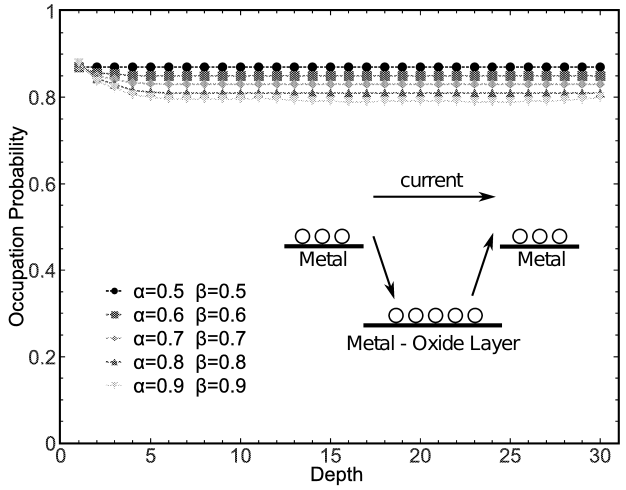
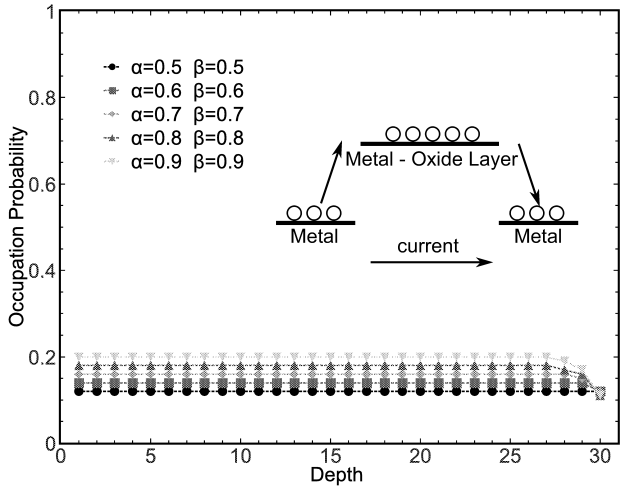


Figure 4: Calculated distribution of electron occupation probabilities for unidirectional next nearest neighbor hopping between the V_o (the 1^{st} V_o is near the cathode, the last V_o is near the anode): (a) $\alpha > 0.5$ and $\beta > 0.5$, $p_c = 0.5$; (b) $\alpha < 0.5$ and $\alpha < \beta$, $p_c = \alpha$; (c) $\beta < 0.5$ and $\beta < \alpha$, $p_c = 1 - \beta$.



(a)



(b)

Figure 5: Calculated distribution of electron occupation probabilities for unidirectional next nearest neighbor hopping between the V_o $\alpha > 0.5$ and $\beta > 0.5$: (a) $f = 0.9$; (b) $f = 0.1$.

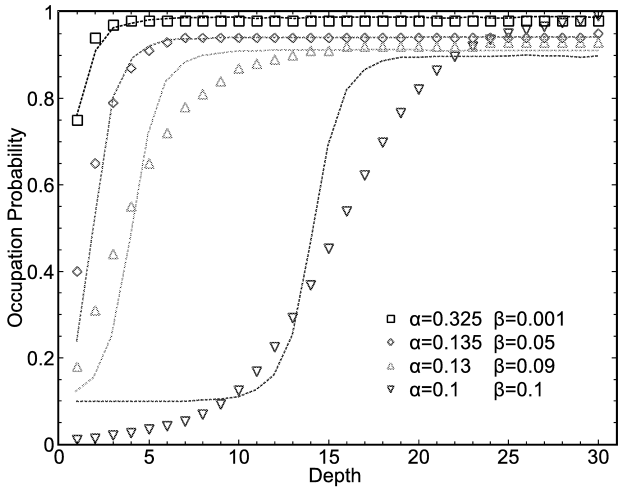


Figure 6: Calculated distribution of electron occupation probabilities under different biasing voltages. Lines are from (Gao et al. 2009), symbols are obtained from our model.

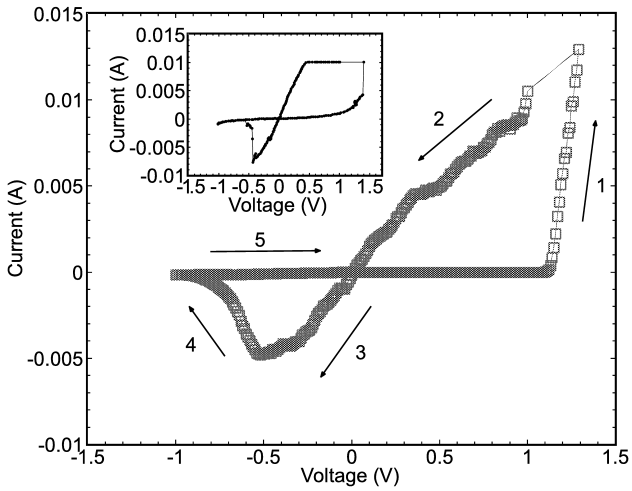


Figure 7: I-V characteristics showing the hysteresis cycle obtained from our model ($\alpha = 0.1$ and $\beta = 0.1$). The inset shows the hysteresis cycle for M-ZnO-M from (Lee et al. 2009).

Modeling of the hysteresis cycle

All calculations of the RRAM $I - V$ characteristics are now performed on a two-dimensional lattice (10x30). We have investigated the $I - V$ hysteresis by applying a saw-tooth-like voltage V. We have assumed that the coefficients α and β of the boundary conditions are constant and equal to 0.1.

The simulated RRAM switching hysteresis cycle is shown in Fig. 7. The simulated cycle is in good agreement with the experimental cycle from (Lee et al. 2009) shown in the inset to Fig. 7.

The interpretation of the RRAM hysteresis cycle obtained from the microscopic model is as follows. If a positive voltage is applied, the formation of a CF begins, when the voltage reaches a critical value sufficient to create V_o by moving O^{2-} to an interstitial position. The formation of the CF leads to a sharp increase in the current (Fig. 7, Segment 1) signifying a transition to a state with low resistance. When a reverse negative voltage is applied, the current flows in a different direction and first increases linearly in voltage (Fig. 7, Segment 3), until the applied voltage reaches the value at which an annihilation of V_o is triggered by means of moving O^{2-} to V_o. The CF is ruptured and the current decreases (Fig. 7, Segment 4). This is the transition to a state with high resistance.

CONCLUSION

In this work we have presented a microscopic model of the bipolar resistive switching mechanism. The distribution of the electron occupation probabilities calculated with the model is in excellent agreement with previous work. The simulated RRAM switching hysteresis cycle is in good agreement with the experimental result. The proposed microscopic model can be used for performance optimization of RRAM devices.

ACKNOWLEDGMENTS

This research is supported by the European Research Council through the grant #247056 MOSILSPIN.

REFERENCES

Chen, Y.S.; T.Y. Wu; P.J. Tzeng. 2009. "Forming-free HfO2 Bipolar RRAM Device with Improved Endurance and High Speed Operation." *Symp. on VLSI Tech.*, 37-38.

Derrida, B. 1998. "An Exactly Soluble Non-Equilibrium System: The Asymmetric Simple Exclusion Process." *Phys. Rep.*, vol. 301, no. 1-3, 65-83.

Dong, R.; D.S. Lee; W.F. Xiang; S.J. Oh; D.J. Seong; S.H. Heo. 2007. "Reproducible Hysteresis and Resistive Switching in Metal-CuxO-Metal Heterostructures." *Appl. Phys. Lett.*, vol. 90, no. 4, 42107/1-3.

Fujii, T.; M. Kawasaki; A. Sawa; H. Akoh; Y. Kawazoe; Y. Tokura. 2005. "Hysteretic Current-Voltage Characteristics and Resistance Switching at an Epitaxial Oxide Schottky Junction SrRuO3/SrTi0.99Nb0.01O3." *Appl. Phys. Lett.*, vol. 86, no. 1, art. no. 012107.

Gao, B.; B. Sun; H. Zhang; L. Liu; X. Liu; R. Han; J. Kang; B. Yu. 2009. "Unified Physical Model of Bipolar Oxide-Based Resistive Switching Memory." *IEEE Electron Device Lett.*, vol. 30, no. 12, 1326-1328.

Kim, S.; Y.K. Choi. 2009. "A Comprehensive Study of the Resistive Switching Mechanism in Al/TiOx/TiO2/Al-Structured RRAM." *IEEE Trans. Electron Devices*, vol. 56, no. 12, pp. 3049-3054.

Kinoshita, K.; T. Tamura; H. Aso; H. Noshiro; C. Yoshida; M. Aoki; Y. Sugiyama; H. Tanaka. 2006. "New Model Proposed for Switching Mechanism of ReRAM." *IEEE Non-Volatile Semicond. Memory Workshop*, 84-85.

Kryder, M.H.; C.S. Kim. 2009. "After Hard Drives - What Comes Next?" IEEE Trans. Magn., vol. 45, no. 10, 3406-3413.

Kugeler C.; C. Nauenheim; M. Meier; A. Rudiger; R. Waser. 2008. "Fast resistance switching of TiO2 and MSQ thin films for nonvolatile memory applications (RRAM)." *NVM Tech. Symp.*, 6.

Lee, S.; H. Kim; D.J. Yun; S.W. Rhee; K. Yong. 2009. "Resistive Switching Characteristics of ZnO Thin Film Grown on Stainless Steel for Flexible Nonvolatile Memory Device." *Appl. Phys. Lett.*, vol. 95, no. 26, 262113.

Lee, B.C.; P. Zhou; J. Yang; Y.T. Zhang; B. Zhao; E. Ipek; O. Mutlu; D. Burger. 2010. "Phase-Change Technology and the Future of Main Memory." *IEEE Micro*, vol. 30, no. 1, 131-141.

Lin, C.C.; C.Y. Lin; M.H. Lin. 2007. "Voltage-Polarity-Independent and High-Speed Resistive Switching Properties of V-Doped SrZrO3 Thin Films." *IEEE Trans. Electron Devices*, vol. 54, no. 12, 3146-3151.

Nian, Y.B.; J. Strozier; N.J. Wu; X. Chen; A. Ignatiev. 2007. "Evidence for an Oxygen Diffusion Model for the Electric Pulse Induced Resistance Change Effect in Transition-Metal Oxides." *Phys. Rev. Lett.*, vol. 98, no. 14, 146403/1-4.

Nishi, Y.; J.R. Jameson. 2008. "Recent Progress in Resistance Change Memory." *Dev. Res. Conf.*, 271-274.

Press, W.H.; S.A. Teukolsky; W.T. Vetterling; B.P. Flannery. 1992. "Numerical Recipes in C: the art of scientific computing." *Cambridge University Press.*

Rozenberg, M.J.; I.H. Inoue; M.J. Sanchez. 2004. "Nonvolatile Memory with Multilevel Switching: A Basic Model." *Phys. Rev. Lett.*, vol. 92, no. 17, 178302-1.

Russo, U.; D. Ielmini; C. Cagli; A.L. Lacaita; S. Spiga; C. Wiemer; M. Perego; M. Fanciulli. 2007. "Conductive-Filament Switching Analysis and Self-Accelerated Thermal Dissolution Model for Reset in NiO-Based RRAM." *IEDM Tech. Dig.*, 775-778.

Sawa, A.; T. Fujii; M. Kawasaki; Y. Tokura. 2004. "Hysteretic CurrentVoltage Characteristics and Resistance Switching at a Rectifying Ti/Pr0.7Ca0.3MnO3 Interface." *Appl. Phys. Lett.*, vol. 85, no. 18, 4073-4075.

Schmidt-Mende, L.; J.L. MacManus-Driscoll. 2007. "ZnO - nanostructures, defects, and devices." *Materials today*, vol. 10, 40.

Seo, S.; M.J. Lee; D.H Seo; S.K. Choi; D.S. Suh; Y.S. Joung; I.K. Yoo; I.S. Byun; I.R. Hwang; S.H. Kim; B.H. Park. 2005. "Conductivity Switching Characteristics and Reset Currents in NiO Films." *Appl. Phys. Lett.*, vol. 86, no. 9.

Sverdlov, V.; A.N. Korotkov; K.K. Likharev. 2001. "Shot-Noise Suppression at Two-Dimensional Hopping." *Phys. Rev. B*, vol.63, 081302.

Szot, K.; W. Speier; G. Bihlmayer; R. Waser. 2006. "Switching the Electrical Resistance of Individual Dislocations in Single-Crystalline SrTiO3." *Nature Materials*, vol. 5, 312-320.

Watanabe, Y.; J.G. Bednorz; A. Bietsch; Ch. Gerber; D. Widmer; A. Beck; S.J. Wind. 2001. "Current-Driven InsulatorConductor Transition and Nonvolatile Memory in Chromium-Doped SrTiO3 Single Crystals." *Appl. Phys. Lett.*, vol. 78, no. 23, 3738-3740.

Wu, S.X.; L.M. Xu; X.J. Xing. 2008. "Reverse-Bias-Induced Bipolar Resistance Switching in Pt/-TiO2/SrTi0.99Nb0.01O3/Pt Devices." *Appl. Phys. Lett.*, vol. 93, no. 4, 043502/1-3.

Reliable Initialization of GPU-enabled Parallel Stochastic Simulations Using Mersenne Twister for Graphics Processors

Jonathan Passerat-Palmbach[1,3] Claude Mazel[1,3] Antoine Mahul[2] David R.C. Hill[1,3]
passerat@isima.fr david.hill@univ-bpclermont.fr

[1] *Clermont Université, Université Blaise Pascal, LIMOS, BP 10448, F-63000 CLERMONT-FERRAND*
[2] *Clermont Université, Université Blaise Pascal, BP 10448, F-63000 CLERMONT-FERRAND*
[3] *CNRS, UMR 6158, LIMOS, F-63173 AUBIERE*

KEYWORDS
Parallel Stochastic Simulation, Random Numbers Generation, GP-GPU, CUDA, MTGP, TestU01

ABSTRACT

Parallel stochastic simulations tend to exploit more and more computing power and they are now also developed for General Purpose Graphics Process Units (GP-GPUs). Consequently, they need reliable random sources to feed their applications. We propose a survey of the current Pseudo Random Numbers Generators (PRNG) available on GPU. We give a particular focus to the recent Mersenne Twister for Graphics Processors (MTGP) that has just been released. Our work provides empirically checked statuses designed to initialize a particular configuration of this generator, in order to prevent any potential bias introduced by the parallelization of the PRNG.

INTRODUCTION

Over the past five years, simulationists have leant towards Graphics Process Units (GPUs) to compute the heavy tasks bound to their research activity. Stochastic simulations, especially Monte Carlo (Gentle 2003), are widely spread through many scientific communities. General-Purpose GPUs (GP-GPUs) with many cores are very interesting for this simulation technique. Depending on the domain in which this kind of simulation is used, results may suffer from a weak parallelization technique with some stochastic streams of poor quality. We have shown that for sensitive research domains, like nuclear medicine, parallel simulations cannot cope with biased results (El Bitar et al. 2006; Reuillon et al. 2008).

Finding a fast and reliable Pseudo Random Number Generator (PRNG) to feed a sequential stochastic simulation is not a problem for many application domains since more than a decade. This issue has been tackled in many reference studies for CPU based PRNG (L'Ecuyer 1990). (Park and Miller 1988) raised the fact that one should consider the couple (application, PRNG) instead of limiting the study to the intrinsic qualities of the PRNG. For instance, a very good generator like Mersenne Twister (MT) (Matsumoto and Nishimura 1998) should not be trusted for a cryptographic application and the initialization of such generators has to be done carefully. Indeed, for some years, this very nice generator was sensible to its initialization status. Even if we never have a universal generator, the MT family of generators (Saito and Matsumoto 2008), the WELL generators (Panneton et al. 2006) and some advanced Multiple Recursive pseudo-random numbers Generators (MRGs) from L'Ecuyer (L'Ecuyer et al. 2002) give very good results when considered for parallel computing in a wide range of applications.

If some criteria have been gathered by Coddington (Coddington 1996) for sequential and parallel simulation to characterize good PRNGs, it is often safer to refer to empirical testing software. Several libraries have been designed in this way, the oldest and most renowned include the statistical tests proposed by Knuth (Knuth 1969), the DieHard testing suite designed by Marsaglia (Marsaglia 1996) and the Brown's DieHarder suite (Brown et al. 2010). The TestU01 library provided by L'Ecuyer (L'Ecuyer and Simard 2007) is the ultimate test software at the time of this paper writing. The latter contains a battery of tests called BigCrush, the most stringent set of tests that a PRNG can be faced nowadays.

The main problem we are still facing today is to ensure the correct behavior of the PRNG when distributed across a tight or large coupled computing architecture. The literature currently provides quite a few references about stochastic streams distribution on classical hardware architectures (Mascagni 1997), (Traore and Hill 2001), (Bauke and Mertens 2007), (Hill 2010). The set of references is even poorer when considering a GP-GPU environment. Indeed, restricted parallel hardware architectures like the Single Instruction Multiple Data (SIMD) family, which GPUs belong to, do highly impact the implementation of generators. In addition, we

still have to select the best way to allocate random substreams to these manycore architectures.

In this paper, we bring our contribution to these problems:

- We survey the current PRNGs implementations and solutions available to distribute stochastic streams through a GPU architecture.
- The parameterized PRNGs family is proposed taking the example of MTGP (Saito 2010b), the generator we find the most interesting in this context.
- We present the latter and discuss its quality with empirical benchmarks and statistical analysis. Reliable initialization data structures for a particular configuration of MTGP are also presented.

In this manner, we first intend to analyze the features of a particular generator designed for GPU hardware architectures: MTGP. The second purpose of this study is to give reliable parameters to initialize this PRNG, without introducing any potential bias in the parallel stochastic simulations based upon it.

PSEUDO RANDOM NUMBERS FOR GP-GPUs

Until recently, designing a PRNG for GPU-enabled platforms could be very tricky as it forced programmers to deal with graphics Application Programming Interfaces (APIs). Some implementations are presented in (Sussman et al. 2006). The authors especially list the limitations of these GPU dedicated PRNGs due to the past weaknesses of the hardware. Limited output per thread or untruthful operations were part of the restrictions that made these PRNGs feeble for High Performance Computing (HPC) applications. Consequently, a common way to deal with random numbers on GPU was to generate them on CPU before transferring them on the graphics processor. This solution has to face the well-known bottleneck of data transfer between the CPU host and the GPU device. Even with nowadays PCI Express 16X running at 8GB/s, this remains a challenge for high performance applications.

Since 2008 and the recent advances from NVIDIA, new GPU software and hardware architectures offer the precision and speed needed for many HPC applications. Now, PRNGs can be directly implemented into the GPU. Recent works propose this new kind of generators. Langdon presents a minimal implementation of the standard Park Miller PRNG (Park and Miller 1988) on a NVIDIA 8800 GTX GPU in its paper from 2008 (Langdon 2008). He announces a speed up of more than 40 compared to his Intel 2.40 GHz CPU. One year later, (Langdon 2009) increased again the speed of his application by four by using the new NVIDIA technology CUDA (Compute Unified Device Architecture) (NVIDIA 2010) with a Tesla T10 GPU. Nevertheless, we do not advise the use of this old generator which has many known flows, though it was still in use until recently in some well distributed networking simulation software (Entacher and Hechenleitner 2003).

CUDA has been designed to allow developers to easily harness the computation power of GPUs. In his first implementation, Langdon had to deal with a complex and unadapted graphics API. With CUDA, developers can program GPUs without wasting their time making algorithms and their data fit into graphics dedicated data structures, such as pixels shaders. Furthermore, CUDA does not propose a new programming language but only some C extensions, making it easier to learn for C familiars. The CUDA appellation also refers to the name of the new NVIDIA GP-GPU architecture. This generation of graphic boards tries to fulfill the requirements noted in the conclusion of the previously cited (Sussman et al. 2006), with for instance an implementation of the IEEE 754 floating point numbers standard. The new generation of boards based upon the Fermi architecture is now proposing configurable L1 cache, ECC memory and a considerable increase of performance in double precision, while owning twice as much cores as the antecedently mentioned T10 processor.

Although these highly parallel devices bring much more peak performances than CPUs, they must be carefully programmed to deliver the expected power. In fact, GPU architectures combine a manycore approach with SIMD vector cores. As vector processors do, GPU-enabled algorithms need to repeat the same operation on different data to correctly exploit the device. This is the main reason of the recent dedicated PRNGs proposals. In 2006, (Saito and Matsumoto 2008) proposed an SIMD version of the famous 'Mersenne Twister' called SIMD-oriented Fast Mersenne Twister (SFMT). Although this algorithm can be used on regular CPUs or on SIMD enabled CPUs (using either SSE or AltiVec vector instructions), it cannot be directly transposed to a GPU architecture. Most PRNGs have to be rethought from scratch to leverage GPUs characteristics, once again, we always have to take into account the target application. In the case of a CUDA implementation, these couples are well surveyed in (Howes and Thomas 2007).

Given that CUDA defines software levels that map the device architecture, PRNGs implemented using this technology can be organized at one of the following scopes, corresponding to the main elements of the CUDA framework: a thread, a block of threads or a kernel (the program running on the whole GPU). All

these approaches have been studied in the literature. In (Zhmurov et al. 2010), authors present three basic generation algorithms working either with a single instance of the PRNG for the kernel or with an instance per thread. The three algorithms exposed are quite basic: Ran2, Hybrid Taus and a Lagged Fibonacci generator. In the same way, (Langdon 2009) chooses to generate a number per thread in its GPU version of the Park-Miller algorithm. The last strategy is proposed in (Saito 2010b) where a new variant of the MT algorithm spreads independent PRNGs through each thread block, thanks to an algorithm known as Dynamic Creator (DC) (Matsumoto and Nishimura 2000) that we will detail later.

Beyond the nature of the PRNG algorithm, we prefer to focus on the scope chosen for each implementation. Indeed, we formerly insisted on the need to consider the target application and the PRNG as a pair. Obviously, new PRNG algorithms have to take advantage of GPU intrinsic properties such as heterogeneous memories, or thread organization. The former highly impacts the PRNG performance. Considering the approach using a generator per thread approach, an internal state array has to be saved in each thread. CUDA related works like (Kirk and Hwu 2010) specify that arrays declared for a thread are stored in the local memory, implemented in RAM. Equivalently, with a PRNG for all thread approach, the global memory is solicited to store the state of the PRNG. Each thread draws a number and updates its component of the state in global memory, implemented in RAM too. These two approaches make a heavy use of global memory, which has the advantage to be persistent across kernel launches within the same application. Yet, this RAM area is quite slow, it implies a 400 to 800 clock cycles latency because it is not cached (NVIDIA 2010). So, even if the global memory storage is compulsory to save the PRNG state between two kernel calls, one can use the shared memory, reachable by every thread within a block, to manipulate PRNG data. Indeed, it is implemented on-chip and is consequently as fast as registers. A good example of this choice is the "to be published" paper introducing MTGP (Saito 2010b).

In our opinion, a good GPU PRNG should employ shared memory. A PRNG per block approach seems to be the most appropriate way to implement a source of randomness, first, because it exploits the quickest memory, second, for the sake of applications upgradability. Since hardware architectures evolve very quickly, we cannot afford to rethink algorithms every time the memory amount or number of threads available doubles. So, fixing a block of threads grained scope for a PRNG algorithm is the safest solution to eschew lots of modifications tied to frequent hardware evolutions. This is the reason why we have decided to study in details the Saito proposition: MTGP.

DESCRIPTION OF MTGP

Data Structure

At the time we are writing this paper, MTGP was not referenced by any scientific publications yet, except that we found on the Internet that Saito's work is to be published soon (Saito 2010b). We must describe its features in order to introduce the goal of our study.

First of all, MTGP is obviously inheriting from the properties of its elder, though it is quite different from a simple GPU implementation of the original MT, as seen in (Podlozhnyuk 2007). As a matter of fact, Saito uses the original paper describing MT (Matsumoto and Nishimura 1998) to lay his generator out. Thus, we will see that MTGP can suffer from some little problems already spotted for the MT "family". Since we often champion this family of generators, (Reuillon 2008) has studied 2^{16} statuses of the original MT algorithm using the TestU01 Crush test battery from (L'Ecuyer and Simard 2007). The involved tests verify the linear complexity of the random sequence. MTGP is based upon the same linear recurrence to create random sequences, so it is not recommended for cryptographic purposes.

We have also noted that MTGP specified a common notion of the generators belonging to the parameterized family. We distinguish this cast of generators by the compound form of their data structures. It contains two distinct elements implied in the generation algorithm, we call them seed status and parameterized status. The first is basically the common seed given by the user to initialize a generator. The second stores parameters determined at a particular PRNG creation, it is supposed to be sort of a unique signature of the generator. Both these concepts were already present in MT. MTGP makes them more precise by explicitly using a data structure of the form we described in this paragraph. We propose a simple class diagram of this concept in figure 1:

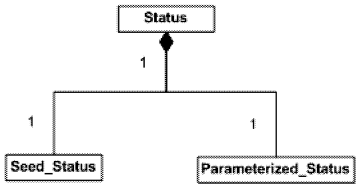


Figure 1: Class diagram of a parameterized PRNG

MTGP takes this idea one step further by introducing two kinds of statuses: the references and the fasts. The latters use pre-stored elements to decrease the initialization time, and programming techniques such as inlining to speed up the execution time. However, it results in a memory greedier status.

Parameterized statuses are a common way to ensure independence between parallel stochastic streams. This technique is called parameterization in the literature (Hill 2010). Depending on the way it is settled, it can lead to poor results (De Matteis and Pagnutti 1995). Unfortunately, we do not have any mathematical theorem allowing us to check the independence between two generators, according to their parameters. However, MT came along with the DC algorithm, designed to create large sets of independent generators. This algorithm integrates an identifier, often the one of the processor or thread that will host the PRNG, to distinguish parallel random streams. It uses it as a part of the characteristic polynomial of the matrix used by the MT algorithm. We can easily conclude that if characteristic polynomials are prime to each other, their associated matrices will also be unique in the same context. Hence, we obtain independent parameter sets.

Hopefully this algorithm has been renewed for MTGP, enabling us to proceed in the same way. Furthermore, it improves the original algorithm by allowing the user to get a larger set of 2^{32} parameters, where the original algorithm could only deliver 2^{16} sets. This number is now too small for large computing grids such as the European Grid Initiative (EGI), with more than 240000 cores at the time we are writing this paper. This latter point forces us to keep skeptical concerning the independence of the MTGP produced by the new DC. Saito explains that a SHA1 (Secure Hash Algorithm) checksum of each characteristic polynomial is generated to let the user check he did not get duplicated entries.

Architecture Independence

One of the most interesting features of MTGP is to be available for both CPUs and GPUs architectures. Even if MTGP has been designed to run on GPUs, you can also find a CPU version at the same location (Saito 2010a). We have based our study on the capability of the generator to merge transparently in CPU-based applications. Hence, we were able to test the PRNG on any CPU-based host with reliable and well-known tools like TestU01. This way, we have avoided the hazardous implementation of a new empirical test battery, which would have to be validated before.

Moreover, this property is really precious in our opinion. We intend to use this PRNG for stochastic simulations following the hybrid computing paradigm, where the sequential part of the application runs on a CPU host, while the parallel-one is executed on a GPU board. In such cases, stochastic streams will furnish random numbers to both the CPU and the GPU. With a PRNG like MTGP, we can keep our simulations homogenous, using the same PRNG on each computing element engaged. Handling independent parallel stochastic streams becomes understandably important when you have to deal with such hardware configurations. We will study in a next paragraph the specific DC coming along with MTGP to maximize this independence.

Now considering the new elements we introduced in this whole section, we can extend our previous object model to the particular MTGP. Figure 2 depicts its main components:

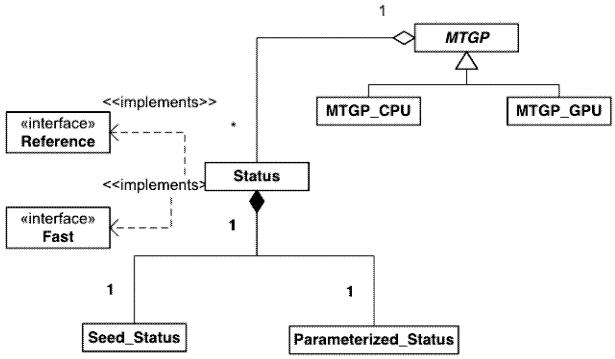


Figure 2: Class Diagram for MTGP and its components

We have widely presented MTGP statuses in this section. Since this generator utilizes a PRNG per block of threads approach, we will need a different status per block to ensure the independence between the random streams produced. The sample program furnished by Saito takes a number of blocks to use as an argument and owns a set of 128 different statuses to feed these blocks. In the next section, we present the protocol that helped us to issue a large number of statuses for MTGP users.

MTGP BENCHMARK

Empiric Test of 10000 Statuses

We formerly introduced the DC tool, enabling us to create independent PRNGs to use in a parallel environment. It has supplied us 10000 independent parameter sets (parameterized statuses), each corresponding to a different MTGP. This step can be very computationally expensive when the algorithm has to look for generators displaying huge periods, such as 2^{19937} for the original MT. According to (L'Ecuyer 2010), periods contained between 2^{100} and 2^{200} should be sufficient for nowadays stochastic applications. Thus, we decided to manipulate generators of the lowest period allowed by the MTGP DC, which is still 2^{3217}. Moreover, the lower the period is, the fewer it wastes bits to store the internal state vector of the PRNG, helping us to save some GPU memory. With this configuration, we have been able to get our

entire ready to use MTGPs in a single day, using a 256-node Opteron-enabled Linux cluster.

The second phase consisted in applying the BigCrush test battery to each newly created generator, in order to check their quality. First of all, to easily analyze such an amount of results, we modified the TestU01 library output to enable it to produce lighter results output files. In this manner, we have been able to parse results files using script tools like *Sed* or *Awk* to generate statistics. Moreover, since lots of our computations have taken place on the European computing Grid Infrastructure (EGI), we reduced the quantity of data transferred on this slow bandwidth storage space. The use of this HPC tool was compulsory in our case, in fact (L'Ecuyer and Simard 2009) forecasts BigCrush to take about 8 hours of CPU time on an average 64-bit processor. We could not afford to perform the equivalent of 80000 CPU-hours on a single cluster to get our results in a decent time.

Our final aim is to provide verified parameterized statuses to allow the simulation community to initialize GPU-enabled PRNGs without introducing any bias in stochastic simulations, according to the current knowledge. A basic selection consisted in keeping only statuses that had perfectly passed all tests of the battery. But given that this approach eliminated approximately 40% of the statuses, we tried to determine whether other statuses could be kept with a good confidence level. We set up a more formal analysis to answer this question.

Statistics-Based Analysis

Each test of the TestU01 Bigcrush battery (L'Ecuyer and Simard 2007) is governed by the H_0 hypothesis, that the successive output values of the RNG are i.i.d. U(0, 1), i.e. are independent random variables from the uniform distribution over the interval [0;1]. These tests are defined by a test statistic γ (which is a function of the numbers to be tested). They compute and report a number, called the p-value of the test, which is contained between 0 and 1. Furthermore, if γ has a continuous distribution, the p-value is U(0, 1) under H_0. At this point, let us consider two precisions from L'Ecuyer and Simard:

1. « If the p-value is extremely small (e.g., less than 10^{-10}, then it is clear that the RNG fails the test, whereas if it is not very close to 0 or 1, no problem is detected by this test. » (L'Ecuyer and Simard 2007);

2. « Moreover, when a generator starts failing a test decisively, the p-value of the test usually converges to 0 or 1 exponentially fast as a function of the sample size when the sample size is increased further. » (L'Ecuyer and Simard 2009).

According to these quotations, we decided to consider three p-value types, detailed hereafter:

- p-values contained between [0.001;0.999] are reckoned as correct, (these values are proposed by the TestU01 library);

- those included in the range [0;0,001[U]0.999;1] are counting as suspect;

- lastly, we refined the previous range since we needed to take into account extremely small p-values (less than 10^{-10}), called disastrous afterwards.

As mentioned previously, we ran our tests on 10000 independent statuses, with regards to MTGP DC. The column chart appearing on figure 3 represents the number of suspect p-values noticed for statuses where no disastrous p-values were obtained (due to layout considerations, only the thirty-two first tests are present on figure 3):

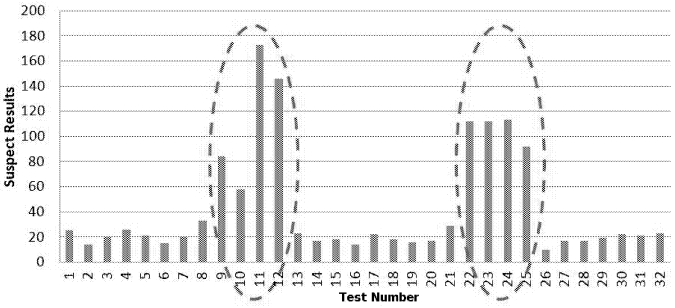


Figure 3: Number of suspect results versus test numbers (extract displaying tests 1 through 32)

The latter figure helped us to easily identify three test groups. They distinguish from others by recording more than 40 suspect p-values. The interesting point here is that these three groups characterize some aspects of the generator behavior. In fact, tests belonging to the same group are just differently parameterized versions of the same test. The noticeable tests are described as follows in the TestU01 documentation (L'Ecuyer and Simard 2009):

- *smarsa_CollisionOver* (tests 9 to 12), is an overlapping pairs sparse occupancy (OPSO) test introduced in (Marsaglia 1985);

- *snpair_ClosePairs* (tests 22 to 25), is a m-nearest-pairs (m-NP) test (L'Ecuyer and Simard 2009);

- *swalk_RandomWalk1* (tests 74 to 79), applies simultaneously several tests based on a random walk of length l over the integers, for several (even) values of l (L'Ecuyer and Simard 2009).

Under the H_0' hypothesis of a uniform distribution of the p-values over the interval [0;1], the distribution of the number of suspect p-values is binomial with parameters n = 10000 and p = 2/1000. So, we can reject the H_0' hypothesis with a very good confidence level (about 7.10^{-6} for each test that accumulates more than 40 suspect p-values. However, we cannot do the same with H_0. To do so, the γ-statistic used by the test should present a continuous distribution, which is not the case for the considered tests. Concretely, it means that pointing out suspect p-values brings useful information, but no matter the excesses of suspect p-values, such results do not provide a formal statistic proof of the test failure. That is why we use to consider a test is failed only when it returns disastrous p-values.

Six tests are missing on the previous figure: four, the 70, 71, 80, 81, were introduced in our MT description as problematic tests for any MT-like generator. So we increased the execution speed of the test battery simply by disabling those tests that would have systematically produced disastrous p-values. The other two tests, numbered 35 and 100, are the problematic ones. Only these assessments issue disastrous p-values in a non negligible quantity. We noted that about a tenth statuses were failing the 100^{th} Test, while more than a fifth went wrong with the 35^{th} Test. Figure 4 gives a graphical representation of the announced proportions:

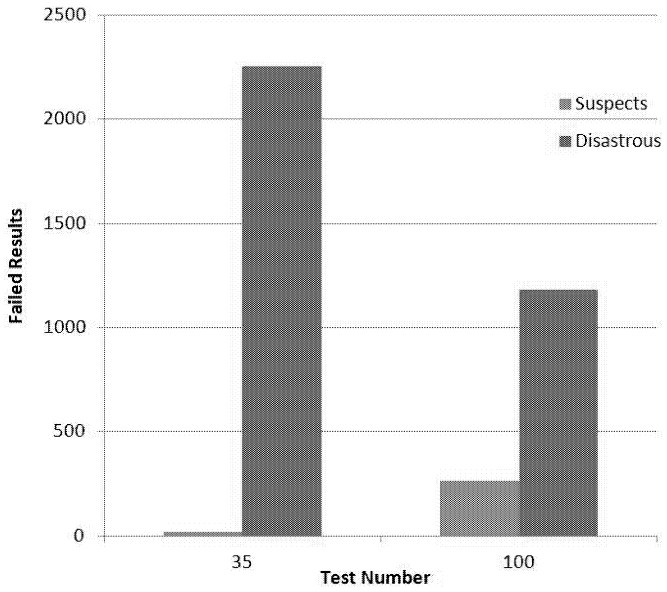


Figure 4: Detailed Results for Tests 35 and 100 of the BigCrush battery

Test number 35 is the *sknuth_Gap* test with N = 1, n = 3.10^8, r = 25, Alpha = 0 and Beta = 1/32 (Knuth 1969). This test counts, for s = 0, 1, 2, ... « the number of times that a sequence of exactly s successive values fall outside the interval [Alpha, Beta] (this is the number of gaps of length s between visits to [Alpha, Beta]). It then applies a chi-square test to compare the expected and observed number of observations. » (L'Ecuyer and Simard 2009). A typical generator miscarrying this test « wanders in and out of [Alpha, Beta] for some time, then goes away from [Alpha, Beta] for a long while, and so on » (L'Ecuyer and Simard 2007). However, one should note that the same test is perfectly passed with other Beta values. In view of the analysis we propose, this fact is obviously logical for Beta values lower than the incriminated one (1/32), but it is rather strange not to find disastrous p-values with a higher Beta value (Test 34 sets Beta to 1/16).

The test number 100 is referenced as *sstring_HammingIndep* test with N = 1, n = 107, r = 25, s = 5, L = 1200 and d = 0. It applies two tests of independence between the Hamming weights of successive blocks of L bits (L'Ecuyer and Simard 1999). According to François Panneton, this test measures Hamming-weight dependencies between random values issued by a given generator. It tends to demonstrate that the recurrence does not shuffle bits enough from an iteration to another (Panneton 2004).

With this first experiment, we have put under the spotlight difficulties encountered by MTGP 2^{3217}. Assuming that these problems are mostly concentrated on the two properties checked by tests 35 and 100, those producing disastrous p-values, we have focused our further studies on them.

PARAMETERIZED STATUS INFLUENCE

Seed Status Variation

Let us recall that the 10 000 tested statuses were only differing from their parameterized parts, whereas they shared the same seed status, fixed to 0. Viewing the previous results, we decided to work out whether the fact of passing a test or not, and by the way the quality of a generator, was due to either its parameterized status, or its seed status, or both. To do so, we settled a new experiment, sieving a set of 100 parameterized statuses alternately associated with 100 seed statuses randomly chosen. This initialization technique, called Random Spacing, represented a total of 10 000 combinations put to the proof of the guilty tests presented above. Figure 5 shows an extract of the graphical output for test 35. Red crosses mark a disastrous p-value, while blue ones indicate suspect p-values. An aggregation of red crosses on vertical lines shows that the parameterized status on the abscissa failed the considered test for all the seed statuses it was associated to. So, in the case of parameterized generators, it seems that the parameter-

ized status establishes the generator quality on its own, independently from the selected seed status.

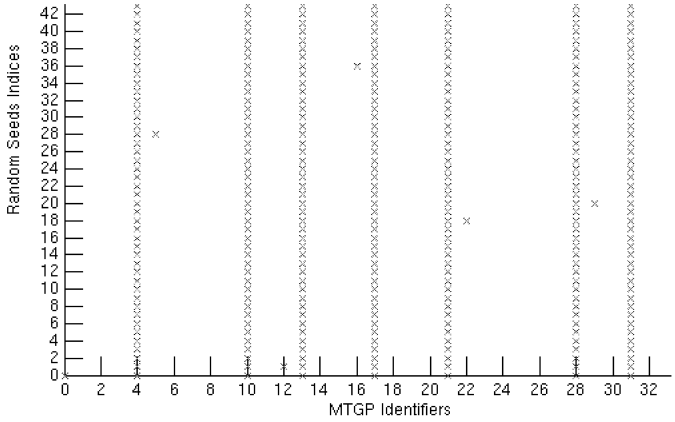


Figure 5: Extracts of the results for Test 35: MTGP identifiers versus random seeds indices

Period Variation

To comfort our previous assertion, we tried to make other elements of the parameterized status vary to observe their impact on the quality of the generator. As long as DC tries to figure out a tempering matrix such as the PRNG produces a well distributed sequence (Matsumoto and Nishimura 2000), modifying this period should highly impact this property for the newly created statuses. Now, we previously brought forward that Test 35 (*sknuth_gap*) of BigCrush based its judgment on this characteristic. So, we intended to obtain much better results using 1000 MTGPs of period 2^{23209}, confronted to tests 35 and 100 only. The result is crystal-clear since 99.5% of the statuses passed both tests without any problem. Moreover, we only noticed suspect p-values in the other 0.05%.

Obviously, a higher period eliminates sequence distribution issues, but this latter result could hide potential intrinsic weaknesses of the MTGP algorithm. Our last experiment will introduce as a standard a quality-proven PRNG of the same family: the original MT (Matsumoto and Nishimura 1998).

Algorithm Variation

The original MT is designed with linear-recurrences preventing its recommendation for some particular applications such as cryptography. As far as we know, no other problems are referenced concerning this PRNG. That is why we consider it as a good standard to compare with the target of our investigations. An interesting point is that its own DC is able to produce parameters for the 2^{3217} period, thus allowing us to work with MTs and MTGPs of the same period. This way, we focus our experiment on the algorithm-dependent parts of the parameterized statuses. Once again, we observed the behavior of each generator when faced to tests 35 and 100. We selected a sample of 1000 independent PRNGs to compare outputs with our previous benchmarks. Results are even more clear-cut than before: 99.9% of MT-dedicated statuses passed the two tests without any failure, whereas about 64% of MTGP statuses did.

The two previous results tend to show the influence of parameterized statuses. Here we have shown that this data structure is tightly bound to the algorithm using it. MTGP seems to present weaknesses when configured with shorter periods, since MT, running with the same relatively small period, eschews traps where its recent variant falls frequently into. This section results are summed up on the column chart displayed on figure 6:

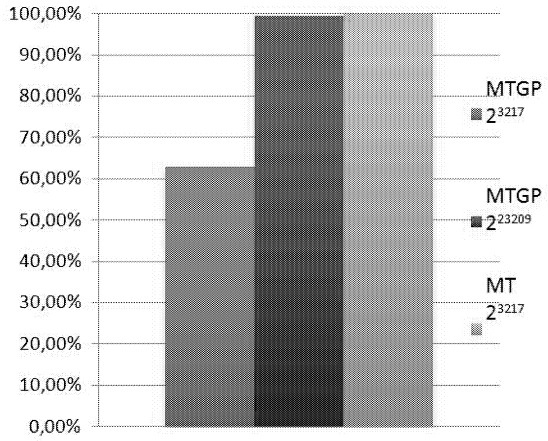


Figure 6: Percentage of passed results noticed for tests 35 and 100 depending on the PRNG

CONCLUSION

This work was first intended to study a particular generator designed for GPU hardware architectures: MTGP. After an introduction to other recent approaches mentioned in the literature, in order to compare them with the MTGP strategy, we provided a description of MTGP and proposed a generic object model representing generators using distinct seeds and parameters. To complete our description, we achieved some analysis of this PRNG by facing it to the current most stringent test battery: BigCrush from the TestU01 test software library. Weaknesses identified during these experiments have been reduced by comparisons with other configurations of the generator as well as with the original MT.

The second goal of this study was to furnish parameterized statuses to the simulation community, to allow them to quickly and safely initialize their MTGPs without introducing any potential bias with a bad status.

Only statuses that have passed BigCrush have been kept and will be freely available on the Internet.

We have shown that MTGP was safer with longer periods, according to TestU01 criteria, but the more a PRNG period is long, the more space it needs to store the internal state vector used by its algorithm. Nowadays, GPUs memory characteristics do not allow us to waste bytes to store PRNG data without influencing the whole application speed. By selecting only parameterized statuses referenced by our study or proofed by an equivalent benchmarking protocol, scientists using MTGP with the lowest available period (i.e. 2^{3217}) can dramatically reduce the memory footprint of their hybrid stochastic simulations.

However, we should not forget that the empirical tests provided by TestU01 consider random sequences individually. So, we are now thinking of a way to test mixed random sequences, the way they can be in a single GPU application using several blocks of threads fed with different sources of randomness.

REFERENCES

Bauke, H. and Mertens, S. (2007). Random numbers for large scale distributed monte carlo simulations. *Physical Review E*, 75(6).

Brown, R., Eddelbuettel, D., and Bauer, D. (2010). Dieharder: A random number test suite. `http://www.phy.duke.edu/~rgb/General/dieharder.php`.

Coddington, P. (1996). Random number generator for parallel computers. Technical report, Notheast Parallel Architecture Center.

De Matteis, A. and Pagnutti, S. (1995). Controlling correlation in parallel monte carlo. *Parallel Computing*, pages 73–84.

El Bitar, Z., Lazaro, D., Breton, V., Hill, D., and Buvat, I. (2006). Fully 3d monte carlo image reconstruction in spect using functional regions. *Nucl. Instr. Meth. Phys. Res*, 569:399–403.

Entacher, K. and Hechenleitner, B. (2003). Pitfalls when using parallel streams in omnet++ simulations. In *Interdomain Performance and Simualtion (IPS) Workshop.*

Gentle, J. (2003). *Random Number Generation and Monte Carlo Methods (Statistics and Computing).* Springer, 2 edition. ISBN-13: 978-0387001784.

Hill, D. (2010). Practical distribution of random streams for stochastic high performance computing. In *International Conference on High Performance Computing & Simulation (HPCS 2010)*, pages 1–8.

Howes, L. and Thomas, D. (2007). *GPU Gems 3*, chapter 37 - Efficient Random Number Generation and Application Using CUDA. Addison-Wesley Professional.

Kirk, D. and Hwu, W. (2010). *Programming Massively Parallel Processors.* Morgan Kaufmann.

Knuth, D. (1969). *The art of computer programming. Vol. 2, Seminumerical algorithms.* Addison-Wesley.

Langdon, W. (2008). A fast high quality pseudo random number generator for graphics processing units. In *IEEE CEC 2008, Hong Kong*, pages 459–465.

Langdon, W. (2009). A fast high quality pseudo random number generator for nvidia cuda. In *GECCO'09*, volume 10, pages 2511–2514. ACM Press.

L'Ecuyer, P. (1990). Random numbers for simulation. *Communications of the ACM*, 33:85–98.

L'Ecuyer, P. (2010). *Encyclopedia of Quantitative Finance*, chapter Pseudorandom Number Generators. Wiley.

L'Ecuyer, P. and Simard, R. (1999). beware of linear congruential generators with multipliers of the form a = ±2q±2r. *ACM Transactions on Mathematical Software*, 25(3):367–374.

L'Ecuyer, P. and Simard, R. (2007). Testu01: A c library for empirical testing of random number generators. *ACM Transactions on Mathematical Software*, 33(4):22:1–40.

L'Ecuyer, P. and Simard, R. (2009). *TestU01:A Software Library in ANSI C for Empirical Testing of Random Number Generators - User's guide, detailed version.* Département d'Informatique et de Recherche Opérationnelle - Université de Montréal.

L'Ecuyer, P., Simard, R., Chen, E., and Kelton, W. (2002). An object-oriented random-number package with many long streams and substreams. *Operations Research*, 50:1073–1075.

Marsaglia, G. (1985). A current view of random number generators. In *Computer Science and Statistics, Sixteenth Symposium on the Interface*, pages 3–10. Elsevier.

Marsaglia, G. (1996). The marsaglia random number cdrom, with the diehard battery of tests of randomness. `http://stat.fsu.edu/pub/diehard/cdrom`. Produced under a grant from the National Science at Florida State University.

Mascagni, M. (1997). Some methods of parallel pseudorandom number generation. *Algorithms for Parallel Processing*, pages 277–288.

Matsumoto, M. and Nishimura, T. (1998). Mersenne twister: A 623-dimensionnally equidistributed uniform pseudorandom number generator. *ACM Transactions on Modeling and Computer Simulations: Special Issue on Uniform Random Number Generation*, 8(1):3–30.

Matsumoto, M. and Nishimura, T. (2000). Dynamic creation of pseudorandom number generators. In *Monte Carlo and Quasi-Monte Carlo Methods 1998*, pages 56–69. Springer.

NVIDIA (2010). *NVIDIA CUDA Programming Guide Version 3.0.*

Panneton, F. (2004). *Construction d'ensembles de points basée sur des récurrences linéaires dans un corps fini de caractéristique 2 pour la simulation Monte Carlo et l'intégration quasi-Monte Carlo.* PhD thesis, Département d'informatique et de recherche opérationnelle, Université de Montréal, Canada.

Panneton, F., L'Ecuyer, P., and Matsumoto, M. (2006). Improved long-period generators based on linear reccurences modulo 2. *ACM Transactions on Mathematical Software*, 32:1–16.

Park, S. and Miller, K. (1988). Random number generators: Good ones are hard to find. *Communications of the ACM*, 31(10):1192–1201.

Podlozhnyuk, V. (2007). Parallel mersenne twister. Technical report, NVIDIA.

Reuillon, R. (2008). *Simulations stochastiques en environnements distribués - Application aux grilles de calcul.* PhD thesis, Université Blaise Pascal - École Doctorale Sciences pour l'Ingénieur.

Reuillon, R., Hill, D., El Bitar, Z., and Breton, V. (2008). Rigorous distribution of stochastic simulations using the distme toolkit. *IEEE Transactions on Nuclear Science*, 55(1):595–603.

Saito, M. (2010a). Mtgp download page. `http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/MTGP/index.html`.

Saito, M. (2010b). A variant of mersenne twister suitable for graphics processors. *CoRR.*

Saito, M. and Matsumoto, M. (2008). Simd-oriented fast mersenne twister: a 128-bit pseudorandom number generator. In Keller, A., Heinrich, S., and Niederreiter, H., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2006*, volume 2, pages 607–622. Springer Berlin Heidelberg.

Sussman, M., Crutchfield, W., and Papakipos, M. (2006). Pseudorandom number generation on the gpu. In *Graphics Hardware.*

Traore, M. and Hill, D. (2001). The use of random number generation for stochastic distributed simulation: application to ecological modeling. In *Proceedings of the 13th European Simulation Symposium*, pages 555–559.

Zhmurov, A., Rybnikov, K., Kholodov, Y., and Barsegov, V. (2010). Efficient pseudo-random number generators for biomolecular simulations on graphics processors. Technical report, CERN.

BIOGRAPHY

JONATHAN PASSERAT-PALMBACH is a PhD student at the LIMOS (ISIMA) - UMR CNRS 6158 of Blaise Pascal University of Clermont Ferrand (France). His work, is focused on high performance computing tools and discrete events simulation, applied to ecological modeling.
email: `passerat@isima.fr`

CLAUDE MAZEL is an associate professor at the ISIMA Computer Science and Modelling Institute, where he currently manages the Software Engineering and Computing Systems Department. His main scientific interests concern modelling, discrete event simulation, and design methods for simulation software, applied to ecological modelling.
email: `mazel@isima.fr`

ANTOINE MAHUL is currently an engineer in the field of scientific computing and managing of HPC and grid resources for universities of Clermont-Ferrand. He obtained his PhD in computer science at the LIMOS of Blaise Pascal University in 2005.
email: `antoine.mahul@univ-bpclermont.fr`

DAVID HILL is currently Vice President of Blaise Pascal University in charge of Computer Science and past co-director of ISIMA Computer Science & Modeling Institute. Since 1990, Professor Hill has authored or co-authored more than a hundred technical papers and journal papers and he has published various text books including free e-books from Blaise Pascal University Press (`http://www.isima.fr/~hill`).
email: `david.hill@univ-bpclermont.fr`

COMPLEX SOFTWARE SYSTEMS SIMULATION

MODEL DRIVEN REVERSE ENGINEERING FOR A TRANSCRANIAL MAGNETIC STIMULATION SIMULATION APPLIED TO SOFTWARE VERSIONING

Eric Innocenti
Università di Corsica - Pasquale Paoli
22, av. Jean Nicoli, 20250 CORTI – FRANCE
ino@univ-corse.fr

Sébastien Luquet, Vincent Barra and David R.C. Hill
UMR CNRS 6158 LIMOS
Blaise Pascal University - Clermont-Université
david.hill@univ-bpclermont.fr

KEYWORDS

Model Driven Engineering, Reverse Engineering, Metamodeling, Versioning, External Software libraries, Transcranial Magnetic Stimulation.

ABSTRACT

Transcranial Magnetic Stimulation (TMS) is a new technique for brain stimulation. TMS has several applications in medical and clinical research. Its use, however is still empirical and requires many stimulations to find the best coil position for stimulation. We have developed a simulation software of transcranial magnetic stimulation which computes the electromagnetic field induced in the cortex by TMS. This object-oriented software development has been revisited with a model driven approach. We have organised this article in two main parts. First the simulation tool with the computation of potential magnetic field outside the head is described. Then, we discuss the software engineering problems encountered with some possible solutions. The experience gained in this development is finally sketched in a model for software versioning.

INTRODUCTION

Transcranial Magnetic Stimulation (TMS) is a new technique for brain stimulation [Ruohonen, 1998]. As other techniques like ElectroConvulsive Therapy (ECT) or the implantation of electrodes into the motor cortex, TMS excites neurones. But contrary to those techniques, TMS is painless and noninvasive. Indeed, neurones are excited by electrical current induced by a rapidly changing magnetic field. This field is created by the discharge of thyristor (during about 300 µs) into a coil (a copper winding surrounded by a water-cooled system). The current circulating through the coil is about 10000 A in order to allow the magnetic field to cross the skull of the patient. TMS has several applications in medical and clinical research:

- brain mapping [George et al., 1999]
- psychiatry : treatment of mood disorder and schizophrenia [Davey et al., 1997]
- treatment of epilepsy [Ziemann et al., 1998]
- treatment of chronic pain [Lefaucheur et al., 2001]

But today, using TMS is still empirical. The clinician who wants to stimulate a specific cortex area from an expected result (phosphene, modification of vision, motor effect) will do a lot of stimulations, scanning over the patient's scalp to find the most important response (hot spot) [Thielscher and Kammer, 2002]. To avoid all those stimulations that modify the electrical activity of a wider zone of the cortex, we have developped a transcranial magnetic stimulation simulator. According to the position, the orientation of the coil and the parameters of the stimulation, the simulator gives a modeling of the stimulation effects. Results can be mapped on the patient's MRI image (Magnetic Resonance Image). A final objective could be to compute magnetic field, eddy currents within the skull, electrical induced field etc. All those scalar or vector fields are directed by complex laws as Maxwell equations. If we want to manage the main characteristics of a head; skull and cortex shape, non-homogenity, conductivity and permeability following biological structures [Krasteva et al., 2002], [H´edou, 1997], we need to use a very sophisticated finite-element scheme.

For the TMS part, the scope of this paper is limited to the numerical tool which computes the magnetic fields generated by a given coil at any point of the empty-space [Luquet et al, 2005a,b]. In this software development we encountered many problems linked to the modification and use of external libraries. Such an adaptation implies a careful versioning. Our errors have been analyzed and we propose possible solutions and a software versioning model which helps a better understanding of software development under the constraints of external libraries.

BASIC TMS MODELING AND RESULTS

The distribution of magnetic field $\vec{B}$ is defined by the Biot-Savart law:

(1)

$$\vec{B}(\vec{r},t) = \frac{\mu_0}{4\pi} I(t) \oint_C \frac{d\vec{l} \times (\vec{r} - \vec{r'})}{|\vec{r} - \vec{r'}|^3}$$

where $\mu 0$ is free space permeability, $I(t)$ intensity of the current going through the coil, $\vec{r}$ the point where the magnetic field is computed. $\vec{r'}$ is a vector describing the coil i.e the winding C. Although this law was established more than 100 years ago, it allows to understand parameters that will be crucial during computation. So as to develop a simple and re-useable tool, we will consider that $I(t)$ is constant and equal to its maximum. Indeed, even if a TMS is time-varying we retain this quasi-static case as a first step similarly to what [Ruohonen, 1998] and [Krasteva et al., 2002] and proposed. In addition we will use the magnetic potential vector $\vec{A}$ formulation, where:

(2) $\vec{B} = r\vec{o}t\vec{A}.$

Then, the Biot and Savart law becomes:

(3)
$$\vec{A}(\vec{r},t) = \frac{\mu_0}{4\pi} I \oint_C \frac{d\vec{l}}{|\vec{r}-\vec{r'}|}$$

and is thus mainly dependent of the winding shape. That's why manufacturers have developed different kinds of coil [Jalinous, 1998]:

- simple coil : a circular winding,
- 8-shaped coil : similar to two simple coils (currents circulating in opposite ways)
- double cone coil : an 8-shaped coil with an angle between the two wings of the coil.

The tool we have proposed is able to take into account the specificity of each winding copper shape. For a given contour, the user has only to define a 3D-parametrical curve and the expression of its derivative (in the long term, the derivative may be computed from the other parametrical equation with a symbolic computation module). The modeling of a single-turn coil whose centre is (x0, y0, z0), radius r and axis 0z, is for example:

$$\begin{aligned} x(t) &= x_0 + r\cos(t) & x'(t) &= -r\sin(t) \\ y(t) &= y_0 + r\sin(t) & y'(t) &= r\cos(t) \\ z(t) &= z_0 & z'(t) &= 0 \end{aligned}$$

Where t belongs to $[0, 2\pi]$. To describe a n-turns coil, t may lie in $[0, 2\pi n]$.

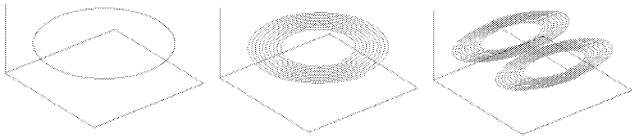


Figure 1: Circular, spiral and figure-8 shaped winding

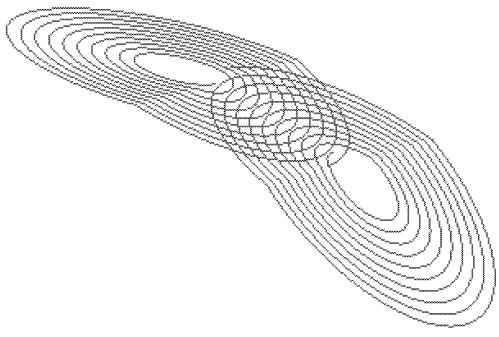


Figure 2: Medtronic MC-B70 coil

Every coil kind may thus be described by a parametrical model. Several particular details may also be managed e.g : a physical n-turn coil has a specific thickness, all turns are not overcome. It might be useful to model it with a spiral – shaped coil. In addition some coil (see [Nadeem et al., 2003] modelling) may have a covering winding. An example can be found in Figure 2 where we show the modeling of the Medtronic MC-B70 coil in our simulator. The two wings overlap on the middle of the coil and each wing does not belong to a simple plane. In such a case, a naive modelling would be deficient.

With those descriptions, the potential magnetic vector and by spatial derivate, $(r\vec{o}t)$ the magnetic field will be numerically computed at any point of the empty-space (Figure 3 hereafter). An 8-shaped coil is justified because $\vec{A}$ and $\vec{B}$ are more focused. Those fields cannot be computed for the points that are nearest the winding. Indeed, by definition the field is not defined on the winding (division by 0). In a second time $\vec{A}$ and $\vec{B}$ values could be given in a located plane parallel the to coil's one. Figure 4 shows the magnetic field modulus on plane $z = z0-r$. The results shown are typically similar to those obtained by [Nadeem et al., 2003] and as expected, an 8-shaped coil will produce a stronger magnetic field.

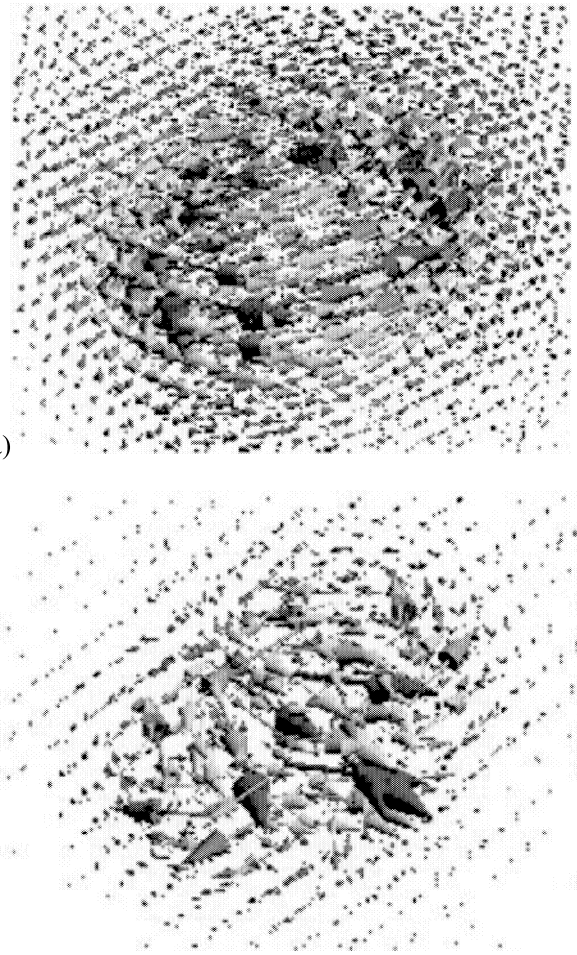


Figure 3: Potentiel magnetic field computation for circular (a) and 8-shaped coil (b)

In many references like [Krasteva et al., 2002], the time variation going through the stimulation coil is modelled as a

periodic signal. In fact the discharge could have different waveforms: slope, sinusoidal periods... Although a discharge is by definition short and time limited, a pragmatic way is to consider our system as a harmonic one. Under this assumption we can deduce the electric field $\vec{E}$ generated in empty space by stimulation:

$$\vec{E} = \frac{\partial \vec{A}}{\partial t} = i\omega \vec{A} \quad (4)$$

where ω is the frequency of the harmonic signal (typically 10 kHz) and i is the complex notation. Electric field amplitude $|\vec{E}|$ is then proportional to $|\vec{A}|$ (Fig 5).

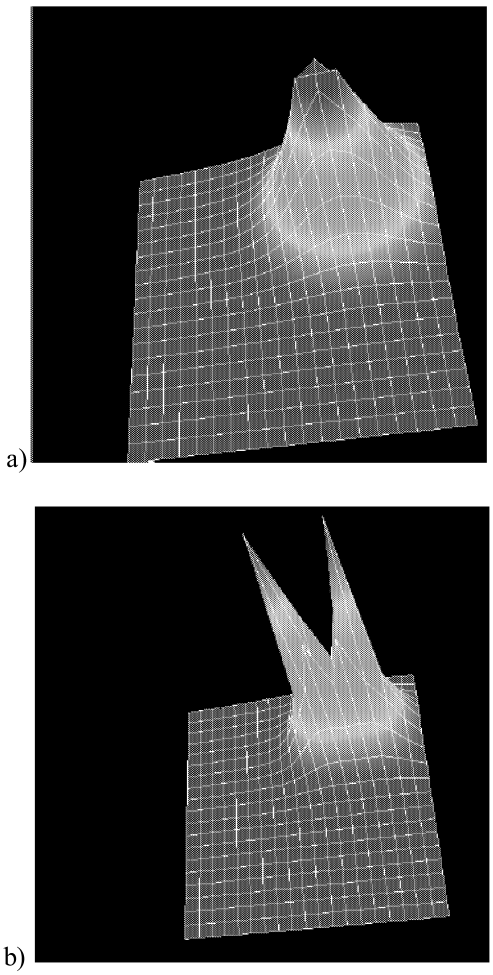


Figure 4: Normalized magnetic field modulus on plane $z = z0-r$ for circular (a) and 8-shaped coil (b)

An 8-shaped coil has a better focusing and the electric field is stronger under the centre of the 8-shaped coil. That is why according to several authors stimulation is expected to be the strongest when following coil axis. Therefore, many image-guided stimulation devices use this approximation. Future works could try to give a better definition area of stimulation and study the time effect. Our image-based navigation tools are restricted to mapping $|\vec{A}|$ on a patient's head-model generated from an MRI image (Figure 6).

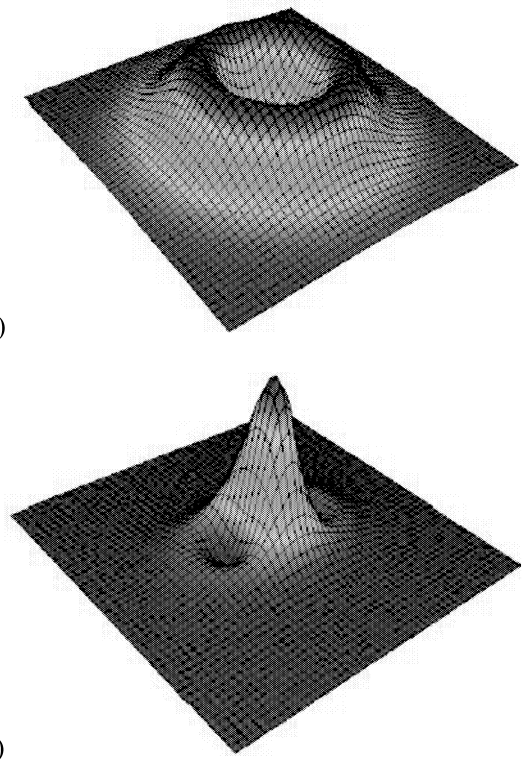


Figure 5: Normalized electric field modulus on plane $z = z0 - r$ for circular (a) and 8-shaped coil (b)

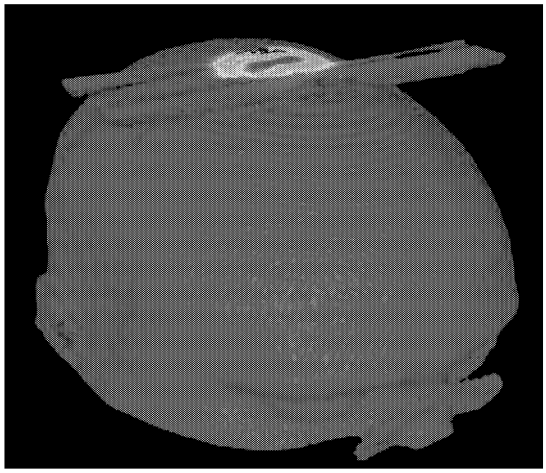


Figure 6: Potential magnetic field mapping generated by 8-shaped coil on patient's head

In figure 7 we see the graphical user interface of the simulator, the image shows the Module of the magnetic field generated by a double coil. This field presents a focalization below the intersection of the two wings of the coil. The intensity is represented by the height field, colors are just used as visual indicators. In figure 8 the simulator is used to test various placements of the coil and we noticed that a reverse placement of an 8-shaped coil gave a better focalization.

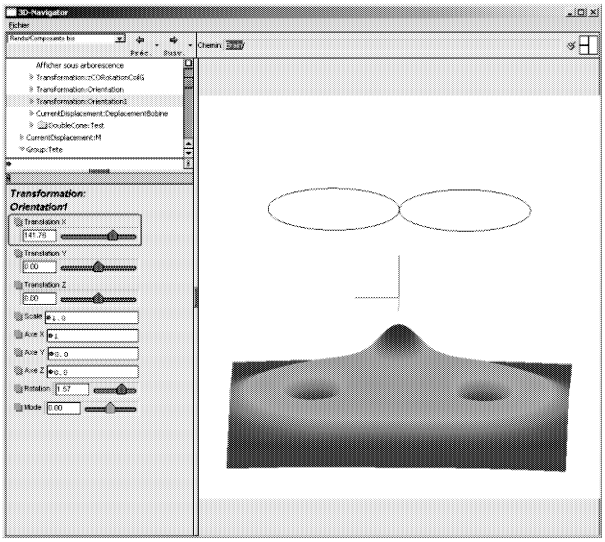


Figure 7: Module of the magnetic field generated by a double coil.

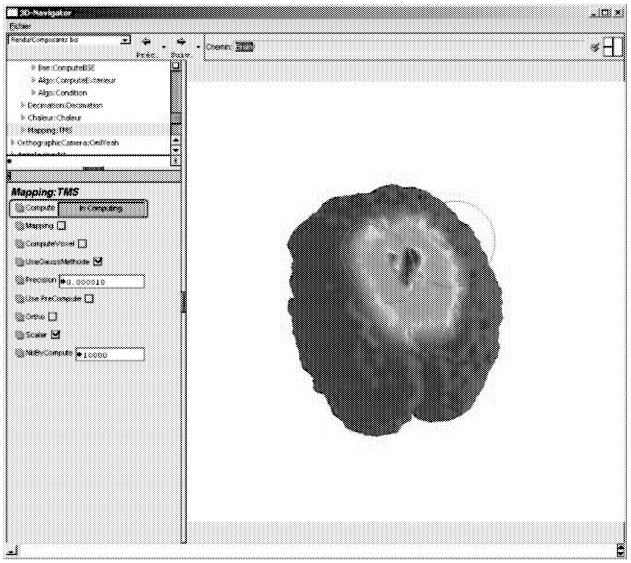


Figure 8: Use of the simulator with a reverse placement of the coil giving a better focalization.

MODEL DRIVEN ENGINEERING

Model Driven Engineering is a part of software engineering which studies the development, maintenance and evolution in an industrial context. MDE emphasises that software development is often code-centred and is the only artefact in the engineering domain which involved the use of adequate management tools. MDE relies on three fundamental elements: the 'model', the 'meta-model' and the 'transformation procedure'.

A model is a simplified representation of a system. The system is an entity modelled in order to understand, analyze and predict a resulting behaviour. The main objective of modelling is to answer questions emerging from the system under study. The relation 'a simplification of ' is one of the key element of modelling. It links the 'System Under Study' (SUS) and the resulting model. Similar definitions and complements can be found in [Hill 1996] [Atkinson and Kuhne 2003], [Seidewitz and Technologies, 2003] and [Bézivin, 2004]. Using this kind of definition, we are limited because the model resulting from this paradigm should be considered only again as another system and, consequently, it would be described again with a model. Using the notion of model is understandable only if the model is productive, i.e., if the model is exploitable by the computer and consequently enhances our knowledge of the system. In this logic, many authors characterize a model as a description of (part of) a system written in a well-defined language [Kleppe et al., 2003].

Introducing a well-defined language in the modeling procedure indirectly points the second fundamental relation of MDE. It deals with the relation 'is conform to' which is related to the concepts of model and meta-model. The meta-model is an entity which allows model description and the conformity to this description. For instance in the context of Object-Oriented Programming, the relationship 'is conform to' corresponds to the relation which links an instance and its class: an object is conform to its class, the class being itself a description of objects (a meta-object). This relationship 'is conform to' can be found in many other contexts. Since MDE focuses on models and especially on meta-models, it allows grouping under a same denomination a set of concepts present in a lot of technical domains such as:

- databases: a record of a relational database is conform to the database scheme,
- language theory and compilation: a source code is conform to a grammar,
- XML technologies: an XML file is conform to a DTD or XML scheme,
- object-oriented programming: an object is conform to its class.

The main force of MDE relies on this unifying power [Bezivin, 2005]. When a model is conform to its meta-model, it can be improved using a set of model modifications which allow transform the model. We are now dealing with this last MDE concept i.e, model transformations.

The 'transformation' concept is the third fundamental MDE concept and certainly the most significant, but also the one which is not consensual for a definition yet [Rahim and Mansoor, 2008], [Lano and Clark, 2008], [Iacob et al., 2008]. [Favre et al., 2006] give different definitions of the transformation concept. In this presentation we focus on a few examples in different domains:

- migration scripts in relational databases,
- compilation, code generation in language theory,
- XSLT transformation into XML.

In the case of code generation, the notion of transformation has a particular meaning, since this

transformation changes a model into another one with a lower abstraction level, closer to the underlying machine. Thus, the binary code (from assembly language) can be executed on the machine for which it has been compiled. We have in this case a productive model for a specific platform – a Platform Specific Model as named in MDA.

MODEL VERSIONING AND MODEL DRIVEN ENGINEERING

In this section we will propose a possible view of software versioning through a model driven process. To help understanding these concepts, we present a concrete case of the SVN software with the introduced notions of model, metamodel and transformation.

The emergence of Model Driven Engineering (MDE) at the beginning of the millenium was noticed with the proposal of the Model Driven Architecture (MDA) by the Object Management Group (OMG). As we previously stated, one of the main inputs of MDE is to say that software development was too much oriented towards source code. Whereas source code is very important, it is an artefact as many others in the process producing software applications. With an MDE approach, the whole source code of an application can be modeled just as another system, this model can be described and stored in regular files onto our hard disks. There is nothing new if we do not also define a metamodel, describing the software model with a conformity relationship and if we do not introduce the concept of model transformation.

In the context of versioning software like `svn`, this conformity relationship can be understood if we consider the folders regenerated by `svn export` and `svn checkout`. The first folder is a regular directory and the second contains a set of sub-directories with metadata allowing various requests and transformations on the software modeled. For instance, just to cite common requests, `svn status` and `svn diff` inform the developer about the working copy of the software by giving the current modifications. This kind of information permits to distinguish between a simple refactoring and a major change where the application behaviour has slightly changed.

The folder handled by versioning software can receive updates with `svn update` or `svn merge`. Such updates can be viewed as transformations in the MDE context. Each changeset is a kind of patch that will be applied to the model of the source code. In [Favre et al., 2006], we note that a transformation whose purpose is to correct a bug is not generic in current versioning systems, however once identified, a change set or transformation can be applied to a set of specified branches (version 1.9.x or 2.x for instance).

Unfortunately, the version management is currently orthogonal to the rest of the software development. Even if it is possible to develop without versioning tools, the productivity gain is such that the file comparison tool is more often called than the source code editor. In addition to the main view proposed by modern IDEs (Integrated Development Environment), versioning tools enable to add a view listing all the current modifications. Even though versioning tools are essential in modern development, we have encountered many problematic issues that will be discussed in the next section. With the modelling experience we had, we proposed a metamodel describing the main concepts in use for the management of software versions.

MODEL AND METAMODEL OF SOFTWARE VERSIONNING

In this section we will describe the model and metamodel we propose for the management of software versions. In this UML diagram (figure 9) we first have the **Software** class with three subclasses: **Target Software, Library** and **Version Management Software**. The two first classes contain many **Software Version Elements.** The **Version Management Software** is precisely handling such elements which are a key concept of this versioning model. This **Software Version Element** class is sub-classed in the two major elements encountered in version management software: the **Trunk** and **Branch** classes. The **Branch** class itself is sub-classed in three main subclasses being: the **Maintenance Branch, Feature Branch** and **Vendor Branch** classes.

On the left side of the UML diagram we have the classes implied in the dynamic aspects of software versioning. First, we have the **Change Set** class which contains all the **Software Changes** proposed by the **Developer**. The **Meta Change Set** contains all the metadata describing changes. This analysis helped us to have a better understanding of this part of software engineering which is hidden in various versioning software.

In the next section we see that a common problem arises in software development when using external libraries. It has been the case of the TMS software we developed. We relate the troubles generated by a bad use of branches that were developed to take into account different versions of libraries. The experience gained in this study is presented with a description of the problems we had when developing the simulation software. A set of possible solutions is also discussed and proposed in the next section.

PROBLEMS WITH THE VERSIONING OF LIBRARIES

When libraries need adaptation and that the wanted modifications cannot be produced by the library developers, we have to adapt the library to our specific use.

In our case, here is the concrete example: we had to use an old compiler (Code Warrior) and we could not abandon this choice since it would have implied too many modifications (handling of specific C/C++ constants or

macros…). For instance, in the WxWidget library, specific compilation cases between Cygwin, Visual and CodeWarrior were initially handled. But later the case of Codewarrior was abandoned and we had to maintain the backward compatibility for Codewarrior and thus to maintain our own version of the WxWidget library.

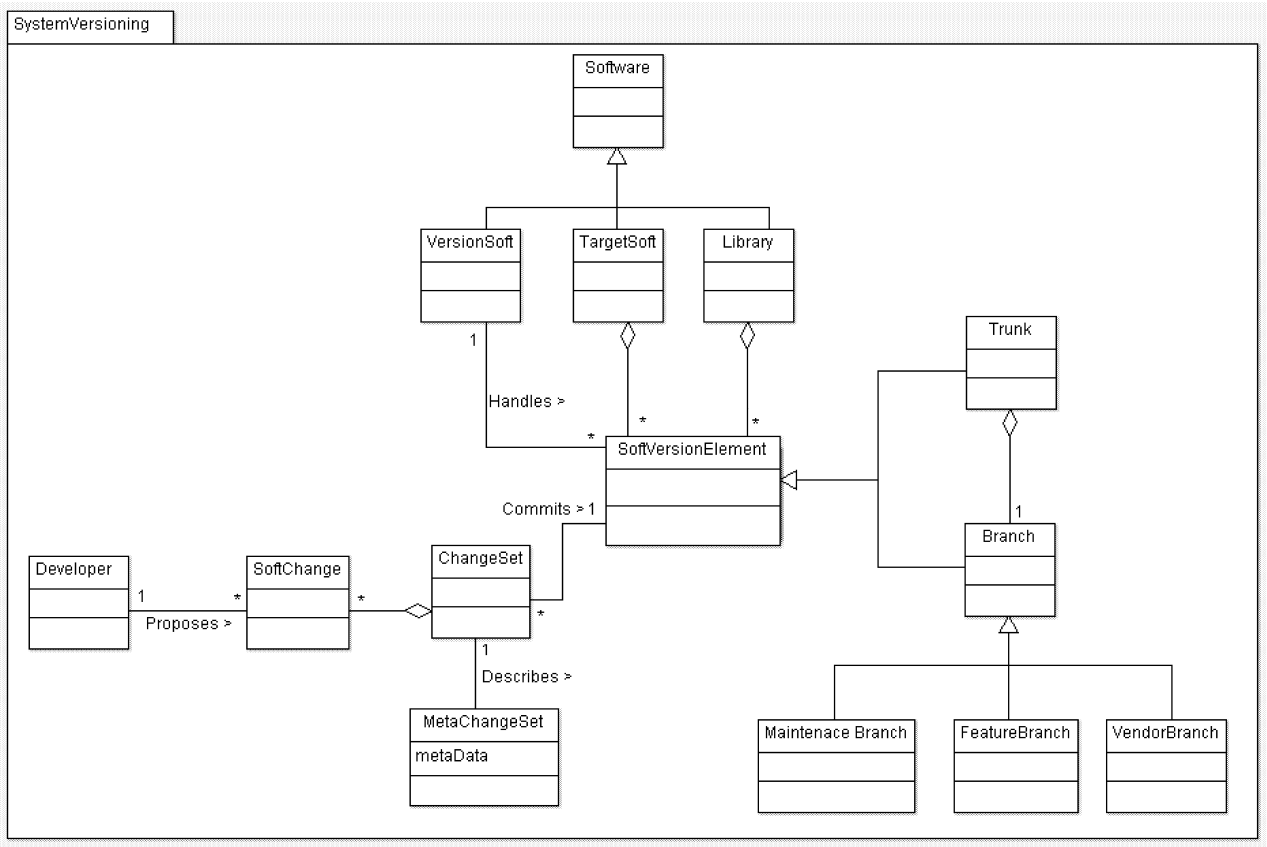


Figure 9: UML Model and metamodel of the concepts involved in version management software

In addition to this example, we can cite another case, when the internal development of the library proposed by the company is too fast, even for other internal projects; In this case, one has to keep a stable branch to enable the development of software pieces without having the daily problems induced by the constant modifications of internal libraries. To avoid perpetual modifications, we often create a software branch and we maintain the modifications between the trunk and the branch (both are committed / updated). When this occurs, it then becomes difficult to apply rigorously such modifications to avoid "spaghetti branches". With such examples, we see that the management of the different software versions – the software itself or the libraries it uses induces strong difficulties when we have frequent releases of libraries or when the new library version doesn't support some classes, methods or languages still in use in our application.

DISCUSSION AND POSSIBLE SOLUTIONS

In order to avoid the problems previously stated, such as spaghetti branches for instance, we propose to use the equivalent of a "rebase" in the **git** software. This "rebase" is not a feature of common version management software which often only proposes a merge of branches. The concept behind a "rebase" is simple: we have to copy a version of the trunk and to apply the modifications on this copy to avoid the spaghetti branches. If we continue the development, we will create / fork a new branch (by copying the previous modified trunk (Figure 10).

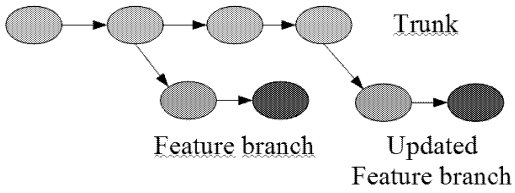


Figure 10: An example of rebase to add a new functionality (feature branch).

A feature branch is created to add a new functionality. Development on trunk goes ahead and the new feature needs some patches added to the trunk. A new feature branch will be created and all patches applied to the firstly created branch will be applied on the freshly created one. Development of the new feature will be carried on the new branch. The old one can be deleted.

In the context of library use, when we need to modify a library to adapt it to our specific usage, this implies the maintenance of our own library version. Dynamic languages, such as Ruby for instance, enable to apply a "monkey patch" on a method. We can open the class and overload the method to propose the wanted behaviour. This can be done only with inheritance if we use static typing languages. Monkey patching has side effects, implying difficulties in software maintenance and debugging. When we look for an error we are often not aware of the method version used by the software. If the method evolves or becomes deprecated in the library evolution, our software will continue to use the method version at the stage it was when we made the copy.

CONCLUSION

We have presented the former development of a friendly and reusable tool for the computation of magnetic field generated by any coil during TMS. This tool has been developed thanks to external and internal libraries and the purpose of this paper has been to show the problems we have encountered and the potential solutions. We have proposed a metamodel of the concepts in use in the management of different software versions. This model is used for a better understanding of software versioning and helped us in proposing possible solutions to avoid "spaghetti branches". A model driven engineering approach has been used to unify software versioning with the notion of model transformation. Version management software do indeed handle models of software to which they apply a set of changes producing another model, i.e. the new software version.

REFERENCES

Atkinson, C. and Kuhne, T., " Model-driven development: a metamodeling foundation". Software, IEEE, 2003, 20(5) :36–41.

Bézivin, J., "In Search of a Basic Principle for Model Driven Engineering". Novatica Journal, Special Issue, March-April, (2004), 2:21–24.

Bezivin, J., "On the unification power of models. Software and Systems Modeling", 2005, 4(2) :171–188.

Davey, N., Puri, B., Lewis, H., Lewis, S., and Ellaway, P., "Effects of antipsychotic medication on electromyographic responses to transcranial magnetic stimulation of the motor cortex in schizophrenia". J. Neurol. Neurosurg. Psychiatry., 1997, 63(4):468–73.

Favre, J., Estublier, J., and Blay-Fornarino, M., "L'ingéniérie dirigée par les modèles au delà du MDA.", Lavoisier, 2006.

George, M., Lisanby, S., and Sackeim, H., "Transcranial magnetic stimulation: Applications in neuropsychiatry." Arch. Gen. Psychiatry, 1991, 56(4):300–311.

H´edou, V. "Méthodes numériques pour la modélisation électro-anatomique du cerveau. Math´ematique et applications". Thèse de doctorat de 3ème cycle. Université de Rennes I, France, 1997.

Hill D.R.C., "Object-oriented Analysis and Simulation", Addison-Wesley, 1996, 291 pages.

Jalinous, R., "Guide to magnetic stimulation". The Magstim Company Limited, United Kingdom, 1998.

Kleppe, A., Bast, W., and Warmer, J. (2003). MDA Explained : The Model Driven Architecture : Practice and Promise. Addison-Wesley Professional.

Krasteva, V., Papazov, S., and Daskalov, I., "Magnetic stimulation for non-homogeneous biological structures". BioMedical Engineering OnLine, 2002, 1(1):3.

Lefaucheur, J., Drouot, X., and Nguyen, J., "Interventional neurophysiology for pain control: duration of pain relief following repetitive transcranial magnetic stimulation of the motor cortex". Neurophysiol. Clin., 2001, 31(4):247–52.

Luquet, S., Barra, V., and Lemaire, J., "Transcranial magnetic stimulation : Magnetic field computation using a parametrical coil model". OICMS, 1st Open International Conference on Modeling & Simulation Clermont-Ferrand 2005, pages 451–455.

Luquet, S., Barra, V., and Lemaire, J., "Transcranial Magnetic Stimulation : Magnetic Field Computation in empty free space. Engineering in Medicine and Biology Society", 2005. IEEE-EMBS 2005. 27th Annual International Conference of the, pages 4365–4368.

Nadeem, M., Thorlin, T., Gandhi, O., and Persson, M., "Computation of electric and magnetic stimulation in human head using the 3-d impedance method". IEEE Transactions on Biomedical Engineering, 2003, 50(7):900– 907.

Ruohonen, J., "Transcranial magnetic stimulation: Modelling and new techniques". Doctor of technology, Helsinki University of Technology, 1998.

Seidewitz, E. and Technologies, I. (2003). What models mean. Software, IEEE, 20(5) :26–32.

Thielscher, A. and Kammer, T., "Linking physics with physiology in TMS: A sphere field model to determine the cortical stimulation site in TMS.", Neuroimage, 2002, 17(3):1117–1130.

Ziemann, U., Steinhoff, B., Tergau, F., and Paulus, W. "Transcranial magnetic stimulation: its current role in epilepsy research". Epilepsy Res., 1998, 30(1):11–30.

Rahim, L. and Mansoor, S., " Proposed Design Notation for Model Transformation". In Software Engineering, 2008. ASWEC 2008. 19th Australian Conference on, pages 589–598.

Lano, K. and Clark, D., "Model Transformation Specification and Verification". In Quality Software, 2008. QSIC'08. The Eighth International Conference on, pages 45–54.

Iacob, M., Steen, M., and Heerink, L., "Reusable Model Transformation Patterns". In Workshop on Models and Model-driven Methods for Enterprise Computing (3M4EC) (2008), page 1.

APPLYING BCMP MULTI-CLASS QUEUEING NETWORKS FOR THE PERFORMANCE EVALUATION OF HIERARCHICAL AND MODULAR SOFTWARE SYSTEMS

S.Balsamo
Università Ca' Foscari di Venezia, Dipartimento di Informatica
Via Torino 155, Venezia
email: balsamo@dsi.unive.it

G. Dei Rossi
Università Ca' Foscari di Venezia, Dipartimento di Informatica
Via Torino 155, Venezia
email: deirossi@dsi.unive.it

A. Marin
Università Ca' Foscari di Venezia, Dipartimento di Informatica
Via Torino 155, Venezia
email: marin@dsi.unive.it

KEYWORDS
Performance evaluation, Software engineering, Queueing networks, Product-form solutions, BCMP

ABSTRACT

Queueing networks with multiple classes of customers play a fundamental role for evaluating the performance of both software and hardware architectures. The main strength of product-form models, in particular of BCMP queueing networks, is that they combine a flexible formalism with efficient analysis techniques and solution algorithms. In this paper we provide an algorithm that starting from a high-level description of a system, and from the definition of its components in terms of interacting sub-systems, computes a multiple-class and multiple-chain BCMP queueing network. We believe that the strength of this approach is twofold. First, the modeller deals with simplified models, which are defined in a modular and hierarchical way. Hence, we can carry on sensitivity analysis that may easily include structural changes (and not only on the time parameters). Second, maintaining the product-form property allows one to derive the average system performance indices very efficiently. The paper also discusses the application of the algorithm for the performance evaluation of Web Sites with modular architectures, such as those based on Content Management Systems.

INTRODUCTION

Performance analysis of modular and hierarchical systems has always been an important topic for the performance evaluation and software engineering research communities (see, e.g., Smith (1990)). In particular, a good approach to software design requires the definition of a modular and hierarchical architecture. From a high-level point of view, the software may be seen as the interaction of several black-box components. The definition of these sub-components follows the same approach in a hierarchical fashion until the very low-level layer of the architecture is reached. Performance evaluation of such models is important since the earlier stages of development as shown in Smith and Williams (2006). In this context, the main problem consists in the definition of efficient algorithms capable of deriving the required performance indices efficiently.

The class of models we consider in this paper is the well-known class of Markovian models. In particular we focus on those models whose underlying stochastic process is a Continuous Time Markov Chain (CTMC). Particular attention will be devoted to BCMP queueing networks introduced in Baskett et al. (1975), i.e., a class of queueing networks with separable solution and for which efficient analysis algorithms have been introduced for instance in Buzen (1973), Resiser and Lavenberg (1980), Bruell et al. (1984), Conway and Georganas (1986), Conway et al. (1989). One of the main features of BCMP queueing networks is the possibility of characterising the customers of the system by assigning them a class (temporary characterisation) and a chain (permanent characterisation). Under a set of assumptions, the class and the chain of a customer determines its probabilistic routing among the queueing stations and the service time distributions.

In this paper we propose a methodology, supported by a novel algorithm, which aims to simplify the performance evaluation of systems designed according to a hierarchical and modular architecture. This methodology is based on the definition of a high level model

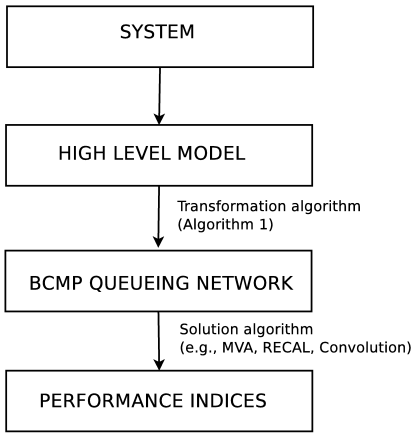


Figure 1: Sketch of the methodology proposed for performance evaluation of modular and hierarchical systems.

of the system consisting of several components. Each of these may be further specified in a hierarchical fashion. Under some assumptions that will be detailed later, we provide an algorithm which transforms this abstract model into a BCMP queueing network. Figure 1 illustrates the steps of this analysis. Let us consider an example. Modern Web Sites are often built on Content Management System (CMS) applications. CMSs are flexible re-programmable software systems consisting of a set of modules that are specialised in some task, e.g., rendering the web page, forum or wiki management, news and comments, user management. Modules are programmed by communities of developers who often work autonomously and must respect the interface given by the core system. Examples of modern CMSs are Drupal, Joomla!, PostNuke, Typo3 just to mention a few. Users who visit the web sites based on CMS are usually not aware of such a modular architecture. Nevertheless, a log analysis may reveal their behaviour among the site modules and clustering techniques may be adopted to distinguish user habits. We aim to provide a modelling approach that allows the system administrator to predict the performance of its web portal under different scenario from the knowledge of:

- the customer behaviours among the modules
- the resource requirements of each module
- the mapping of a required resource to a physical device.

This kind of analysis is not trivial and a hierarchical and modular approach should be adopted as observed for instance in Smith (1990). On the other hand we aim to provide a methodology that is compatible with known exact analysis algorithms to avoid the need of simulation or approximated technique as usually done in Woodside et al. (1995).

The main contribution of this paper consists in providing a methodology supported by an original algorithm that allows the modeller to specify the system in terms of a multiple-class and multiple-chain queueing network (QN) in which each station is itself a multiple-class and multiple-chain QN. The peculiarity of this hierarchical approach is that each station at a given level of abstraction is defined in isolation but, given two or more stations, they may share one of their components at a lower level of abstraction. In the example of the CMS one may think that at the top level of the CMS one has the routing of customers among the site modules (stations at the top level). Each module is then defined in terms of usage of resources (e.g., database, CPU, etc.). However, when these modules are combined one should be able to specify whether a resource is shared among different modules or the modules have distinct resources available. Obviously, this may have great impact on the overall performance of the system (e.g., are the DB and the multimedia resources stored in the same hard disk?). The goal of the algorithm that we introduce is to transform a QN defined at a top level into one defined at a lower level until the lowest level is reached. Once this is done, under some assumptions that will be described in the following sections, we obtain a product-form BCMP QN that may be analysed by the well-know algorithms for the computation of the average performance indices in steady-state.
The paper is structured as follows. First, we briefly recall the theoretical background on multiple-class BCMP queueing networks. Then, we describe the proposed methodology and we define of the algorithm. The last section provides an application example of the proposed approach. Some final remarks conclude the paper.

THEORETICAL BACKGROUND

This section aims to briefly recall the fundamental theorem on multiple-class product-form queueing networks, i.e., the BCMP theorem (Baskett et al. (1975)). Informally, we can say that it states sufficient conditions for a QN with multiple classes of customers to yield a product-form solution. Its importance is not only theoretical because several algorithms have been defined to compute the average performance indices in steady-state efficiently (e.g., Buzen (1973), Resiser and Lavenberg (1980), Conway and Georganas (1986)). This section first briefly illustrates the BCMP theorem and then lists the algorithms for the analysis with their computational complexity.

The BCMP theorem

BCMP queueing networks consist of a set of queueing centers and a (possibly infinite) set of customers. At a given epoch, each customer in the network has a class which may determine its routing probabilities or the ser-

vice time distribution at a given service station. When a customer changes its class we talk about *class switching*. Note that, in this paper, we use the concept of class in a local sense as in Chandy and Sauer (1980) rather in the global one used in Baskett et al. (1975). Classes form a temporary partition of the customers while chains are a permanent partition. Each class of customers belongs to a chain and routing may occur only within the same chain. Some conditions on the probabilistic routing must be assumed in order to ensure the ergodicity of the underlying process (see Balsamo and Marin (2007) for a recent survey). A chain may be open or closed. In the former case, customers arrive from the outside according to a Poisson process with a given rate, while in the latter the number of customers for that chain must be specified. The network is called *open* if all its chains are open, *closed* if they are all closed or *mixed* otherwise. Queueing stations must belong to one of the following types:

Type 1 : The queueing discipline is First Come First Served (FCFS) and the service time distribution is exponential and class-independent,

Type 2 : The queueing discipline is Processor Sharing (PS),

Type 3 : The station has infinite servers (IS), hence customers never waits in queue (*Delay Stations*),

Type 4 : The service discipline is Last Come First Served with Preemptive Resume (LCFSPR).

Stations of type 2, 3 or 4 may have a Coxian distributed service time that depends on the customer class. Moreover, the station service time may depend on the queue length at a given epoch (some non-strict conditions must be satisfied). This allows one to model important features such as the effect of multiple servers in the same station. Table 1 illustrates the notation we adopt and that we now briefly summarise. We use $\Omega = \{S_1, \ldots, S_M\}$ to denote the set of M queueing stations of the network, and let $\mathcal{R}_i$ be the set of classes served by station S_i, with R_i elements that are usually denoted by letters $r, s, \ldots$. Let $\mathcal{C} = \{1, \ldots, C\}$ be the set of labels for the C chains of the QN, then $\mathcal{R}_i^{(c)}$ is the set of classes served by station S_i and belonging to chain c (with $R_i^{(c)}$ elements), $1 \leq c \leq C$ and $1 \leq i \leq M$. Clearly, $\cup_{c=1}^{C} \mathcal{R}_i^{(c)} = \mathcal{R}_i$ for all i. The state-independent probabilistic routing is described by the probability matrix $\mathbf{P}^{(c)}$ for each chain c. Elements of $\mathbf{P}^{(c)}$ are $p_{ri,sj}^{(c)} \geq 0$, with $1 \leq i, j \leq M$ and $r \in \mathcal{R}_i^{(c)}$, $s \in \mathcal{R}_j^{(c)}$ and represent the probability of a customer entering station S_j with class s after being served in station S_i as class r. Label 0 represents the outside (hence matrix $\mathbf{P}$ has $1 + \sum_{i=1}^{M} R_i$ rows and columns). Sometimes, we have just one routing matrix $\mathbf{P}$ and we desire to derive the partition in $\mathbf{P}^{(c)}$, i.e., identify the chains in the QN. This can be reduced to the problem of identifying the ergodic sub-components in a Markov Chains and, since the structure of the network is usually rather small, the problem is known to be computationally tractable (see, e.g., Kant (1992), Balsamo and Marin (2007)). If c, $1 \leq c \leq C$, is a closed chain, then $K^{(c)}$ denotes the number of customers and $p_{ri,0}^{(c)} = p_{0,ri}^{(c)} = 0$ for all $r \in \mathcal{R}_i^{(c)}$ and $i = 1, \ldots, M$. If c is open $\lambda^{(c)}$ is the total arrival rate and matrix $\mathbf{P}^{(c)}$ is such that element $p_{0,ri}^{(c)}$ is the probability that a customer arriving from the outside enters station i as class r and element $p_{ri,0}^{(c)}$ is the probability for a customer to leave the system after being served at station i with class r.
Before briefly stating the BCMP theorem, we recall the definition of the QN traffic equations. For an open chain c, the system of traffic equations are:

$$\mathbf{e}_{ri}^{(c)} = \lambda_{ri}^{(c)} p_{0,ri}^{(c)} + \sum_{j=1}^{M} \sum_{s \in \mathcal{R}_j^{(c)}} e_{sj}^{(c)} p_{sj,ri}^{(c)} \tag{1}$$

for all $i = 1, \ldots, M$ and $r \in \mathcal{R}_i^{(c)}$. If c is a closed chain, the corresponding system of traffic equations is:

$$\mathbf{e}_{ri}^{(c)} = \sum_{j=1}^{M} \sum_{s \in \mathcal{R}_j^{(c)}} e_{sj}^{(c)} p_{sj,ri}^{(c)} \tag{2}$$

for all $i = 1, \ldots, M$ and $r \in \mathcal{R}_i^{(c)}$. In the latter case the system is under-determined, and the solution is defined up to an arbitrary non-null constant that has to be chosen. Solutions $e_{ri}^{(c)}$ of systems (1) and (2) represent the (relative) visit ration to station i, class r of chain c. Vector $\mathbf{e}_i = (e_{ri})$ with $r \in \mathcal{R}_i$ plays a pivotal role for the network steady-state solution. We can now state the salient result of the BCMP theorem given in Baskett et al. (1975).

Theorem 1 (BCMP (salient results)) *Let us consider a multiple-class and multiple-chain QN, open, closed or mixed, whose queueing stations are of type* 1, 2, 3 *or* 4. *Then, if the underlying stochastic process is ergodic, the steady-state probabilities are in product-form with respect to the queueing stations, i.e., let* $\mathbf{n} = (n_1, \ldots, n_M)$ *be the vector representing the state of the network, where component* n_i *is the state of station* S_i*, then the following relation holds:*

$$\pi(\mathbf{n}) = \frac{1}{G} \prod_{i=1}^{M} g_i(n_i), \tag{3}$$

where π *is the steady-state distribution of the QN, and* $g_i(n_i)$ *is the steady-state distribution of station* S_i *considered in isolation, with arrival rates* $\mathbf{e}_i$*, and* G *is a normalising constant.*

Ω	Set of queueing stations of the network
$\lambda^{(c)}$	Arrival rate to open chain c
$\mathcal{C}$	Set of the chain labels
C	Number of chains
$e_{ri}^{(c)}$	(Relative) arrival rate to station S_i, class r of chain c
$K^{(c)}$	Population of chain c
$\mathbf{e}_i$	Solution for the traffic equation systems (1) or (2) of station S_i
M	Number of queueing stations
$\mathbf{P}^{(c)}$	Routing probability matrix for chain c
$p_{ri,sj}^{(c)}$	probability for a customer to enter station S_j, class s, after being served at station S_i, class r, where both the classes s and r belong to chain c
$\mathcal{R}_i^{(c)}$	Set of classes of chain c served by station S_i
$\mathcal{R}_i$	Set of classes served by station S_i
$R_i^{(c)}$	Number of classes served by station S_i belonging to class c
R_i	Number of classes served by station S_i

Table 1: Table of the notation.

Solution algorithms

Theorem 1 and the class of BCMP networks have been widely applied for system performance analysis, because several efficient solution algorithms have been defined to compute the stationary state distribution π and a set of average performance indices. Such algorithms specifically apply to analyse closed or mixed networks, where we have to compute the normalising constant G, as stated by Theorem 1. Note that for open networks we have $G = 1$, the solution $e_{ri}^{(c)}$ of the traffic equations (1) already gives the throughputs of each node S_i for classes r in chain c. Then one can easily derive the other average performance indices by classical queueing system results.

Various solution algorithms have been defined for closed and mixed BCMP networks. Some algorithms, such as the Convolution Algorithm, directly compute the normalising constant G in equation (3) and hence a set of mean performance indices, such as the mean response time, the average queue length, and the throughput of each queueing station. Other algorithm, such as MVA (Mean Value Analysis) avoid the computation of the normalising constant G and iteratively (over the number of customers) directly compute a set of average performance indices. For multiple-class and multiple-chain BCMP networks some algorithms apply special recursive scheme on the number of chains, and/or take advantage of the possible sparsity of the chains (e.g., chains that contain few classes) to derive efficient solution. Although it is out of the scope of this paper to describe these well-known algorithms, we just cite them and recall their computational complexity. Several tools for the analysis of queuing networks have been implemented over the last decades. A recent work, called qnetworks toolbox, is described in Marzolla (2010). Such an implementation of several algorithms is given in terms of library of functions for Octave, i.e., a programmable environment for numerical computation. This allows one to integrate easily the algorithms of qnetworks with new ones, for instance that presented by Algorithm 1. Hereafter, we consider a queueing network with multiple chains but where each station has just one class per chain (single-class, multiple-chain QN). One can show that for each multiple-class and multiple-chain BCMP QN it is possible to define another BCMP QN with single-class and multiple-chain with the same average performance indices (see, e.g., Kant (1992)). For the sake of clarity, we consider the QN consisting of only closed chains.

The Convolution Algorithm computes the normalising constant from which the average performance indices may be derived. The computational complexity, given the solution of the traffic equations system (2), depends on the type of stations in the QN. In particular each iteration has a cost of $\mathcal{O}(CH)$ for load-independent stations and of $\mathcal{O}(H^2)$ for the others, where $H = \prod_{c=1}^{C}(K^{(c)}+1)$. If all the chains has the same population κ and no load-dependent stations are present, then the computational cost is $\mathcal{O}(MC\kappa^C)$.

The Mean Value Analysis algorithm (MVA) is based on the *Arrival theorem* that provides an efficient recursive scheme to compute the steady-state average performance indices. For a QN without load-dependent stations, and with identical chain populations, its complexity is identical to that of the Convolution.

The Recursion by Chain Algorithm (RECAL), defined in Conway and Georganas (1986), computes the normalising constant and, in a similar fashion of Convolution, from this it derives the average performance indices. It is particularly interesting because despite of a greater complexity in the implementation, its computational complexity grows in a polynomial way with the number of chains, i.e., for high number of chains,

$\mathcal{O}(C^{M+1})$. RECAL has been improved from its original definition in several ways and is now widely applied for the solution of QNs with high number of chains.
Note that several other algorithms for the exact or approximate computation of the average performance indices in multiple-chain BCMP QNs have been defined in literature. A survey may be found in Balsamo and Marin (2007).

FRAMEWORK DESCRIPTION AND ALGORITHM DEFINITION

In this section we first illustrate how to describe a model in our framework, and then we present the algorithm to obtain the underlying BCMP QN. Once this is derived one of the algorithms presented in the previous section may be applied in order to obtain the desired performance indices.

Model description

As we pointed out in the introduction we aim to provide a framework for the specification of software and hardware architectures which enhances the modularity and hierarchical features. In this setting, we see a system, at its highest level of abstraction, as consisting of a set of components $d_1, d_2, \ldots, d_{\ell_1}$. The easiest way to interpret the model specification is seeing these components as the queueing stations of a multiple-class and multiple-chain QN. Hence, probabilistic routing and customer characterisations are allowed. Each of the components d_i, with $d_i = d_1, \ldots, d_{\ell_1}$, seen at the highest level of abstraction, may be defined as:

- A BCMP queueing station
- A sub-model consisting of components $d_{(i)1}, \ldots, d_{(i)\ell_2}$. Note that it is *not* the case that $d_{(i)k} \neq d_j$ for all $1 \leq j \leq \ell_1$ and $1 \leq k \leq \ell_2$, i.e., a component may use a resource which has already been described at a higher level. These components interact as stations of an open multiple-class and multiple-chain QN. Each sub-model from the outside can be seen as a black box, with a set of access points with some labels, i.e., the classes of the customers arriving from the outside (input classes) and the classes of the customers leaving the sub-model (output classes). We require that the set of input classes must be equal to the set of output classes of each component.

This recursive definition is the basis of the algorithm that follows.
We now introduce the concept of *well-formed* model.

Definition 1 (Well-formed models) *Given a model consisting of m components (in any level of abstraction) then we define the binary relation $\succ$ as follows:*

- *$d \succ d'$ if and only if d' appear in the definition of d*

and the binary relation $>$ as follows:

- *$d \succ d' \Rightarrow d > d'$ or*
- *$d > d'$ if there exists d'' such that $d \succ d''$ and $d'' > d'$.*

A model is well-formed if and only if relation $>$ is a strict partial order.

Roughly speaking, a well-formed model does not have cycles in the definition of the components. However, it is possible, at a given level of abstraction, to refer to components specified at higher levels. Hereafter, we consider only well-formed models.

Algorithm definition

In order to better understand our approach to modular BCMP network design, we will first consider the algorithm, then we show how it could be further optimized.
Let $\mathcal{D}$ be the set of the m components $d_1 \ldots d_m$ that form the model, and $\mathcal{R}_i$ be the set of the n classes $r_{i,1} \ldots r_{i,n}$ for the component d_i. In each component, we call EI (external input) the arrival streams from the outside, and EO (external output) the departure streams.
Binary relation $d \succ d'$ given by Definition 1, means that d *contains* d', i.e., if d is not a simple QN station, then d' appears in the routing matrix of d.
Let $\mathbf{P}_d$ be the routing matrix of component d and let be $\mathbf{P}_{d'}$, with $d \succ d'$, the routing matrix of a subcomponent of d' of d, then we aim to unfold d' in order to specify a routing matrix for its subcomponents.
If $d_i \succ d_j$, let $A_{d_i,d_j} : \mathcal{R}_i \to \mathcal{R}_j$ be a partial function that associates class names of component d_i with class names of component d_j. Notice that A is not necessarily injective, i.e., two or more classes of the container component can be mapped into the same class of the contained component. Whenever it happens, we should add a new class set to the submodel.
$A_{i,j}$ functions must be given by the user of the algorithm, and define how classes of a high level component maps on classes of its subcomponents. This allows the design of a low level component without knowledge of its future use in a high level one.
In Algorithm 1 we show a method for routing matrix unfolding. To keep the code simple, we use a single routing matrix that combines every chain of the component, but the algorithm preserves chains, i.e. (node,class) pairs that are in different chains in high-level model description remain in different chains in the solution computed by the algorithm.
Notice that here the insertion and replacement operators for rows and columns have a loose semantics, i.e., whenever a row or a column with less elements is assigned to

Input: routing matrix $\mathbf{P}_d$ of component d, component counter array Dc, functions $A_{i,j}$
Output: unfolded routing matrix $\mathbf{P'}_d$ of component d
if *d is a station* **then** `/* base case */`

```
    foreach class r of d do
        insert in P' rows and a columns for EI_d,r and EO_d,r of P
    end
else
    foreach d_i | d ≻ d_i do
        Dc_i ← Dc_i + 1
        Let Rc_i be a class counter array
        foreach class r_k of d_i as named in d do
            r_i,j ← A_d,d_i(r_k)
            Rc_i,j ← Rc_i,j + 1
            if Rc_i,j > max Rc_i then
                U = UnfoldComponents(P_d_i)
                rename each class r_i,j of d_i in U as r_i,j,Dc_i,Rc_i,j
                insert in P' rows and columns of U
            end
            replace column EI_d_i, r_i,j,Dc_i,Rc_i,j in P' with column d_i, r_k of P
            replace row EO_d_i, r_i,j,Dc_i,Rc_i,j in P' with row d_i, r_k of P
        end
    end
end
return P'
```

Algorithm 1: Algorithm *UnfoldComponents* to derive a BCMP QN from a model specified at a higher level of abstraction.

a greater one, all elements that are not indexed in the smaller vector are set to 0.
The main idea of the algorithm is to distinguish the use of the same component class in different incoming and outgoing path, creating a new class whenever the component or the class itself is reused more than once. In order to achieve this, we use a global component usage counter and a local one for class usage. The algorithm then recursively expand the hierarchical model in a top-down fashion, until it reaches a standard BCMP station, i.e., a sub-model that has no components.

Comments on the algorithm. It is possible to show that starting from a high-level model whose routing is specified correctly, i.e., none of the classes become empty or its population grows indefinitely with probability 1 in the long run, we obtain a valid BCMP QN. Hence, under stability assumptions, the steady-state distribution and the average performance indices may be derived. The strength of the algorithm is that parametric analysis are simplified with respect to using the BCMP model directly. In fact, it suffices to change the labels associated with some resources to generate a totally different

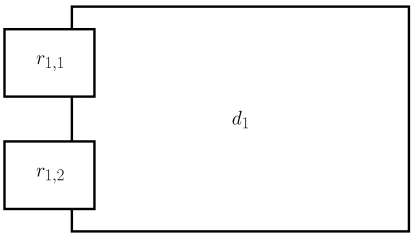


Figure 2: A black-box model of a database-indexed file archive CMS module.

routing. In the former example of the CMS, the administrator may be able to predict the performance measures of its system in case of a new server quite easily by mapping which server modules will be run in the new server at the highest level of abstraction.
The algorithm could be optimized, without changing its behaviour, saving partial computations of the $\mathbf{U}$ matrix before the execution of the innermost foreach cicle and renaming, at each iteration, all classes accordingly.
Under the assumption that, on average, every component has the same number of sub-components d, every component that is not a BCMP queueing station has the same number of classes r and the depth of the model, i.e., the length of the longest chain $d_1 \succ \ldots \succ d_k$, is n, we estimate that, in the worst case, the algorithm complexity is $\mathcal{O}(rd^n)$.

Illustrating example

We now provide an example that aims to illustrate the modelling methodology and the application of Algorithm 1 to a case-study. For the sake of brevity we keep the modelled system rather simple even if, obviously, the algorithm usefulness is enhanced by larger systems.
System description. We consider just a single module of a CMS, i.e., a database-indexed file archive. This is a typical feature of many websites that provide downloadable resources (e.g., multimedia or documents). Suppose that this module serves two classes of customers, one which models the file upload, and the other the file download. A download request passes through the database and then accesses the disk. An upload request, passes through the database and the accesses the disk to be written. If the operation is succesfull then a message for the database is generated to confirm the correct operation.
Model definition. The black-box model of the CMS module is shown in Figure 2. We can see the two incoming and outgoing classes of customers. Let this component name be d_1. Figure 3 represents the internal structure of d_1, where d_2 is the database component and d_3 the disk component. Hence, we clearly have $d_1 \succ d_2$, $d_1 \succ d_3$ and $A_{1,2}(r_{1,1}) = r_{2,1}, A_{1,2}(r_{1,2}) = r_{2,1}, A_{1,2}(r_{1,3}) = r_{2,2}, A_{1,3}(r_{1,1}) = r_{3,1}, A_{1,3}(r_{1,2}) =$

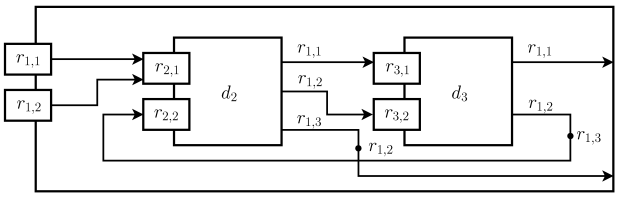


Figure 3: The internal definition of the module of Figure 2.

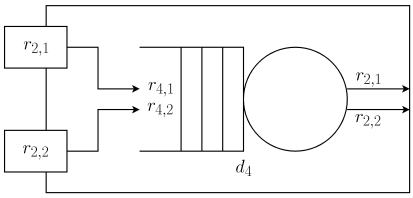


Figure 4: Minimal DB module design.

$r_{3,2}$. The routing matrix $\mathbf{P}_1$ is, ignoring impossible (component,class) combinations, a 9×9 square matrix.

Let us suppose, for the sake of brevity, that component d_2, as in Figure 4, is made of a single component, d_4, that is a multiple-class station, and that $A_{2,4}(r_{2,1}) = r_{4,1}, A_{2,4}(r_{2,2}) = r_{4,2}$. Then we show how an application of the algorithm to d_1 transforms d_2. Note that we limit the observation to this part of the system for the sake of readability, since the number of classes arising from this simple example may be difficult to represent graphically.

The algorithm invokes recursively the UnfoldComponents function of Algorithm 1 until it finds a BCMP queueing station, in this case d_4, then, assuming it is encountered for the first time, it renames its classes $r_{4,1}$ and $r_{4,2}$ in $r_{4,1,1,1}$ and $r_{4,2,1,1}$. At the end of the invocation of UnfoldComponents($\mathbf{P}_{d_2}$), the resulting routing matrix is like in Table 2. Notice that rows $EO_{r_{2,1}}$ and $EO_{r_{2,2}}$ are both zero, because they represents departures from the component, that have yet to be connected to a higher level sub-model.

During the invocation of UnfoldComponents($\mathbf{P}_{d_1}$) both $A(r_{1,1})$ and $A(r_{1,2})$ return $r_{2,1}$, and therefore $Rc_{2,1}$ is incremented twice. The algorithm, then, inserts the elements of $\mathbf{P}_{d_2}$ twice in $\mathbf{P}'$, with different class names, e.g., $r_{4,1,2,1}$ and $r_{2,1,1,2}$.

As previously stated, the usefulness of the algorithm become more noticeable when the model is complex, e.g., if the Database module, instead of being made of a single queueing station, was described in terms of interacting submodels, like a CPU, one or more disks, a caching system, et cetera. All this subsystems may be also used by other modules.

CONCLUSION

This paper addresses the problem of combining a modular and hierarchical modelling technique with an efficient analysis method. The main theoretical contribution is an algorithm which allows the transformation of models defined at a higher level of abstraction into models defined at a lower one. A recursive application of this algorithm produces, under a set of conditions, a product-form multiple-class and multiple-chain BCMP QN. Although to obtain this we must limit the formalism expressivity, e.g., fork and join constructs are not permitted, our aim is to provide a methodology supported by efficient algorithms for the exact analysis. Other more expressive hierarchical approaches, such as those defined in Woodside et al. (1995), may require approximate algorithms for deriving the average performance indices in equilibrium. Future works have several directions. From a theoretical point of view, an extension of the class of models tractable by such a formalism would be important. Another important aspect is the integration of this framework with well-known and recent advances in web mining, particularly in log analysis. These techniques provide a partition of application users according to their behaviour. Although these techniques have been traditionally applied to predict the customer interests (e.g., for proposing context aware advertisements) we claim they may be very useful also for performance evaluation purposes.

REFERENCES

Balsamo S. and Marin A., 2007. *Queueing Networks in Formal methods for performance evaluation*, M. Bernardo and J. Hillston (Eds), LNCS, Springer, chap. 2. 34–82.

Baskett F.; Chandy K.M.; Muntz R.R.; and Palacios F.G., 1975. *Open, Closed, and Mixed Networks of Queues with Different Classes of Customers. J ACM*, 22, no. 2, 248–260.

Bruell S.C.; Balbo G.; and Afshari P.V., 1984. *Mean Value Analysis of Mixed, Multiple Class BCMP Networks with Load Dependent Service Stations. Perform Eval Elsevier*, 4, 241–260.

Buzen J.P., 1973. *Computational algorithms for closed queueing networks with exponential servers. Commun ACM*, 16, no. 9, 527–531.

Chandy K.M. and Sauer C.H., 1980. *Computational algorithms for product form queueing networks. Commun ACM*, 23, no. 10, 573–583.

Conway A.E.; de Souza e Silva E.; and Lavenberg S.S., 1989. *Mean Value Analysis by Chain of Product form Queueing Networks. IEEE Trans Comput*, 38, no. 3, 432–442.

$$\mathbf{P}' = \left[\begin{array}{c|cccccc} & EI, r_{2,1} & EI, r_{2,2} & EO, r_{2,1} & EO, r_{2,2} & d_4, r_{4,1,1,1} & d_4, r_{4,2,1,1} \\ \hline EI, r_{2,1} & 0 & 0 & 0 & 0 & 1 & 0 \\ EI, r_{2,2} & 0 & 0 & 0 & 0 & 0 & 1 \\ EO, r_{2,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ EO, r_{2,2} & 0 & 0 & 0 & 0 & 0 & 0 \\ d_4, r_{4,1,1,1} & 0 & 0 & 1 & 0 & 0 & 0 \\ d_4, r_{4,2,1,1} & 0 & 0 & 0 & 1 & 0 & 0 \end{array}\right]$$

Table 2: Routing table $\mathbf{P}'$ at the end of UnfoldComponents($\mathbf{P}_{d_2}$).

Conway A.E. and Georganas N.D., 1986. *RECAL - a new efficient algorithm for the exact analysis of multiple-chain closed queuing networks. J ACM*, 33, no. 4, 768–791.

Kant K., 1992. *Introduction to Computer System Performance Evaluation.* McGraw-Hill.

Marzolla M., 2010. *The qnetworks Toolbox: A Software Package for Queueing Networks Analysis.* In *Proc of Int. Conf. ASMTA.* Cardiff, UK, *LNCS*, vol. 6148, 102–116.

Resiser M. and Lavenberg S.S., 1980. *Mean Value Analysis of Closed Multichain Queueing Network. J ACM*, 27, no. 2, 313–320.

Smith C.U., 1990. *Performance Engineering of Software Systems.* Addison-Wesley.

Smith C.U. and Williams L.G., 2006. *Five steps to establish software performance engineering.* In *CMG Conf.* Reno, Nevada, USA, 507–516.

Woodside C.; Neilson J.; Petriu S.; and Mjumdar S., 1995. *The Stochastic rendezvous network model for performance of synchronous client-server-like distributed software. IEEE Transaction on Computer*, 44, 20–34.

BIOGRAPHY

SIMONETTA BALSAMO is a full professor of Computer Science at the University Ca' Foscari of Venice, Italy. Her research interests include performance and reliability modeling and analysis of computer and communication systems, parallel and distributed processing, distributed simulation, quantitative analysis of software architectures and integration of specification languages and performance models. She has published several papers in international journals, conference proceedings, books and special editions. has servedd as general chair, program chair and program committee member for several internatinal conferences, is associated editor of Performance Evaluation Journal.

GIAN-LUCA DEI ROSSI received his M.Sc. degree in Computer Science from the University of Venice in 2010. He is currently a Ph.D. student at the same University. His research area is on stochastic models and performance evaluation.

ANDREA MARIN received his degree in Computer Science from the University of Venice in 2002, and the Ph.D. in Computer Science from the same University in 2009. Most of his research is focused on the analysis of compositionality of Markovian stochastic models, in particular in stochastic Petri net domain. Some novel results about product-form solutions have been derived in this context. He now works as a post-doctoral researcher at the University of Venice

EXECUTION-DRIVEN SIMULATION OF NON-FUNCTIONAL PROPERTIES OF SOFTWARE

Antti P. Miettinen
Nokia Research Center
email: antti.p.miettinen@nokia.com

Vesa Hirvisalo
Aalto University
email: vesa.hirvisalo@tkk.fi

Jussi Knuuttila
Aalto University
email: jussi.knuuttila@tkk.fi

KEYWORDS
Simulation, Software Development, Performance Engineering

ABSTRACT

We present our evaluation of a simulation mechanism for understanding non-functional properties of computer software. Considering the prevalence of computer-based appliances, such as mobile hand-held devices, it is important for software to have high performance and energy-efficiency. Understanding the timing of software execution is essential for estimating such non-functional properties of software.
The traditional way of simulating the timing behavior is based on cycle-accurate computer hardware simulators. Such simulators are orders of magnitude slower than real hardware, and thus unsuitable for software developers.
Our simulation methodology is based on execution-driven simulation. We use a novel approach to bind simulation models together with dynamic binary translation of target software. Our experimentation with a production-quality simulator, QEMU, shows that reasonable simulation times and accuracy can be achieved by using this method.

INTRODUCTION

The non-functional properties of software are usually addressed relatively late in the product development cycle. The focus of the conventional edit-compile-debug cycle of software development is the functionality of the software. Addressing performance and power consumption is especially challenging in cross-development setups typical for software targeting mobile hand-held devices and embedded systems.
Cross-development environments often employ functional simulators for interactive development and debugging of software. High simulation speed is essential in these setups as rapid feedback is critical for the productivity of the software development. However, the use of functional simulators makes it very challenging to optimize the non-functional properties of software. In fact, relying on simulation time on the development host as an indication of performance of the software on a real target device can be severely misleading.
In order to effectively provide feedback about the non-functional properties of software during the natural development cycle, the simulators used for software development should be able to estimate the timing of software execution. However, traditional timed simulation techniques are unsuitable for interactive software development and debugging because of the high overhead of such simulation.
In our approach, we have enhanced a fast functional simulator with parametrized models that enable estimating the timing of software execution with varying degrees of accuracy. The parametrized models operate at a sufficiently high level of abstraction to allow maintaining high simulation speed. Parametrization allows calibrating the models with measurements from real hardware platforms or cycle accurate simulators to provide timing estimates sufficiently accurate to guide developer design decisions during early phases of software development. The approach is flexible as it allows a trade off between accuracy and simulation speed and thus enables the developer to switch between a faster, less accurate simulation, and a slower, more accurate simulation as required.
The rest of the paper is organized as follows. In the following section we review the state of the art of the related research. We then describe our simulation methodology, followed by a section providing details of our experimental setting and a section summarizing our results. We then discuss the results and finally end our paper with our conclusions.

STATE OF THE ART

Understanding the timing of the execution of software is important, even for non-real-time applications. Many other non-functional properties can be estimated if the timing of the execution is known (Miettinen and Hirvisalo 2009). Importantly for many modern appliances, such as cell phones, laptops, etc., such derived properties include power and energy consumed.
The traditional way of simulating the timing behavior is

based on instruction set simulators (Burger and Austin 1997). They are used by computer hardware developers to understand the design space, act as reference models, aids to verification, etc. Computer software developers also use instruction set simulators, but much faster ones with very limited simulation capabilities. Such simulators are called functional simulators or emulators.

Given these widely different applications, instruction set simulators have a wide spectrum of capabilities and underlying implementation techniques. Accuracy of simulation models range from very accurate gate-level simulators (with accurate timing information) to functional simulators without any model of the computer hardware micro-architecture (with no timing information).

Similarly, the speed of simulation has a wide range. Proper cycle accurate simulators are typically at least 4 orders of magnitude slower than the actual hardware, whereas purely functional simulators can almost reach the speed of the actual hardware (Topham and Jones 2007, Weber et al. 2004).

The slowness of cycle-accurate simulators makes them unsuitable for software development, because reasonably sized programs cannot be run on them using the edit-compile-debug cycles typical for software development. Purely functional simulators are fast enough for software development, but provide no information on non-functional properties of software.

Recently, this has led to the development of cycle-approximate instruction set simulators (Franke 2008). Cycle-approximate simulators use approximate timing models that can provide the software developer with accurate-enough information on the timing properties of the software under development.

Fast functional simulators are typically based on just-in-time (JIT) compilation techniques (Aycock 2003). JIT compilers are dynamic, i.e., software is translated from the source language to the target language along with the execution of the target language code. Functional instruction set simulators typically apply a specific form of JIT, binary translation (Sites et al. 1993), where the the binary code of the simulation target is translated into the binary code of the simulation host.

Considering the development of software for mobile hand-held devices, using functional simulation is the state-of-the-art practice. Using the devices themselves as development hosts is not practical because of the small size and limited resources of the devices. Instead, software developers use cross-development environments on separate development hosts. A functional simulator is an essential component of any such environment to allow the software to be tested. Typical development hosts are based on the Intel x86 family of processors and typical mobile hand-held devices are based on the ARM family of processors (ARM 2010). As the processor families differ significantly, fast functional simulation is nontrivial.

Memory access related behavior is often critical for the performance of software. Simulation of the memory hierarchy even without detailed CPU core modelling can yield surprisingly good timing estimates (Weaver and McKee 2008). Traditional cache simulation (Edler and Hill 1998) is based on explicit simulation of the memory traffic and maintaining accurate information about the state of the memory system. The overhead of such simulation can be significant. Analytical cache models (Gecsei et al. 1970) require knowledge of the memory access distribution and are therefore even more costly to evaluate during execution driven simulation.

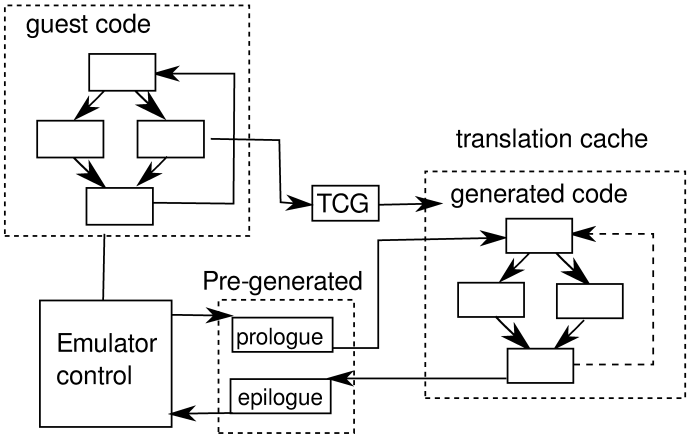


Figure 1: QEMU binary translation

EXECUTION-DRIVEN SIMULATION

For functional simulation, we use QEMU (QEMU 2010), an open source machine emulator that uses dynamic binary translation. QEMU is able to simulate both single guest applications and entire guest operating systems while itself running on a different computer architecture.

The operation of QEMU is based on dynamic binary translation, in which the machine code of the guest program is translated into functionally equivalent machine code for the host computer at run time. Compared to interpretation, this translation is likely to be slower. However, programs typically spend most of their execution time executing loops. By caching and chaining translated code (Bellard 2005), QEMU seeks to take advantage of this characteristic and achieve execution times comparable to actual hardware.

The translation process is done in two phases (see Figure 1). In the first phase, a guest architecture specific front-end decodes the guest instructions and translates them into Tiny-Code-Generator operations (TCG ops), an intermediate representation. The TCG ops are then translated into native host instructions using a host architecture specific back-end.

We have modified QEMU to make it suitable for timing measurements (Hirvisalo et al. 2010). Our modified ver-

sion, called pQEMU, is currently based on two separate instrumentations:

- An instrumentation to classify and count executed guest instructions. These counts can then be used to estimate execution times.
- An instrumentation of guest memory accesses to count the amounts of memory accesses and cache and translation look-aside buffer (TLB) misses using run-time cache and TLB simulations.

pQEMU is currently under development at Aalto University.

Counting each instruction type individually would be problematic because of the large amount of different instructions. To avoid doing so, we exploit the fact that similar instructions are likely to have similar execution times. The instructions are classified based on their memory and branching characteristics and the data they operate upon. Instructions belonging to the same class are counted together.

To implement instruction counting, we have modified the front end of the binary translator. Each instruction is classified during its translation and an additional counter incrementing function call is emitted as part of its translation. Since the translations of the performance-critical parts of the guest code are typically executed many times, the cost of this classification is amortized over the execution of the guest program. As instruction classification is highly processor-specific, we have currently only implemented it for the ARM instruction set architecture.

For memory accessing instructions, this approach alone is not enough to obtain reasonably accurate results because the execution times of those instructions depend heavily on whether the accessed locations can be found from the cache and the TLB. We have therefore also implemented a separate instrumentation of memory accesses. During translation, for each explicit or implicit (i.e. instruction fetch) memory access in the guest program, an extra function call is emitted. The called function uses simple cache and TLB simulators to determine if the memory access causes a cache or TLB miss and then increments the corresponding counters.

All this instrumentation comes with a cost. When all instrumentations are enabled, each executed instruction has to increment counters and simulate cache and TLB behavior for the instruction fetch. If the instruction itself accesses memory, cache and TLB behavior must be simulated for the accessed location as well. The extra processing and particularly any host CPU cache misses caused by the instrumentation significantly slow down the simulation compared to unmodified QEMU. Regardless, the performance benefit of using dynamic translation is so large that this slowdown is still small compared to using cycle-accurate simulation. Additionally, we anticipate that in the future, the instrumentations can be optimized to significantly recover performance.

Table 1: Test platforms

Platform	CPU	Speed	Cache (L1, L2)
PB11MPCore	ARM11	210MHz	32k+32k, 1M
NaviEngine NE1	ARM11	400MHz	32k+32k, none
Beagleboard C3	Cortex-A8	500MHz	16k+16k, 256k
Beagleboard C4	Cortex-A8	720MHz	16k+16k, 256k
KZM CA9	Cortex-A9	500MHz	32k+32k, 512k
Tegra	Cortex-A9	1GHz	32k+32k, 1M

EXPERIMENTAL SETTING

Our timing estimation is based on statistics extracted from the simulation and calibration of the statistics against measurements from the following hardware platforms:

PB11MPCore
The Platform Baseboard for ARM11MPCore is an evaluation platform from ARM Ltd. employing a four core ARM11MPCore processor.

NaviEngine NE1
NEC NaviEngine NE1 test board from NEC Electronics uses a four core ARM11MPCore System-on-Chip targeted for automotive applications.

Beagleboard C3
The Beagleboard revision C3 is a single board computer based on OMAP3530 processor employing an ARM Cortex-A8 core running at 500MHz.

Beagleboard C4
The Beagleboard C4 is a newer revision of the board running at 720MHz.

KZM CA9
KZM-CA9-01 is an ARM Cortex-A9 test chip board from Kyoto Microcomputer Co., Ltd. employing a four core ARM Cortex-A9 processor.

Tegra
The NVIDIA Tegra 250 Harmony development kit is an evaluation board employing a dual core ARM Cortex-A9 processor.

Central characteristics of the platforms are listed in Table 1.

The workload set used for calibration and evaluation purposes should be representative of the applications typically run on mobile devices. However, the resources and capabilities of our test platforms limit the selection. Practical considerations are also repeatability of tests and reasonable simulation times. Our application workloads use the following set of applications from Ubuntu Linux 9.10 distribution:

faac Freeware Advanced Audio Coder (AAC encoder)

flac Free Lossless Audio Codec (audio encoder)

twolame MPEG Audio Layer 2 (MP2) encoder

toast/untoast GSM speech compressor/decompressor

mcrypt Encryption tool with multiple algorithms

ccrypt AES encryption tool

lzma Lempel-Ziv/Markov-chain compressor

gzip/minigzip Deflate compression tools

bzip2 Burrows-Wheeler block-sorting compression tool

cjpeg/djpeg JPEG encoder/decoder

x264 H.264 video encoder

ffmpeg Video converter

mencoder Video encoder

html2text HTML to text converter

pdftotext PDF to text converter

For exercising specific features of the memory hierarchy we have also constructed some micro-benchmarks:

ictrasher
: Pseudo random branching test intended to cause instruction cache misses according to given parameters.

propread
: Memory read test performing local and non-local memory reads with given proportions.

propwrite
: Memory write test performing local and non-local memory writes with given proportions.

randread
: Memory read test accessing memory in pseudo random order.

randwrite
: Memory write test accessing memory in pseudo random order.

Our instrumentations gather the following statistics:

class
: Instruction class, one of: data processing, load, store, branch, media processing or coprocessor access.

L1 cache
: Instruction cache misses and data cache misses.

L2 cache
: Misses in a unified level two cache.

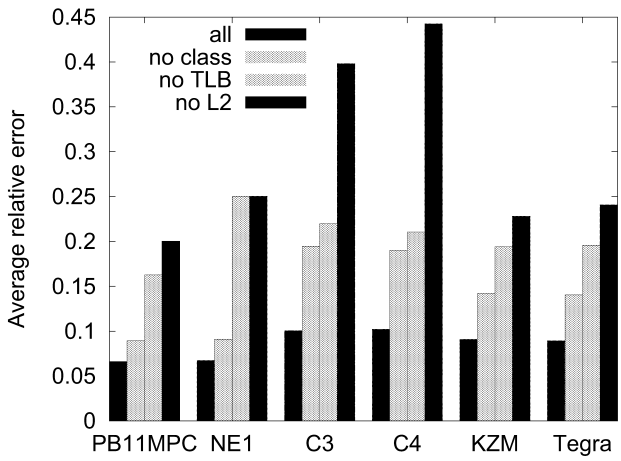


Figure 2: Estimation error with varying simulation detail

TLB
: Instruction TLB, data TLB and main TLB misses.

The event counts extracted from simulation runs were fitted to a simple linear model, i.e., run-time is assumed to depend linearly on event counts:

$$t_j = \sum_i \beta_i x_{i,j} \quad (1)$$

where t_j is the run-time of workload j, β_i is the linear coefficient of event i and $x_{i,j}$ is the event count of event i for workload j. The coefficients were calculated with the linear least squares method and the accuracy of the estimation was evaluated with the leave-out-one cross-validation method. Simulation overhead was measured on a desktop PC with an AMD Phenom processor running at 2.2GHz. The micro-benchmarks were used for calibrating our model parameters but for validating the accuracy and overhead of our method only the application set was used.

RESULTS

Figure 2 shows the average estimation error for the application workloads for different hardware platforms when the simulation detail is varied. The first bars (all) show the estimation error with full instrumentation. The second bars (no class) show the estimation error when the instructions are not classified based on their type. The third bars (no TLB) show the estimation error when TLB statistics are discarded in addition to the instruction classification and the fourth bars (no L2) when also L2 cache statistics are discarded.
As shown in Figure 2, the effects of different statistics on the accuracy of estimation vary on different target platforms. For example instruction classification and L2 cache simulation seem quite important for the Cortex-A8 based platforms. The last two bars for the

Table 2: Simulation overhead compared to native execution

Platform	Instrumentation				
	none	classify	L1	L2	TLB
PB11MPCore	0.3	1.3	2.1	2.1	8.5
Naviengine NE1	0.5	2.1	3.4	3.4	14
Beagleboard C3	1.0	4.3	6.9	7.0	28
Beagleboard C4	1.4	5.8	9.4	9.5	39
KZM	0.9	3.8	6.1	6.1	25
Tegra	2.4	9.7	15	16	63

NaviEngine NE1 have the same height because there is no L2 cache in the hardware.

Table 2 shows the average relative simulation time of the application workloads on our simulation host compared to native execution on real hardware. The first instrumentation column (none) shows the relative runtime with none of the instrumentations enabled. The second column (classify) shows the overhead of instruction counting and classification. The following columns (L1, L2, TLB) show the overhead when in addition L1 cache, L2 cache and TLB simulations are enabled.

Several observations can be made from Table 2. First of all, the difference between no instrumentation and even the simplest statistics collection is significant. This is largely due to the unoptimized implementation of our instrumentation. Even with empty instrumentation, the cost of calling a function for each translated instruction is quite high. The overhead could be reduced significantly by modifying the instrumentation to operate at the translation block level or by emitting TCG code directly instead of using function calls.

Another observation is that the cost of adding L2 simulation is almost immeasurable. This illustrates the fact that estimation accuracy can sometimes be improved dramatically with virtually no overhead.

The last column of the table illustrates the fact that high simulation detail can cause prohibitive overhead. The TLB simulation tries to model the hardware accurately by using fully associative micro-TLB models separately for the instruction and data sides and a two way set associative main TLB. Especially, simulating high associativity structures is very costly.

DISCUSSION

Figure 2 and Table 2 illustrate the flexibility of our approach. When high execution speed is important, simulation detail can be reduced by trading off estimation accuracy. When better accuracy is required, simulation details can be gradually increased by trading off execution speed.

In our current modelling we have tried to follow the hardware details with reasonably high accuracy. Our L1 cache models are four way set associative and L2 cache is simulated with eight way set associativity and cache sizes are set according to hardware documentation. The TLB is modelled according to the available documentation as well. However, using parametrized models enables abstracting the architectural details. Instead of relying on the level of detail of the hardware models, we can use calibration to get the timing estimates to acceptable levels of accuracy.

As an experiment we modelled the Tegra platform with direct mapped L1 and L2 caches and one level unified TLB. To compensate for the anticipated increase in miss rates we doubled the sizes of the caches and TLB compared to hardware documentation. With this model the average estimation error for the application workloads was 10%, very close to the average 9.9% error with the more detailed modelling. However, the effect on simulation overhead was quite dramatic. Compared to using no instrumentations, using the detailed model resulted in a slowdown factor of thirty, whereas using the abstracted model reduced the slowdown factor to seven.

Our model calibration is based on linear least squares estimation which assumes that the observed variables are independent. However, there can be strong correlations between the events collected with our instrumentation. This can result in unrealistic values for some of the model parameters. Even though the average estimation error can remain reasonable even with unrealistic parameter values, such parameters can cause large estimation errors for atypical workloads. This problem is of particular concern if the test set used for calibration is not sufficiently extensive compared to the number of model parameters. Therefore, relying on just the average estimation error can be misleading, which is why the model parameter values and the maximum estimation error should also be examined.

In our current experiments the maximum estimation error varied between 16% and 34% for full instrumentation, depending on the hardware platform. Model parameters were mostly surprisingly realistic with memory write related parameters being a notable exception. This is most probably due to our current simulation models being unable to capture the temporal locality related behavior of software execution. Modelling spatial locality with parametrized models seems quite feasible whereas modelling the hardware structures related to temporal locality, like write buffers, interconnects and memory controllers is not as straight forward.

The use of a linear model for our current instrumentation is reasonable given that the cost of executing a particular type of instruction or a miss in a cache should be pretty constant. However, various hardware structures (e.g., pipelining and buffers) do cause non-linear effects. Therefore using nonlinear models would be an interesting future experiment. Such models could also enable raising the abstraction level of the modelling even higher.

CONCLUSION

In this paper, we presented an evaluation of a simulation mechanism for understanding non-functional properties of computer software. In our experimentation, we concentrated on simulating the timing of the execution of software as many other non-functional properties can be estimated if we know the timing.

The traditional way of simulating the timing behavior is based on accurate instruction set simulators. Such simulators are orders of magnitude slower than real hardware, and thus, unsuitable for software developers. Our methodology is based on modern binary translation schemes that enable attaching simulation models to the functional simulators used in software development work. Our experimentation is based on a production quality development tool, QEMU.

For a software developer, approximate information of non-functional properties is enough to make a correct design decision. Our experiments show that reasonable accuracy can be achieved using abstract machine models in simulations that have feasible simulation speed. Further, we show that there is a clear trade-off between simulation accuracy and simulation speed. Thus, such simulations are not fixed to any specific level. Rather, we can gain more accuracy by using more simulation time and vice versa.

Our current methodology supports simulation of multicore processors, but does not support the simulations themselves to be run on multiple processors or cores. As the number of processor cores is rising in cell phones, laptops, etc., we see this direction of research as the most important one for the future.

REFERENCES

ARM, 2010. *ARM Processors.* `http://www.arm.com/products/processors/index.php`.

Aycock J., 2003. *A brief history of just-in-time. ACM Computing Surveys (CSUR)*, 35, no. 2, 97–113. ISSN 0360-0300. doi:http://doi.acm.org/10.1145/857076.857077.

Bellard F., 2005. *QEMU, a fast and portable dynamic translator.* In *ATEC '05: Proceedings of the annual conference on USENIX Annual Technical Conference.* USENIX Association, Berkeley, CA, USA, 41–41.

Burger D. and Austin T.M., 1997. *The SimpleScalar tool set, version 2.0. ACM SIGARCH Computer Architecture News*, 25, no. 3, 13–25. ISSN 0163-5964. doi:http://doi.acm.org/10.1145/268806.268810.

Edler J. and Hill M., 1998. *Dinero IV Trace-Driven Uniprocessor Cache Simulator.* `http://pages.cs.wisc.edu/~markhill/DineroIV/`.

Franke B., 2008. *Fast cycle-approximate instruction set simulation.* In *Proceedings of the International Workshop on Software & Compilers for Embedded Systems (SCOPES).* 69–78.

Gecsei J.; Slutz D.R.; and Traiger I.L., 1970. *Evaluation techniques for storage hierarchies. IBM Systems Journal*, 9, no. 2, 78–117. ISSN 0018-8670. doi:http://dx.doi.org/10.1147/sj.92.0078.

Hirvisalo V.; Kiminki S.; Knuuttila J.; and Töyry T., 2010. *Technical Report ESG-pQEMU-1.* ESG/CSE Aalto University, to appear 2010.

Miettinen A. and Hirvisalo V., 2009. *Energy-efficient parallel software for mobile hand-held devices.* In *First USENIX Workshop on Hot Topics in Parallelism (HotPar'09).*

QEMU, 2010. *QEMU – open source processor emulator.* `http://wiki.qemu.org/`.

Sites R.L.; Chernoff A.; Kirk M.B.; Marks M.P.; and Robinson S.G., 1993. *Binary translation. Communications of the ACM*, 36, no. 2, 69–81. ISSN 0001-0782. doi:http://doi.acm.org/10.1145/151220.151227.

Topham N. and Jones D., 2007. *High speed CPU simulation using JIT binary translation.* In *Proceedings of the 3rd Annual Workshop on Modeling, Benchmarking and Simulation (MoBS).*

Weaver V. and McKee S., 2008. *Are cycle accurate simulations a waste of time?* In *7th Workshop on Duplicating, Deconstructing, and Debunking.*

Weber S.J.; Moskewicz M.W.; Gries M.; Sauer C.; and Keutzer K., 2004. *Fast cycle-accurate simulation and instruction set generation for constraint-based descriptions of programmable architectures.* In *CODES+ISSS '04: Proceedings of the 2nd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis.* ACM, New York, NY, USA. ISBN 1-58113- 937-3, 18–23. doi:http://doi.acm.org/10.1145/1016720.1016728.

MODEL BASED CONTROL SOFTWARE SYNTHESIS FOR PAPER HANDLING IN PRINTERS

Chitralekha Pillai
Embedded Systems Institute
P.O. Box 513, 5600 MB Eindhoven
The Netherlands
chitralekha.pillai@esi.nl

Ronald Fabel
Océ-Technologies B.V
P.O. Box 101, 5900 MA Venlo
The Netherlands
ronald.fabel@oce.com

Lou Somers
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven
The Netherlands
l.j.a.m.somers@tue.nl

KEYWORDS
Model based engineering, software synthesis, multidisciplinary design, domain specific languages, behavior specification.

ABSTRACT

Control software is an integral part of new complex electromechanical systems, such as professional high speed printers. The development of these multidisciplinary products involves a number of iterative and incremental cycles of prototype creation. Automatically generating control software is a big leap in rapid prototyping of these products. A model based approach is an effective way to manage the complexity as well as to enable new insights in terms of innovations and technology risks. Models are widely used for understanding and designing the system, but not yet for control software generation. The main challenge in generating complete and executable control software is the difficulty in specifying the behavior at the abstraction level of a generic modeling language. By specifying the static and behavior design information in domain specific models, the level of abstraction is raised to using the concepts and rules of a specific problem domain. Focusing to a narrow domain enables automation to the level of full code generation. This industrial case explains how a domain specific modeling approach was applied to paper handling in printers to generate the control software. The promising result is a motivation to develop this approach further for other printer modules.

INTRODUCTION

Productive printing systems have evolved into innovative and complex multi-disciplinary products. The amount of software control in printers is increasing because of the extensive functionality they offer. As a result, a lot of development effort is shifted towards development of control software. Models provide an effective way to develop large, reliable, real-time software systems because they help engineers to manage complexity (Heemels et al. 2006). A model-based synthesis of control software is generating the control software directly from the models. The generated code can be linked with the underlying platform code and compiled to a finished executable without additional manual effort. The key to this level of code generation is to specify the multidisciplinary design information at the right abstraction level. A domain specific approach is ideal because it fits to the domain needs and relates to the required behavior closely. A domain specific code generator can then translate the high level specification to code.

Automating the control software development opens up opportunities for quicker feedback on design decisions. It works well when software and hardware requirements and specifications are changing during development of a product. The ability to synthesize software from models helps to ensure the consistency of design information throughout the development lifecycle. This ensures that the design specification matches the implementation and thus tremendously decreases the development effort and time as well as improves the quality of the products. The role of embedded software developer now changes to that of a domain expert who can focus on multidisciplinary aspects of the design. A common overview of the essential multidisciplinary design information represented in a domain model serves as an excellent and necessary means to manage design information across different activities and different disciplines during the development cycle.

In Figure 1 we visualize how control software synthesis facilitates the development of complex systems like printers using a model-based approach. The potential benefits of automation in the development process can be observed.

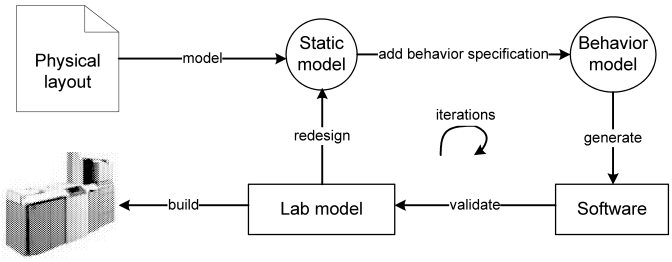


Figure 1: Model Based Control Software Synthesis: Workflow

The development starts with envisioning the printer and coming up with a model of the physical layout. The resulting multidisciplinary design information model represents the machine description at an abstract level which can be communicated across disciplines. If this is a model understood by all disciplines, the consistency of the design information can be maintained throughout the development (Schindler 2008). Based on this static model and a behavior specification language, the functional requirements can be expressed as a behavior model. This behavior model along

with the underlying multidisciplinary design information model can be used to automatically generate the control software. This generated software has I/O control interface calls to control the electromechanical parts of the printer. The generated software can be validated in the lab model. For a redesign, another iteration is started. When the lab model evolves to a sufficient quality, it is used to build the actual machine.

It can be observed that automatic generation enables consistency of the model and the generated software. The development time can be shortened by quick design decision verification and reduced development effort by automating repetitive tasks.

Being free from the manual creation and maintenance of source code significantly improves developer productivity. The reliability of automatic generation compared to manual coding will also reduce the number of defects in the resulting programs and thus improves quality. This automated transformation process is often referred to as "correct-by-construction," as opposed to conventional handcrafted "construct-by-correction" software development processes that are tedious and error prone (Schmidt 2006).

The remainder of the paper explains the details of the case study. First, the domain overview is explained, which introduces the paper path in a printer. This helps in understanding the essential multidisciplinary design information of a paper path. These concepts are mapped to a static design information model of the paper path. Then the domain concepts used to define the sheet behavior specification language and the resulting specification are described. Finally, the software synthesis setup is explained.

PAPER HANDLING IN PRINTERS

A paper handling module consists of both hardware and software responsible for sheet movement inside the printer, from a Paper Input Module to a Finisher. These different elements in the printer (such as mechanical parts, electrical connections and software) work together to move the paper.

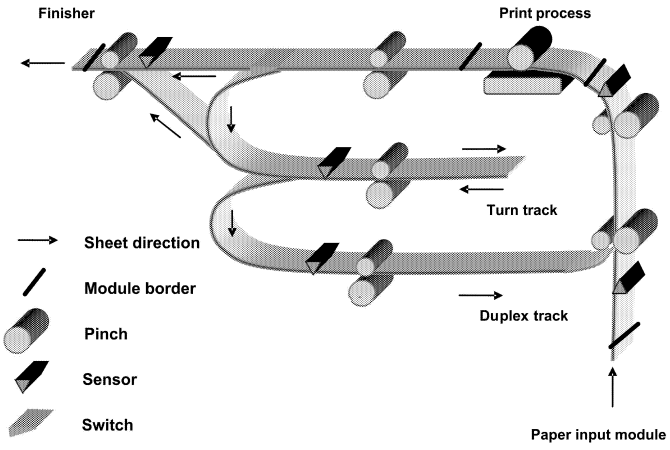


Figure 2: Conceptual Overview of a Paper Path

The track is in the form of iron plates that lead the sheet through a predefined shape of movement. The transportation of sheets on the track is realized by pinches. Each pinch consists of two pinch rollers that hold and move the sheet by means of friction. Pinches are directly or indirectly (through clutches and/or pulleys in between) driven by motors. Note that one motor can drive more pinches simultaneously. The distance between pinches should not be larger than the length of the smallest sheet which the printer is specified to handle, otherwise the paper sheet loses its speed. A sheet may, however, be transported by multiple pinches simultaneously. Paper path sensors detect the sheet. They can be used to track the real position of sheets in the paper path by sheet edge detection. The switch is intended for changing the direction of sheets towards the finisher or the turn track. As it can be seen in the layout example in Figure 2, there may be a turn track for turning the sheet (in case of duplex printing) and multiple tracks to the Finisher, an upper track and a lower track.

The control software is responsible for controlling these various mechanical and electrical parts and delivering sheets of the right format (wrong formats result in an error), with a specified alignment, position, and skewness, with a required speed and temperature on a scheduled time to the Print Process (where the actual printing takes place), and transportation of sheets from the Print Process to the Finisher.

MULTIDISCIPLINARY DESIGN ASPECTS

Though on first sight a paper path seems simple, the realization of sheet transportation involves many subtle trade-offs such as cost, speed, productivity and space. The following (strongly interconnected) aspects play a role in paper handling module design:

Spatial layout: the shape of the paper path. The lengths of the tracks in the path and positions of mechanical parts such as pinches and sensors become the characterizing parameters.

Timing: sheet position in the paper path at a given time. This has to be synchronized with other related functions in the printer, like the printing process.

Drive map: connection of pinches to motors, either directly or by means of clutches and pulleys. Such a connection configuration dictates the constraints on using pinches to implement timing design. For example: if two pinches are connected to the same motor, one cannot turn in another direction than the other.

Corrections: compensation for anticipated issues. Nominal (planned) corrections are part of the alignment process. Corrections compensate for small tolerances in sheet movement caused by different paper types, slipping of pinches, tolerances in mechanical production, or simple wear and tear of mechanical parts.

Exception Handling: takes care of errors that can occur during sheet movement. For example: a sheet may have a crease at the side, which causes it to get stuck in a paper track. Such an error must be handled accordingly (e.g. by moving the sheet forcefully to a certain location where an operator can reach it and remove it).

STATIC DESIGN INFORMATION MODEL

The highest level description element in the model is chosen to be the "Machine", which consists of several aspects. The multidisciplinary design aspects explained above are mapped to these aspects. In this case study, the static design information model consists of only relevant aspects related to the paper path:

Parts: physical things in the machine. Parts describe properties of machine components such as pinches, sensors, motors, that are essential as static design information.

Topology: describes the essence of the spatial layout, which serves as a spatial description including the relevant parts identified as point of interests (POI) in the layout. The descriptive elements of the topology are segments. A sequence of segments specifies a route. Points of interest (POI) can be any points which are used for specifying any important issue onto the topology.

Appearance: describes the position of all interesting physical elements in the machine. It is expressed as co-ordinates in the machine layout.

Drive map: contains the connection type and gear from motor control interface to parts.

I/O aspects: contains input and output lines of parts, required for I/O layer calls.

SHEET BEHAVIOR SPECIFICATION LANGUAGE

The sheet movement in the paper path is termed as "sheet behavior". The control software in the paper handling module is responsible for controlling the various mechatronic devices to achieve the correct sheet behavior. This is crucial since the paper has to be strictly timed with respect to other functionalities of the printer, without any collisions. The timing designer is responsible for describing the timing requirements for different kinds of print jobs to realize the corresponding sheet behavior on the machine. Each sheet undergoes velocity changes according to the timing requirements. Often these velocity changes are derived and calculated from the spatial layout. The timing designer can do simple calculations in an excel sheet to come up with references to when and where the sheet velocity has to be changed. Hence, the sheet behavior can be represented in terms of sheet velocity (Beckers et al. 2007). The corresponding control software is a sequence of control actions for the mechatronic devices to achieve this sheet velocity profile.

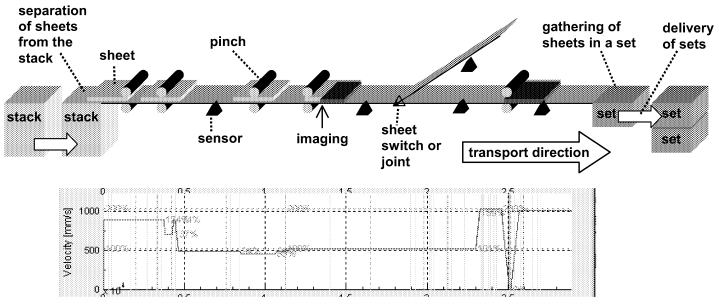


Figure 3: A Typical Sheet Velocity Profile in a Paper Path

The domain specific language consists of a set of directives which helps to specify the sheet behavior in the paper path. These directives were derived by classifying the control to four situations as explained below.

The arrival of a sheet

The information regarding when to expect a sheet is available from the print job definition. This is the starting directive of sheet behavior specification:

SheetArrival (arrivalPoint, arrivalTime, speed)

Here, *ArrivalPoint* indicates the reference position associated with the synchronization time specified by the driving software. This is included as POI in topology. *ArrivalTime* is the synchronization time to expect the arrival of a sheet at the reference position from the paper stack. As mentioned above, this is obtained from the print job. *Speed* is the speed (in mm/s) at which the sheet reaches the reference position.

Sheet velocity changes

The timing designer wants to specify the position of a sheet at any point in time. This is done by calculating the required sheet velocity profile. The drive map of a pinch contains information about the maximum allowed acceleration for that pinch given the mechatronic design. As a future work, the behavior specification can be extended to include a constraint check to make sure the required acceleration is within the maximum acceleration given in drive map.

SheetVelocityChange (Reference time/position, Offset time/distance, speed, acceleration)

Reference Time/Position is the reference based on which the sheetvelocitychange point is based. *Offset Time/Position* is the positive offset to add to the reference to get the exact point. *Speed* is the desired speed value at that point. *Acceleration* is the required acceleration for the speed change given the mechatronic design.

Sensor controls

A sensor will give the software momentarily information of a position/time pair of the leading or trailing edge of a sheet. The absolute position of the sensor is found in topology, the absolute time is available in the software after the sensor senses the sheet. The behavior specification can introduce a reference variable for this captured time which can be used in other directives to specify timing of actions.

DetectSheet (sensor, detectionwindow, time reference)

Sensor indicates the sensor which has to detect the sheet. *Detectionwindow* is the window in which the sensor expects the sheet. If the sheet is not detected within that window, an error should be flagged. *Time reference* indicates the time captured during this action. This can be used as a time reference for usage in other directives.

Sometimes a timing designer wants to specify a speed change point based on the sheet position in the paper path. In order to be able to execute a directive which is based on a sheet position, it is needed to "know" the sheet position. If

the sheet position is not made available by the driving software, any position reference which is not on a sensor position must be calculated by using the time and the velocity profile. In such a case, the calculation can already be done in an excel sheet and the directive can use a time reference. So a position reference can only be used if the sheet position is available. This is done by a position capture at a sensor. The position capture sensors can capture the position of sheet in terms of motor steps. The directive to do so also introduces the position reference identifier that refers to the specific position following service in the driving software.

TrackSheetPosition (sensor, pinch, position reference)

Sensor indicates the sensor point from which the position tracking starts. *Pinch* is the pinch (and thus motor) to be used for this purpose. *Position reference* is the identifier to be used in further directives as the position reference.
Note that this directive is not purely a sheet behavior directive. It enables the timing designer to specify a position reference according to his requirements, which can be used in other directives.

Controlling pinches and switches when there is no paper

When the sheet is approaching a pinch, the pinch has to be prepared to receive the sheet so that there is a smooth takeover of sheet. If the receiving pinch is not running at sheet velocity, the sheet movement is not smooth. We need to control the pinch/motor in this situation.

PinchPrepare (Pinch, Reference time/position, Offset time/distance, speed, acceleration)

Pinch denotes which pinch has to be prepared to receive the sheet. *Reference Time/Position* is the reference based on which the pinchprepare point is based. *Offset Time/Position* is the offset to add to the reference to get the exact pinchprepare point. *Speed* is the desired speed of the pinch. Acceleration is the maximum allowed acceleration for that pinch given the mechatronic design.

The directives explained above can be used to specify the sheet behavior in a modular manner. The sheet behavior specification contains a sequence of these directives with the appropriate parameters. There are references to the multidisciplinary static design information model. Each directive corresponds to a number of control actions to be performed at specific moments. Hence, the generated control software can be visualized as a state machine, with control actions being executed when corresponding time or position guards become true.

SOFTWARE SYNTHESIS

The aim of model based control software synthesis is generating complete finished code for what has been modeled. The software synthesis approach applied in this case is not based on formal methods (Bertens et al. 2008, Markovski et al. 2010), but on domain specific concepts. Every domain contains its own specific concepts and correctness constraints. By working at this abstraction level, the behavior of a system can be precisely specified, which is difficult in terms of a formal method or a generic modeling language (Fowler and Scott 1999). This level of specification is crucial for full code generation. The rules of the domain can be included as constraints, ideally making it impossible to specify illegal or unwanted design models (Kolovos et al. 2006). This results in an expressive yet bounded model within the domain that can be used for code generation (Luoma et al. 2004).

The software synthesis system consists of domain specific models, a domain specific code generator and a domain framework (Kelly and Tolvanen 2008). The generator extracts the information from the models and transforms it into code. This translation is done based on the rules of the domain and how the generated output should look like. The domain framework provides the interface between the generated code and the underlying platform. The generated code is linked with the framework and compiled to a finished executable without any additional manual effort. The generated code is thus simply an intermediate by-product on the way to the finished product, like object files in C compilation.

Let us see how this approach was applied in the printer case. In the printer embedded software architecture, there is a layer called controller which is responsible for mapping a print job to page commands. Each sheet gets a new page command. The page command contains information such as simplex/duplex printing, paper size and when to expect the arrival of the sheet. The sheet behavior for each sheet is based on this page command and the specification in the multidisciplinary design information model.

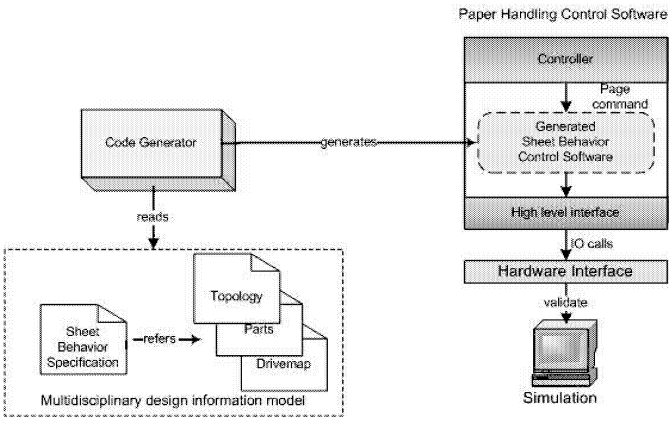


Figure 4: A Simplifed Representation of the Model Based Control Software Synthesis Setup

The code generator reads the multidisciplinary design information model which consists of both static and sheet behavior specification. Each directive in the sheet behavior specification means certain desired control behavior. The translation of a directive by the code generator involves determining which mechanical parts are responsible for the desired result and setting control commands to those parts at specific instants. The details of the mechanical parts are available in the static design information model. The specific instances can be deduced from the arguments in the directive.

For example, consider the directive

SheetVelocityChange(TurnPi,10,890,20000).

This means that the sheet velocity has to be changed to 890mm/s at an offset distance of 10mm from pinch with name TurnPi. Now the code generator has the information about which motor is driving the TurnPi. Based on the paper size, the code generator can also find out which previous pinches are still influencing the current sheet. The control commands have to be transmitted to all concerned motors for the desired speed change.

Thus the output of the code generator is a state machine with a sequence of control commands which gets executed when the specified moment has reached. In order to integrate this state machine to the underlying platform, a high level interface layer has been defined. This layer takes care of abstracting the hardware I/O calls to generated code level. Another advantage of this abstraction is the reuse of the code generator even if the underlying hardware changes.

There are no manual changes to be performed in the generated software. It is linked with the controller software layer and compiled to a finished executable which can be tested on a simulation environment before the actual machine is ready.

Evaluation / results

During this case study, the evaluation was carried out in a simulation environment (Software-In-the-Loop, SIL) of a real printer development project. The multidisciplinary design information model consisted of the static design information of the printer (layout, drive map, etc.) and the sheet behavior specification for a "simplex" print job. The specification started with a SheetArrival, followed by several SheetVelocityChanges and DetectSensor directives. The code generator now translated the specification to a state machine which comprised of all the necessary control commands for seven motors and four sensors which are involved in the simplex behavior. There were 26 state transitions automatically generated to achieve this behavior.

The generated executable code was tested in the SIL environment, which delivers a log showing the timing behavior of each sheet of paper. This log was verified with a timing model to see if the intended paper timing behavior matches with that of the generated code. The results showed that the timing deviation was less than 1ms. The velocity profile from the timing model matched with that from the generated code. The timing was also cross checked with the existing handcrafted code. The results proved the feasibility of full code generation from models, which matches with the performance of a handcrafted implementation that took a lot of time to build.

CONCLUSION

Automation of control software development has been proved feasible in the case of the paper handling function in printers. The concept was demonstrated by synthesizing control software for simplex sheet behavior in a real printer development project.

This research is a stepping stone to an improvement opportunity in the conventional design and development of control software. The multidisciplinary design information model guards consistency and evolution of the system design, and facilitates communication across different disciplines. In model based software synthesis, the focus shifts from solution domain to problem domain. Embedded software developers who have been working at a low level of abstraction, writing code, are motivated to work at a higher level of abstraction and develop the models in the problem domain from which code, tests, documentation, and so on, are generated automatically. Their role will change from that of crafting software and tediously repeating work for every iteration to a domain expert who actually adds more detail to a design, leaving the laborious activity of converting design decisions to working software to tooling, wherever this is sensible and saves effort and time. This leads to improved productivity, better product quality, hiding complexity, and leveraging expertise.

REFERENCES

Beckers, JMJ, W.P.M.H. Heemels, B.H.M. Bukkems, and G.J. Muller. 2007. "Effective industrial modeling for high-tech systems: The example of happyflow". In *Proceedings of the 17th Annual International Symposium of the International Council On Systems Engineering, INCOSE* (San Diego, USA, June 24-28).

Bertens, E., R. Fabel, M. Petreczky, D.A. van.Beek, and J.E. Rooda. 2009. "Supervisory control synthesis for exception handling in printers". In *Proceedings of the 10th Philips Conference on Applications of Control Technology* (Hilvarenbeek, Netherlands, Feb 3-4).

Fowler, M. and K. Scott. 1999. *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. Addison-Wesley.

Heemels, W.P.M.H., E. van de Waal E., and G.J. Muller. 2006. "A multi-disciplinary and model-based design methodology for high-tech systems" In *Proceedings Conference on System Engineering Research, CSER* (Los Angeles, USA, April).

Kelly, S. and J.-P. Tolvanen. 2008. *Domain-Specific Modeling: Enabling full code generation.* John Wiley and Sons, New York.

Kolovos, D.S., R.F. Paige, T. Kelly, F.A.C. Polack. 2006. "Requirements for Domain-Specific Languages". In *Proceedings of the 1st ECOOP Workshop on Domain-Specific Program Development, DSPD 2006* (Nantes, France, July).

Luoma, J., S. Kelly, J.-P. Tolvanen. 2004. "Defining Domain-Specific Modeling Languages: Collected Experiences". In *Proceedings of the 4th OOPSLA Workshop on Domain-Specific Modeling, DSM'04* (Vancouver, Canada, Oct).

Markovski, J., K.G.M. Jacobs, D.A. van Beek, L.J.A.M. Somers, and J.E. Rooda. 2010. "Coordination of Resources using Generalized State-Based Requirements". To appear in *Proceedings of the 10th International Workshop on Discrete Event Systems.* (Berlin, Aug 30-Sep 1).

Schindler, E. 2008. "MoBasE: A Multidisciplinary Model Based Engineering Framework for Development of Production Printers". Technical Report, Stan Ackermans Institute, Eindhoven University of Technology.

Schmidt, D.C. 2006. "Model-Driven Engineering". *IEEE Computer* 39, No. 2 (Feb), 25-31.

ECONOMICS SIMULATION

OUTPUT DYNAMICS AND (s, S) STRATEGIES IN AN AGENT BASED MACROECONOMIC MODEL

Oscar Alonso
Hiroshi Deguchi
Yuhsuke Koyama

Tokyo Institute of Technology, Japan
email: alonso08@cs.dis.titech.ac.jp, deguchi|koyama@dis.titech.ac.jp

KEYWORDS
Agent-Based Macroeconomics, (s, S) Strategies, Monetary Neutrality

ABSTRACT

A well known result from (Caplin and Spulber 1987) shows that even in the presence of price rigidity, aggregated product of an economy may be unaffected by changes in the money supply. Here we relax the assumptions of continuity of such result by using agent based simulation, and analyzing the dynamics of output under such conditions.

We show that in the agent-based version of the model money is no longer neutral and cyclic dynamics emerge. Such dynamics is influenced mainly by expectations of income, expectations of inflation, and implementations of (s, S) strategies. Oscillations in output grow stronger as the economy departs from initial equilibrium conditions.

INTRODUCTION

Price dynamics emerges from the pricing decisions of individual firms and households. A central issue in macroeconomics modeling is to recreate the dynamics of inflation (general price level increase) as the result of the pricing decisions of individual agents. Most macroeconomic models consider time dependent price updating, such as Calvo price setting model and the Taylor staggered contracts model (Walsh 2003), in which the timing of the price changes is exogenously determined. Conversely, a few models feature state based price updating, such as (s, S) strategies (Sheshinski and Weiss 1977), in which the decision of updating prices is endogenous, given the state of each agent.

State based pricing has been difficult to incorporate into macroeconomic models, and has been unable to reproduce the dynamic behavior of inflation, such as its persistence after changes in monetary policy and its relation with aggregated output. Particularly, a result from (Caplin and Spulber 1987) shows that even in the presence of price rigidity (where under some circumstances firms do not change prices in reaction to changes in the market), output may be stable and money neutral. Such result is based on the assumption of a continuum of agents, (s, S) strategies for pricing, and an initial uniform distribution of the logarithm of prices.

This paper aims to contribute to the understanding of the influence of (s, S) strategies in the behavior of an economy. We relax the assumption of continuity from (Caplin and Spulber 1987) by using an Agent-Based model. This approach allow us to model the state of each agent independently, implement state based pricing strategies, and generate aggregated behavior. We use an economic model based on (Blanchard and Kiyotaki 1987) which features monopolistic competition. We extend such model by incorporating (s, S) strategies for price updating.

The paper is structured as follows: First, we introduce some concepts such as (s, S) strategies and expectations modeling. Additionally, we make a short exposition of the Blanchard and Kiyotaki model of monopolistic competition. Then, we describe our extension of such model. Next, experimentation and results are presented. In the final section we present our conclusions.

THEORETICAL BACKGROUND

(s, S) Strategies

Price rigidity occurs when firms, faced with changes in the market, decide not to change the price of the products immediately. So, prices remain fixed for several periods of time, instead of changing continuously. It is considered a cause of monetary non-neutrality, as prices do not react to changes in money supply and then monetary policy influences economic performance. The existence of price rigidity is explained by cost associated to price updating, referred to as *Menu costs*.

(Sheshinski and Weiss 1977) studied optimal pricing under inflation, and determined that under menu costs, optimal pricing strategies are of the (s, S) kind: the real price of a good should fluctuate between two values s and S, which satisfy some optimality conditions.

As nominal price P_i is constant and the price index P grows continually, the real price $z_i = P_i/P$ decreases continually. Under s, S strategies, the optimal behavior of the firm is to wait for the real price to reach the value s, and then update the real price to the upper bound S.

Expectations Modeling

In several situations agents have to anticipate the state of the economy in order to make decisions, and behavior of the economy depends on how such expectations are formed. In analytical models expectations are modeled as rational: they are correct in average ($x_t = E[x_t] + \epsilon$, where ϵ is an error with mean 0), and agents process information optimally. This approach is very useful analytically, but does not address the question of how such expectations are formed.

Recently, the study of the behavior of explicit forecasting rules has gained attention. Research has focused on four aspects: stability of rational expectations equilibrium under learning, equilibrium selection, transition dynamics, and dynamics out of equilibrium (Arifovic and Bullard 2001). In this paper, we study the stability of (Caplin and Spulber 1987) results in a discrete setup. We build an agent-based model as close as possible to the continuous version. We replace the assumption of Rational Expectations, for an implementation of Adaptive Learning.

Adaptive Learning assumes that agents behave like econometricians, and they know the structure of the model, but must learn the parameters from the data. Then, the regression model used by the agents has the following form:

$$x_t = \Theta_t^{'} x_{t-1} + \epsilon,$$

and the parameters matrix Θ_t evolves according to

$$\Theta_t = \Theta_{t-1} + \phi R_{t-1}^{-1} x_{t-2}(x_{t-1} - \Theta_t^{'} x_{t-2})^{'}$$

$$R_t = R_{t-1} + \phi(x_{t-1} x_{t-1}^{'} - R_{t-1}).$$

ϕ is called the gain parameter, and expresses the relative weight of new observations. When $\phi = 1/t$, the algorithm is equivalent to recursive least squares. Also, when the regressor is just a constant, Adaptive Learning is equivalent to the Adaptive Expectations model, which has the form

$$E[x_t] = E[x_{t-1}] + \phi(x_{t-1} - E[x_{t-1}]).$$

BLANCHARD AND KIYOTAKI MODEL

The economy modeled in this paper is a dynamic version of (Blanchard and Kiyotaki 1987). It has n households, indexed by h and m firms indexed by i. Firm i produces good i and house h provides labor kind h. It features monopolistic competition in the goods market and labor market, and output is demand determined.

Household Behavior

At each period of time t, household h sets wage $W_{h,t}$ and consumption budget $B_{h,t}$ maximizing the myopic utility function

$$U_{h,t} = (C_{h,t})^{\gamma}(M_{h,t}/P_t)^{1-\gamma} - H_{h,t}^{\beta},$$

$$C_{h,t} = \left[\sum_{i=1}^{m} \left(\frac{1}{m}\right)^{\frac{1}{\theta}} {C_{i,h,t}}^{\frac{\theta-1}{\theta}}\right]^{\frac{\theta}{\theta-1}},$$

with budget constraint

$$M_{h,t} + B_{h,t} = I_{h,t},$$

$$I_{h,t} = M_{h,t-1} + T_t + \sum_{i=1}^{m} W_{h,t} H_{i,h,t} + V_t.$$

Basically, households receive utility from consumption $C_{h,t}$ and real money holdings $M_{h,t}/P_t$, and suffer disutility from providing labor $H_{h,t}$. Also, they receive money transfers T_t from the government, and a share of profits V_t from firms. T_t and V_t are the same for all agents. Income $I_{h,t}$ is distributed between consumption and money holdings.

Regarding notation: γ is a weighting parameter, $M_{h,t}$ is money holdings, P_t is the price index which satisfies $P_t C_{h,t} = \sum_{i=1}^{m} P_{i,t} C_{i,h,t}$, and $\beta - 1$ is the elasticity of the marginal disutility of labor. $C_{h,t}$ is a CES aggregate of the consumption of individual goods $C_{i,h,t}$, where θ is the elasticity of substitution between goods. $H_{i,h,t}$ is the amount of labor kind h required by firm i.

Firms Behavior

Firms' production function has returns to scale $\frac{1}{\alpha} < 1$, and is given by

$$Q_{i,t} = L_{i,t}^{\frac{1}{\alpha}},$$

$$L_{i,t} = \left[\sum_{h=1}^{n} \left(\frac{1}{n}\right)^{\frac{1}{\sigma}} H_{i,h,t}^{\frac{\sigma-1}{\sigma}}\right]^{\frac{\sigma}{\sigma-1}},$$

where $Q_{i,t}$ is quantity of goods and $L_{i,t}$ is a CES aggregate of labor, with elasticity of substitution σ. Firms maximize current period's profit $V_{i,t}$ given by

$$V_{i,t} = P_{i,t} Q_{i,t} - W_t L_{i,t}.$$

W_t is a wage index satisfying $W_t L_{i,t} = \sum_{h=1}^{n} W_{h,t} H_{i,h,t}$.

Optimal Pricing

Optimization leads to the following expressions for optimal prices and wages:

$$P^*_{i,t} = P_t \left[\alpha \frac{\theta}{\theta - 1} \frac{W_t}{P_t} \left(\frac{Y_t}{m} \right)^{\alpha - 1} \right]^{\frac{1}{1+\theta(\alpha-1)}}$$

$$W^*_{h,t} = W_t \left[\frac{\beta}{\mu} \frac{\sigma}{\sigma - 1} \frac{P_t}{W_t} \left(\frac{L_t}{n} \right)^{\beta - 1} \right]^{\frac{1}{1+\sigma(\beta-1)}}$$

where Y_t is total product, L_t is total labor demand, and $\mu = \gamma^\gamma (1-\gamma)^{1-\gamma}$. m and n are assumed to be large enough to consider the effects of an individual price or wage on P_t and W_t negligible.

ECONOMIC MODEL

In our dynamic model, money M grows at a constant rate $\bar{\pi}$. Money growth is accomplished by money transfers to households.

Pricing Strategy

s and S refer to the real prices between which the real price fluctuates. We use s_F and S_F to refer to markups over the optimal price $\frac{P^*_i}{P}$. Then, the real price $\frac{P_i}{P}$ will fluctuate between the values $s_F \frac{P^*_i}{P}$ and $S_F \frac{P^*_i}{P}$.

The strategy can be stated in the following two ways:

$$P_{i,t} = \begin{cases} P_{i,t-1} \frac{S_F}{s_F}, & \text{if } P_{i,t-1} < s_F E[P^*_{i,t}] \\ P_{i,t-1}, & \text{otherwise.} \end{cases}$$

$$P_{i,t} = \begin{cases} E[P^*_{i,t}] S_F, & \text{if } P_{i,t-1} < s_F E[P^*_{i,t}] \\ P_{i,t-1}, & \text{otherwise.} \end{cases}$$

However, the first definition preserves the uniform distribution of the logarithm of prices, while the second does not. Wage strategy is defined in the same way, with markups s_W and S_W.

Initial Conditions

The model is initialized in equilibrium conditions. The initial distribution of the logarithms of prices is uniform in $(s_F P^*_i, S_F P^*_i)$, as shown in Figure 1.

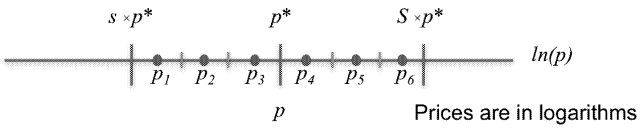


Figure 1: Example of Initial Distribution of Prices

Additionally, prices satisfy $P^*_i = P$. Similar conditions are used for wages.

Defining $\psi = \sum_{i=1}^{m} \left(s_F k_F^{i-\frac{1}{2}} \right)^{-\alpha\theta}$, initial state of the economy is as shown in Table 1.

The value of s_F must satisfy $m = \sum_{i=1}^{m} \left(s_F k_F^{i-\frac{1}{2}} \right)^{1-\theta}$, and is found using numerical methods. The value of S_F is given by $S_F = s_F k_F^m$.

Table 1: Initial State of the Economy

Variable	Value
Y	$m \left[\left(\frac{n}{\psi} \right)^{1-\beta} \frac{1}{\alpha} \frac{\mu}{\beta} \frac{\theta-1}{\theta} \frac{\sigma-1}{\sigma} \right]^{\frac{1}{\alpha\beta-1}}$
L	$\left[\frac{Y}{m} \right]^{\alpha} \psi$
P	$\frac{\gamma}{\gamma-1} \frac{M}{Y}$
W	$\frac{\beta}{\mu} \frac{\sigma}{\sigma-1} P \left(\frac{L}{n} \right)^{\beta-1}$
P_i	$k_F^{i-\frac{1}{2}} s_F P$
W_h	$k_H^{h-\frac{1}{2}} s_H W$

Specification of Expectations

Optimal pricing depends on expectations of P, W, Y and L. Additionally, household's consumption decision requires a forecast of income I. In equilibrium conditions, nominal variables (such as P, W and I) are expected to grow at the rate of inflation, and real variables (such as Y and L) are expected to be constant. Therefore, agents will forecast first group of variables based on the forecast of inflation, and the second group of variables based on a forecast of the change in Y.

Considering the aggregated relation of product and money $Y_t P_t = \sum_{i=1}^{n} \gamma I_{i,t} \approx \frac{\gamma}{1-\gamma} M_t$ (approximation is due to errors in estimation of income), taking logarithms we can estimate the change in product as

$$E[\Delta y_t] = \bar{\pi} - E[\pi_t]$$

Therefore, agents will forecast group of nominal variables based on the expectation of inflation, and the real group of variables based on the stimation of the change in Y.

Adaptive Learning is used to form inflation expectations. The perceived law of motion used is $\pi_t = a_{0,t}$, with initial value $a_{0,0} = \bar{\pi}$.

EXPERIMENTATION AND RESULTS

Parameter Values

Parameters used in the simulation are shown in Table 2. Value of γ is based on (Holman 1998). Values of α, β, θ and σ where taken from (Blanchard and Kiyotaki 1987). Values of m, n, and M are arbitrary. $\bar{\pi}$ is calibrated to represent moderate inflation (3%). Each time period is considered as a week. Value of s_F is calibrated to approximate a frequency of 3 price changes a year. Then, S_F/s_F will be close to 1%.

Table 2: Parameter Values Used in the Simulation

Parameter	Value	Parameter	Value
γ	0.05	m	20
α	1.2	n	100
β	1.4	M_0	100
θ	5	$\bar{\pi}$	$1.03^{\frac{1}{56}} - 1$
σ	5	s_F, s_H	0.995

Behavior of the Output Gap

Initially, we tested the behavior of the output gap (deviation from equilibrium value), considering distribution preserving price updating (P). Results are shown in Figure 2 for several values of the gain parameter. Particularly, the case of $\phi = 1.0$ is equivalent to backward looking expectations, and the case $\phi = 0.0$ represents fixed expectations. Results show the emergence of cyclic dynamics. Additionally, oscillations of output are stronger the higher the value of the gain parameter.

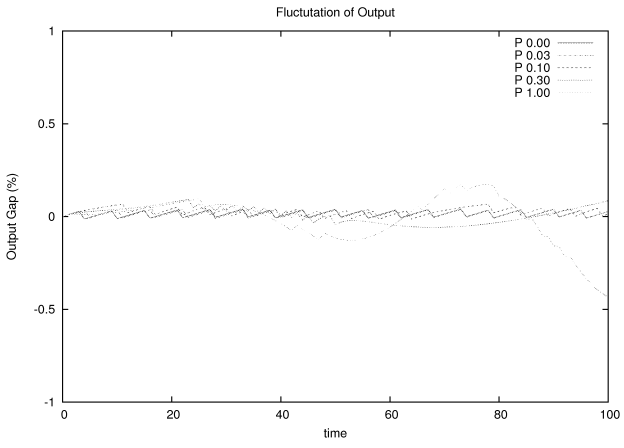


Figure 2: Dynamics of Output Gap for Distribution Preserving Price Updating

Then, we proceed to test the behavior of output for distribution non-preserving price updating (N). Results are shown in Figure 3. Results show that in the case of distribution non-preserving price updating the oscillations of output are stronger. This is due to synchronization of prices.

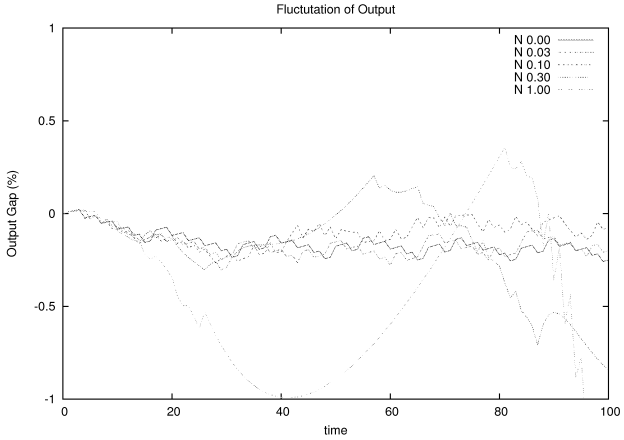


Figure 3: Dynamics of Output Gap for Distribution Non-Preserving Price Updating

In Figure 4 output behavior is shown for a longer period of time. It shows how fluctuations become stronger with time and tend to a limit cycle.

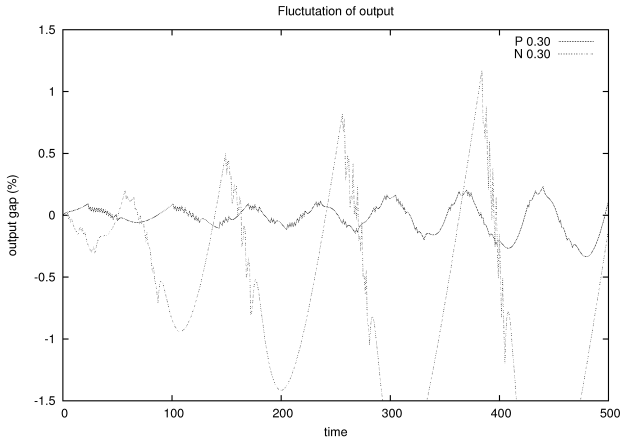


Figure 4: Comparison of Long Term Dynamics of Output Gap for Distribution Preserving and Non-preserving Price Updating

It is important to notice that the model is deterministic. No external stochastic shocks are applied to the model. Therefore, this results show how the effects of discrete time and a finite number of agents may take an economy from a steady state to a richer dynamics.

Rationality of Expectations

In order to test how similar to Rational Expectations are the forecasting rules used in this paper, we performed a test of unbiasedness taken from (Pfajfar and Santoro 2010) and (Mankiw et al. 2004). Having the regression model $\pi_t = a_0 + a_1 E[\pi_t]$, Rational Expectations imply $a_0 = 0$ and $a_1 = 1$. We tested this joint hypothesis and obtained the p-value for several values of the gain parameter, running the model for 200 periods. Results are shown in Figure 5.

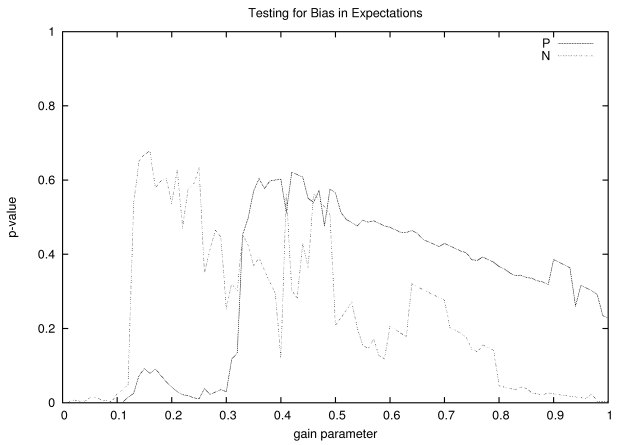


Figure 5: p-value of Unbiasedness Test for Several Values of the Gain Parameter

Results show that unbiasedness is rejected for small values of ϕ but not for larger values. This result is surprising, since lower values are associated with smaller fluctuations of output and therefore closer to the original model, and also smaller values of ϕ generate inflation expectations that are closer to the actual rate of monetary growth.

CONCLUSIONS

This paper evaluated the results from (Caplin and Spulber 1987) in a simulation setup. Agent based simulation allowed to evaluate the consequences of discrete time and finite number of agents in the stability of (Caplin and Spulber 1987) results.

We considered two implementations of (s, S) strategies, one that preserves the distribution of real prices and one that does not. We showed that this factor has an important influence in the behavior of the model: If the price updating mechanism is not preserving, prices tend to synchronize and oscillations in output are stronger. It should be noticed that this fluctuations are endogenous, and not the result of external shocks.

A test of unbiasedness taken from (Pfajfar and Santoro 2010) was used to test rationality of expectations. Results showed that unbiasedness can be rejected for small values of the gain parameter, but not for intermediate or high values. Thus, these forecasting rules behave approximately rational under these money process. This result is surprising, as lower values of the gain parameter generate behavior closer to the original analytical model.

Additional experimentation (not reported due to space limits) shows that the rationality of expectations is affected by the money process. Shocks to the rate of money growth changes the values of the gain parameter for which expectations can be described as rational.

Further experimentation will be directed to testing sensibility of the results to parameter values and to evaluate the results under different money processes.

REFERENCES

Arifovic J. and Bullard J., 2001. *Introduction to the Special Issue: New Approaches to Learning in Macroeconomic Models. Macroeconomic Dynamics*, 5, no. 02, 143–147.

Blanchard O.J. and Kiyotaki N., 1987. *Monopolistic Competition and the Effects of Aggregate Demand. American Economic Review*, 77, no. 4, 647–66.

Caplin A.S. and Spulber D.F., 1987. *Menu Costs and the Neutrality of Money. The Quarterly Journal of Economics*, 102, no. 4, 703–25.

Holman J.A., 1998. *GMM Estimation of a Money-in-the-Utility-Function Model: The Implications of Functional Forms. Journal of Money, Credit and Banking*, 30, no. 4, 679–98.

Mankiw N.G.; Reis R.; and Wolfers J., 2004. *Disagreement about Inflation Expectations.* National Bureau of Economic Research, Inc, NBER Chapters. 209–270.

Pfajfar D. and Santoro E., 2010. *Heterogeneity, learning and information stickiness in inflation expectations. Journal of Economic Behavior & Organization*, In Press, Corrected Proof, –. ISSN 0167-2681.

Sheshinski E. and Weiss Y., 1977. *Inflation and Costs of Price Adjustment. Review of Economic Studies*, 44, no. 2, 287–303.

Walsh C.E., 2003. *Monetary Theory and Policy, 2nd Edition.* The MIT Press, 2 ed. ISBN 0262232316.

STATISTICAL TOOLS FOR CONSOLIDATION OF ENERGY DEMAND FORECASTS

Vincent Micali
ESKOM
Johannesburg, South Africa
email: Vince.Micali@eskom.co.za

Igor Litvine
Abel Motsomi
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa
email:igor.litvine@nmmu.ac.za
email: amotsomi@nmmu.ac.za

KEYWORDS

Energy demand, Forecasting, Consolidation

ABSTRACT

Eskom, as the major electricity provider for the South African economy, needs specialised instruments in forecasting the energy load to be delivered. The current status quo operates with several forecasters from different offices for different purposes. This becomes a challenge to consider, since these forecasts need to derive a "consolidated forecast" for the utility. This paper has attempted to develop a consolidating instrument that will merge all the forecasts from different offices to one *'official forecast'*. Such an instrument should be able to predict with accuracy the anticipated usage or demand. Discrepancies (such as the granularity of data) between the forecasters have created a need for an aggregating instrument. The results presented in this paper present a newly developed procedure of consolidating energy demand forecasts from different users and accounting for different time horizons. Predicting for very-short term, short term and medium term involves different measuring grains which is one aspect this paper tackles.

1. INTRODUCTION

Eskom generates approximately 95% of the electricity used in South Africa and approximately 45% of the electricity used in Africa. Eskom generates, transmits and distributes electricity to industrial, mining, commercial, agricultural and residential customers and redistributors (www.eskom.co.za).

Eskom currently finds itself in a position where the demand for electricity may exceed the available supply from time to time. In order to manage the situation in the best possible way, the predictive ability becomes of paramount importance.

2. PREVIOUS STUDIES

The first question to be asked in this study is; why consolidate? Perhaps, this is a fair question given that most people would rather have authority in calling the shots in anything than give group input and wait for a combined decision. In his book The wisdom of crowds, Surowieki (2004) shows that aggregation of information in groups results in better decisions that are often better than could have been made by any single member of the group. The author continues to show that the group decision is more reliable than the output by any one think tank (person). This justifies the adoption of consolidating, or rather, aggregating forecasts to generate a more reliable and one accurate official forecast.

This paper assesses different ways of consolidating estimates (forecasts). These are; least squares, minimum variance optimal estimate and newly proposed consolidation instrument.

In Surowieki (2004), the author state that some of the advantages of least squares are that combination of different observations taken under the same conditions yields a better result as opposed to simply trying ones best to observe and record one observation accurately. Though the least squares have its advantages, it also has some disadvantages. In Mendenhall and Sincich (2003), one pitfall is that the solution reached is not robust. Another three pitfalls are categorized below; (1) Problem 1: Nonnormal errors: the standard least square assumption of normal errors is violated since the responses y and random error e can take on only two values. For more on this problem, see Mendenhall and Sincich (2003). This problem can be solved by using a sample size n that is large. (2) Problem 2: Unequal variances: it can be shown that the variance s2 of the random error is a function of ρ, the probability that the response y equals 1. Specifically, true for linear models as $\rho = E(y) = \beta_0 + \beta_1 x_1 + . + \beta_k x_k$ since $\sigma^2 = V(\epsilon)$ = p(1-p). (3) Therefore, this implies that σ^2 is not a constant and in fact depends on the values of the independent variables, hence the least squares assumption of equal variances is violated.

3. EMPIRICAL RESULTS

One advantage of using Mathematica as an analysis tool is the vast capabilities it has of with ease the different analysis tools such as for evaluating the strength of the relationship between variables. Comparison was made between the forecasts of each forecaster and the actuals or observed electricity values by calculating the correlation coefficient ρ. Results are posted in Table 1 below

	Observed
Forecaster 1	0.94953
Forecaster 2	0.95474
Forecaster 3	0.96591
Forecaster 4	0.97399
Forecaster 5	0.98313
Forecaster 6	0.98681
Forecaster 7	0.9904

Table 1: Correlation coefficients

Despite the high correlations between forecasts and the observed energy demand, which in itself is a good outcome, the results also concur with the interpretations that Forecaster 7 is the most efficient one as his forecasts are almost perfectly positively correlated to the actuals. Note that Forecaster 7 only predicts a day ahead, therefore, has more probability of being accurate that any other forecaster. Based on the high correlation we noted from the forecasts and the observed, we can then base our rationale for the proposed procedure.
We will call the proposed procedure the Litvine-Motsomi Split (LM split). The first task is to dissect the time intervals into categories; ie, the intervals would then be used as a point index for that period.
The LM split has 4 horizons, namely;

- Extra Short Grain ESG
- Short Grain SG
- Medium Grain MG
- Long Grain LG

For the purposes of this study, the aim is to consolidate forecasts made by different sources to optimally generate one consolidated forecast. To cater for possibility of sources having a specific mandate and focus, the LM split would assist in uncovering and rectifying such a mix-match of different focuses.
In the LM split,

$$ESG = \frac{\mathrm{f_{it}}}{\sum_{\mathrm{t=1}}^{1} \mathrm{a_t}} \tag{1}$$

$$SG = \frac{\mathrm{f_{it}}}{\sum_{\mathrm{t=1}}^{24} \mathrm{a_t}} \tag{2}$$

$$MG = \frac{\mathrm{f_{it}}}{\sum_{\mathrm{t=1}}^{168} \mathrm{a_t}} \tag{3}$$

$$LG = \frac{\mathrm{f_{it}}}{\sum_{\mathrm{t=1}}^{672} \mathrm{a_t}} \tag{4}$$

where f_{it} is a forecast of forecaster i at time t, a_t is actual energy observation at time t.
Assumptions of the LM split;

- All other variables remain the same, ceteris paribus.
- No extra-ordinary events/phenomena should have occurred for the time in question.

In this study, we looked at hourly data and made the split according to that. Where ESG was hour on hour, SG was hour on day, MG was hour on week and lastly, LG was hour on month.
The LM split yielded some good results summarised in Table 2 and Table 3 to explicitly show the distribution of the patterns.

Summer weights		Winter weights	
hour	percent	hour	percent
1:00	3.47%	1:00	3.30%
2:00	3.43%	2:00	3.24%
3:00	3.41%	3:00	3.23%
4:00	3.42%	4:00	3.27%
5:00	3.52%	5:00	3.41%
6:00	3.86%	6:00	3.82%
7:00	4.16%	7:00	4.35%
8:00	4.26%	8:00	4.41%
9:00	4.44%	9:00	4.48%
10:00	4.46%	10:00	4.48%
11:00	4.58%	11:00	4.56%
12:00	4.59%	12:00	4.57%
13:00	4.54%	13:00	4.53%
14:00	4.51%	14:00	4.50%
15:00	4.51%	15:00	4.52%
16:00	4.55%	16:00	4.58%
17:00	4.55%	17:00	4.68%
18:00	4.45%	18:00	4.88%
19:00	4.39%	19:00	4.87%
20:00	4.59%	20:00	4.66%
21:00	4.55%	21:00	4.44%
22:00	4.23%	22:00	4.10%
23:00	3.89%	23:00	3.70%
0:00	3.63%	0:00	3.43%

Table 2: Distribution patterns of load profiles

After consideration of the distribution patterns of the load profiles hourly and daily, it was possible to then create some aggregated forecast based on those patterns. This precipitated the development of the following models.

Summer weights		Winter weights	
Monday	14.61%	Monday	14.08%
Tuesday	14.79%	Tuesday	13.53%
Wednesday	14.55%	Wednesday	15.32%
Thursday	14.63%	Thursday	15.28%
Friday	14.65%	Friday	14.77%
Saturday	13.71%	Saturday	13.78%
Sunday	13.06%	Sunday	13.23%

Table 3: Distribution patterns of load profiles

Model 1 is a linear function of:

$$\text{Official forecast} = \alpha_1 x_1 + \alpha_2 x_2 + \beta_1 y_1 + \delta_1 z_1 + A$$

Let:

- x_1 and x_2 be values from forecasters for hourly grains;
- y_1 is a value from a forecaster for daily grains;
- z_1 is a value from a forecaster for weekly grains;
- α_1 is a ESG coefficient from time a_{t-1}
- α_2 is a ESG coefficient from time $\sum_{t=0}^{-23} a_t$
- β_1 is a SG coefficient from time $\sum_{t=0}^{-167} a_t$
- δ_1 is a MG coefficient from time $\sum_{t=0}^{-671} a_t$
- A is a constant.

The rationale of the model above is to accommodate the existence of other grains and thus includes other parameters for forecasters who predict for different time horizons. The same is not true of the following model 2 as the idea is to visualise the potential for only considering the same bigger grains such as daily and weekly alone. This is further motivated by the existence of the commonly used patterns for daily and weekly profiles.
Model 2 is also a linear function of:

$$\text{Official forecast} = a_1 x_1 + a_2 x_2 + a_3 x_3 + a_4 x_4 + A$$

Let:

- x_1, x_2 , x_3 and x_4 be values from forecasters for daily grains;
- a_1 is a SG coefficient from time $\sum_{t=0}^{-23} a_t$
- a_2 is a SG coefficient from time $\sum_{t=-648}^{-672} a_t$
- a_3 is a SG coefficient from time $\sum_{t=-624}^{-648} a_t$
- a_4 is a SG coefficient from time $\sum_{t=-672}^{-696} a_t$
- A is a constant.

All the parameters are from daily grains only. For this paper, we dwell predominantly on the short term forecasting, ie, hourly, daily and weekly.
Some good results were achieved where it was determined that a forecaster who forecasts a day ahead was competently better than any other, reason being that he has all the information at his disposal than one who forecasts a week ahead as there are more significant variations and uncertainties likely to occur. A 'consolidated forecast' is statistically a better and optimal prediction because over a long term it has smaller variance and mean of residuals than any forecaster. Both models show that for a specific time horizon a consolidation of different forecasters with different grains adds value in the predicting power of the model and adaptable and extensible to the number of forecasters.

4. CONCLUSIONS AND FUTURE STUDIES

In this paper, we have seen that to be successful in consolidation, a correct split of time horizons assists in the process (represented by the LM split). The results of the study were consistent with the general theory in the literature. The study was able to produce individual profiles for each forecaster for daily, weekly and monthly profiles. The newly proposed LM split addresses most of the aspects of forecast consolidation which is the primary object of this study. However, there are gaps which future studies may pinpoint as their priorities, these include; (1) incorporating adjustments in the model, (2) incorporating adjustments by management (coupled with experience).

REFERENCES

Bloomsfield, P., Steiger, W. 1980. *SIAM Journal on Scientific Computing.* Least Absolute Deviations: Curve-fitting. Vol 1,pg 290 -301.

French, S., Smith, J.Q. 1997. *The Practice of Bayesian Analysis.* Arnold publishing.

Lindgren, B.W. 1971. *Elements of decision theory.* Macmillan

Lindley, D.V. 1985. *Making decisions.* 2nd Edition. John Wiley and Sons

Mendenhall, W., Sincich, T. 2003. *A second course in Statistics: Regression Analysis.* Prentice Hall.

Surowieki, J. 2004. *Wisdom of crowds.* Prentice Hall.

www.eskom.co.za

A MULTI-AGENT TOURISTIC CATERING MARKET MODELING METHODOLOGY TOWARD A DSS

A NEW APPROACH BASED ON MULTI-AGENTS AND GEOGRAPHIC INFORMATION SYSTEMS

Dominique Urbani
Marielle Delhom
Stephane Garredu
SPE Labs – UMR CNRS SPE 6134, University of Corsica
Campus Grimaldi, 20250, Corte, France
durbani@laposte.net
delhom @univ-corse.fr
garredu@univ-corse.fr

KEYWORDS
Multi-agent system, agent, model, simulation, geographic information system, environment, tourism, decision support system, mediterranea, corsica.

ABSTRACT

In this paper we present a methodology to build a multi-agent tool dedicated to manage the tourism industry. Linked with a Geographic Information System, our Individual Based Model of the catering market enables to understand how each economic players change to be in accordance with the demand, and foresee consequence of decision makers' choices. Using a GIS and individual models of micro economics players and tourists, our experimentations based on Corsica's ground data join both the predictions of macroeconomic theory and the field observations.

I INTRODUCTION

The combined progress in agriculture and factor industry, the globalization of the economy, the end of protectionism, the scale gains dues to the mergers of world global companies contribute to improve the effectiveness of the primary and secondary sectors of the economy. These sectors reduce their relative weight in the gross world product in aid of services.

With a 7% average growing rate since 1950 (UNWTO 2006), the world travel industry is one of the most dynamic sector in the economy. Leisure and travel industry constitutes a strong economic powerhouse for numerous other activities such as transportation, building, civil engineering, entertainment and media.

In addition to global factors and due to their location out of the main economic flows, geography, lack of natural resources, more and more area are becoming fully dependant to their tourism industry. Thus Mediterranean big islands of Corsica, Baleares, Sardinia, Sicilia, Malta, Crete, and Cyprus are now completely dependent to tourists flow. Moreover in 2010, the World Travel & Tourism Council inventoried 5 countries where more than 50% of the GDP are coming from tourism, and reveled that in 9 counties more than 50% of employments depend of tourism (WTTC).

Clear headed about their situation and about the societal and economic risks due to a high dependency of tourism (Pandey), decision makers are looking for ways to take advantage of the current situation to develop other sectors of their economy (UNEP) (CE). Public policies are often engaged to limit the number of visitors and to increase the average income per visitor aiming at make financial resource available to sustain the creation of non tourism dependant economic and sustainable activities (MTT).

Thus the leaders concerned by this problematic are in quest of dedicated tools to understand at small scale the mechanisms of this system and to foresee the possible consequences of public interventions on the local tourism business. These tools should be useful to find new ways to take advantage of current tourists flow and to influence together the number and the profile of visitors.

In continuation of our works in modeling complex and heterogeneous system (Urbani 2008b) we propose in this paper a methodology based on a distributed artificial intelligence paradigm and a geographic information system. Thus we present an individual based model of the mutual influence between tourists and catering players. To build our model we use studies about the tourism industry, about microeconomic theory, about tourist taxonomy and archetypes (Caccomo) (Fink). This model is the core of a simulator / decision support system dedicated to foresee the influence on local economic players of public efforts to inform tourists and promote a region.

This paper is organized as follows. In Section 2 we present our methodology based on the use of a framework for hybrid decision support system using a multi-agents system and a geographic information system. Section 3 is dedicated to the models and archetypes we develop. In Section 4 we present our first simulations and experimentations. Finally in Section 5 & 6, we discuss our works and conclude the paper.

II METHODOLOGY

A. Methology step by step

The first step of our work is to understand the problems induced by an excessive development of tourism in some areas. Then we focus on ways to provide help to decision makers trying to predict the adaption of local economic players to changes and stakeholders' decisions. Thus we

draw the specifications for a multi-agent model and a decision support system.

Once the aims defined, focusing on qualitative aspects, we analyze the tourism market linking together the products' characterization, the individual tourists' choices process, the stakeholders' influences and behaviors, and the economic and identity evolutions of local community.

Utilizing our previous works on hybrid MAS GIS decision support systems and our SMAG CORMGIS platform (Urbani 2006b & 2008b) we build an individual based model of the tourism economy focusing on catering.

B. Model and DSS specifications

Decision makers facing to the management of tourism industry in region similar to Mediterranean islands dispose of macroeconomic and regulations ways to influence the evolution of their territory. Nevertheless, in democratic and free economic areas, these managers are elected and can't impose their will to individual.

Because decision makers want to respect individual liberties and want to be reelected, they wish tools to predict the consequences of their choices singly for each economic player. Thus the specifications of our decision support system must include the possibility to foresee at individual scale:

- The adaptation process of local players of tourism industry facing to changes.
- The effects of stakeholders' actions of information toward travelers.
- The impact of a new demand ensuing from changes of transportation offer and destination's corporate image.
- The importance of qualitative aspects.

Because decision makers need data fitted to each situation, we focus on the individual scale and our specifications include data from geographic information system.

C. Focus on qualitative aspects of tourism market

Due to the fall of transportation costs and the expansion of the middle class, the tourism market is now a growing world mass market. Because tourism needs time and money to travel and because this market deals goods that are not essential to life, the tourism market is a luxury market (Caccomo).

Travels are complex products, mix of transport, hosting, catering, entertainment, landscape, environment, heritage, and services. Undeniably, due to an evident lack of details and to the asymmetry of information, customers can't properly evaluate a journey before live it. Thus travels can only be evaluated at posteriori, at the end of the trip. Moreover these evaluations are subject to the subjectivity of each person with its own criterions and subjectivity and that make difficult for a candidate traveler to use others' recommendations. Because before its consummation, a precise estimation of the value that a traveler makes of elements composing a trip can't be done, it's difficult to make individual behavioral predictions.

Studies about the travel market revealed that two criterions are really important in the consumer's choice:

1. *The length* is the major element in the evaluation of the trips. Travelers use time to measure the "quantity" of travel they might buy. Moreover the satisfaction issued form a trip is directly correlated to its length.
2. *The localization* of trip is the second most important element used by consumers to estimate a travel's value. This criterion is not only linked to geography: the distance is also psychological, cultural and social.

The taxonomy of the goods and services offered by the tourism industry follows two lines of division (Caccomo) :

1. *Artificial-Standard vs. Quaint-Authentic*

   Artificial-Standard goods and services are available in numerous places in the world; they are not linked with the localization of the trip. Western style accommodations and restaurants, entertainment parks, ski areas, cruises, all-inclusive resorts are examples of these tourism industry's cheap products available all around the world to respond to the demand for low cost journeys. This market of standardized travels is a worldwide competition where prices and profit margins are falling.

   Quaint-Authentic goods and services are closely linked to their localization. The Empire State Building, the Louvre, the cable car of the "Aiguille du Midi" and its "Vallée Blanche" ski area, Venice, the October Fest in Munich are few examples of such unique and authentic attractions. Due to their rarity and lower competition intensity, these products are more expensive than the standard ones. Attracted by the symbolic and cultural load carried by these places, many travelers accept to pay more for these destinations.

   Thus most decision makers and stakeholders want to develop an "authentic" offer to preserve their profit margins and reduce their exposure to the worldwide competition in standardized products market. To develop an authentic product these destination must take several measures: limit the number of visitors, protect the environment, provide good transportation and telecom networks, guaranty health and security, and preserve the local people.

2. *Discovery vs. Escape destinations*

   According to Caccomo travels trips can be classified between discovery and escape trips. The distinction is important to understand the birth of tourists' satisfaction sentiment.

   Trips and travels belonging to the "discovery" class are characterized by a satisfaction sentiment following a bell function of time.

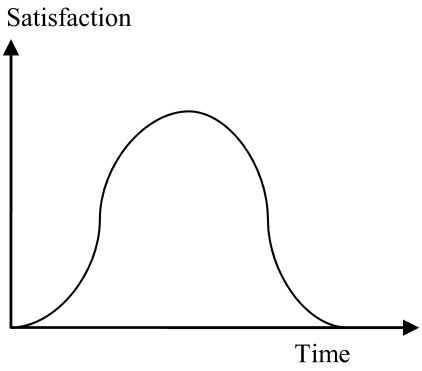


Figure 1 : Satisfaction of tourist during a discovery trip

Figure 1 shows how tourists feel satisfied during discovery entertainment: first the satisfaction grows discovering something new, reaching a peak after a short time, tourists wonder is filled, and finally they become "blasé" and bored. Museum, merry-go-round, small cities like Nafplion are examples of "discovery" trips. Once the discovery has been done the charm quickly disappears.

Trips and travels belonging to the escape class are characterized by a satisfaction sentiment always rising with time.

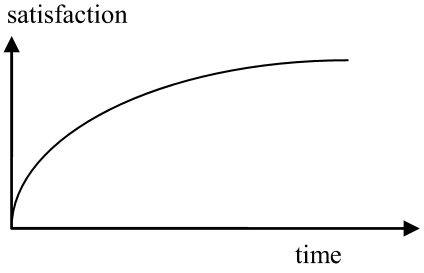


Figure 2 : Satisfaction of tourist during an escape trip

Figure 2 shows how tourists feel always more satisfied during escape travels. Tahiti, Patagonia, and Corsica are some examples of what most of urban westerners consider as escape travels: these destinations satisfy their need to change of scene and to break with their daily routine and environment. In these places, people feel good, don't become bored, and they are sad when they must leave to go back to their home. In such destination, trips are longer than with standard products, people stay more time and accept to pay more for their vacations.

D. Hybrid MAS GIS modeling framework

The modeling methodology exposed in this paper use our SMAG (Système Multi-Agents Géographique) platform architecture dedicated to the development of hybrid MAS-GIS decision support systems (Urbani 2006b) developed at the University of Corsica. This architecture, based on a multi-agents approach of decision support system, was specially fitted to model and to simulate heterogeneous systems with a strong spatial component and complex interactions between numerous stakeholders such as fresh water management (Urbani 2006a), coastal ecosystem (Urbani 2008a), .

SMAG is a modular hybrid architecture based on 2 units:

- The first unit is dedicated to model the behaviors of the systems and the stakeholders according to a multi-agents system (MAS) approach.

- The second unit based on geographic information system (GIS) enables to take into account the spatiotemporal data of each targeted sites. It provides a suite tools usable to define scenarios of simulations and to plan the experimentations.

In this paper, we use our operational platform complying with the SMAG architecture (Urbani 2006b), CORMGIS, to implement the players of the catering system in tourism business.

CORMGIS is a modular operational development platform dedicated to build decision support systems focused on natural renewable resource management, environmental issues and societal systems. It's based on the Visual Works programming environment which enables the development of applications in the object-oriented programming language Smalltalk.

The CORMGIS multi-agent module is based on extensions of the Agricultural Research Centre for International Development (CIRAD) CORMAS (COmmon-pool Resources and Multi-Agent Systems) multi-agents platform (Bousquet), whereas its GIS module is implemented by ArcGIS, the ESRI company (Arcgis) GIS software suite. The CORMGIS extensions of CORMAS we developed enables to plan experimentations, to connect its core simulator to the geographic databases, and to provide graphic interfaces dedicated to the interactions with stakeholders, experts and decision makers.

CORMGIS pre-defined entities are Smalltalk generic classes from which, by specialization and refining, developers can create specific entities to model the players of their system.

The dynamic link between the MAS and GIS CORMGIS modules enables to take into account the characteristics of each case and to fit the simulations to each situation (Westervelt) (Koch).

To instance and initialize the agents and the elements composing the individual based model of the system our platform uses the geodatabase.

The Simulator/DSS can take into account the predefined events supposed to occur during the simulations. Before computations, using a dedicated toolbox joint to CORMGIS, the decision makers define the events constituting scenarios. Thus it makes possible to plan experimentations and explore numerous hypotheses.

For further analysis the data generated by computations are stored in the geodatabase managed by the GIS module. Each event occurring during the digital experimentation is stored by the GIS. Scenarios, agents, and decision makers/experimenters are both at the origin of events.

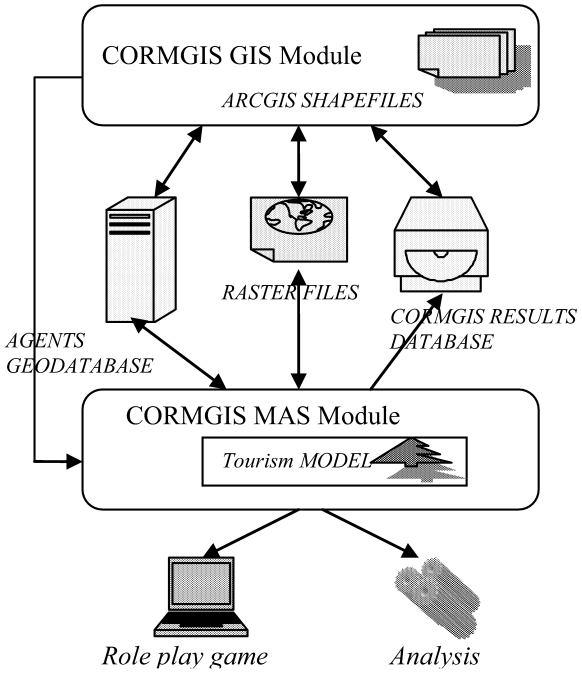


Figure 3 : CORMGIS platform

We previously first used CORMGIS to build a decision support system dedicated to the fresh water problem and to highlight the influence of the social interactions during the water shortage in the north of Corsica (Urbani 2006 a&b) Furthermore we also developed a DSS dedicated to the interactions between watershed and activities in coastal areas focused on sea farming (Urbani 2008a) In this paper we present models and experimentations in continuation with our previous works about a methodology to model the Mediterranean backcountry tourism economy (Urbani 2009)

III MODELING

A. Patterns

In previous works (Urbani 2009) we developed a library of models representing the different types of stakeholders and environment in a touristic area: day tripper, camper, hiker, hotel host, motel host, bicycle touring, packaged tour, camping-hotel-motel manager, bar-restaurant owner, village council, county council.

In this work dedicated to catering, we model the system using patterns : T_Tourist for travelers, T_Restaurant for catering players, NSpot-HSpot-ESSpot for sightseeing, T_Guide to inform travelers, T_DM for decision makers.

B. Modeling Touristic Area

The multi-agents formalization of the ground environmental system is represented by a regular grid composed by instances of GroundParcel pattern. This pattern is refined from a generic SpatialEntityElement agent provided by the CORMAS platform. A GroundParcel agent represents a homogenous ground area taking into account its land cover, its own dynamic and humans' works.

T_Guide agent pattern is used to model a source informing visitors about the touristic area. The tourist agents use Guides to update their representation of the world and to plan their trip. The Guide agent pattern takes into account the media intrinsic credibility, the interest and the location of each referenced entities. This pattern is used to instance tourist information offices, road signs, notice boards, booklets, road map and travel guides.

We model the sightseeing and entertainment players taking into account: appeal value, the cost and the time to visit. Refining a CORMGIS we develop 3 archetypes: NSpot for natural sites, HSpot for heritage, ESpot for standard entertainment players.

C. Modeling catering offer

As all elements of tourism products and services, catering has been widely studied, analyzed and numerous classifications built (Caccomo) (Fink). Criterion used to characterize a restaurant are numerous, subjective and reveal the observer's centers of interest.

According to our goals we choose four attributes to characterize catering economic players:

1. The middle price of menus, reflecting both the social status of customers and the food quality. Prices vary from 5 to 200 Euros.
2. The length of the meal: time is a major criterion in customers 'choice. This attribute varies between 15 to 120 min.
3. The restaurant's standardization degree. Less the meal are standard, more it contributes to make authentic the travel destination and to reinforce the destination as an escape trip. This attribute varies between 0 for standard food like Mc Donald's to 1 for typical restaurant.
4. The number of tables, indeed this is a major element conditioning the tourist's perception of restaurant and destination. Big players reinforce the visitor's impression to be in a standardized discovery destination.

We build the T_Restaurant pattern, refining the CORMAS AgentCommLocation, to model a catering economic player. A T_Restaurant agent is rational and acts to maximize its sales. Thus, analyzing its customers, it computes its new prices and modifies the meal's length and its degree of standardization.

D. Modeling tourist

Tourists, as all consumers, have been studied in depth to understand and to satisfy their wishes. Since decades, the demand of travel has been segmented by economic players in order to provide the right product to each category.

Coming with internet advent, a lot of works focused on studying and modeling individual traveler in order to help him to choose a product fitted to his wish and to provide him real time assistance through GPS, mobiles devices and internet. Moreover the multi-agents paradigm has been used to develop virtual sellers for online travel agencies and personals assistant providing advisory and assistance services to travelers (Fink).

In this paper we present our works on modeling travelers during their trips using a multi-agents approach in order to understand how individual visitors and services providers behaviors interacts and build the public image of a destination.

According to our goals we choose six attributes to characterize a tourist looking for catering:

1. The tourist's wished price of the meal.
2. The weight of the meal price in the choice process, it varies from 0 to 1.
3. The length of the meal time.
4. The weight of the meal time in the choice process, it varies from 0 to 1.
5. The aspiration for an authentic or standard food. This attribute varies between 0 for standard food like Mc Donald's to 1 for typical restaurant.
6. The weight of the meal type in the choice process, it varies from 0 to 1.

Looking for catering, traveler computes a score for each restaurant using his knowledge. The probability for a restaurant to be chosen is proportional to its score. Moreover, each time a restaurant enter in the traveler's area of perception, he updates its score taking into account the number of people currently eating in the restaurant and the neighborhood: more people are eating in the restaurant, more it's attractive. Nevertheless when it becomes crowded its score falls.

We build the T_Tourist pattern, refining the CORMAS AgentCommLocation, to model a traveler.

E. Modeling decision makers

We build the T_DM pattern, refining the CORMAS AgentCommLocation, to model decision makers. These agents influence the system edicting rules, modifying information toward visitors and providing grants. Thus they can:

- Trough their power in transportation, they can modify the structure of the touristic demand.
- Grant some economic players to influence some types of offer.
- Control the local information (T_Guide) of tourist through road signs, booklets and publicities.

We take into account decision makers using T_DM agents for planed experimentation and trough the CORMGIS graphic user interface facilities for real time simulations.

IV RUNING SIMULATION

A. Framework

Experimenters and decision maker can exploit our multi-agents model using CORMGIS under two modes: "Analysis" and "Role Play Game".

Used in the "Analysis" mode CORMGIS core simulator explores predefined scenarios and follows the experimentations planned by users. At the initialization, experimenters define domains for each system's key attributes (fixing the limits min max step) and simulation is run for each combination of attributes' values. During computations, experimenters cannot modify scenarios or models. For any simulations, any time step, the results and the data relevant for users are stored in a dedicated database for further analysis.

The "Role Play Game" mode of CORMGIS enables a total control on current simulation, models and behaviors. We develop dedicated graphic interfaces to interact during experimentations like in a role play game. Developed using Smalltalk, the simulator and Evisa's model enable hot modifications during experimentations and don't require a new compilation or a re-initialization.

In both modes, all events occurring during simulations are registered in the geodatabase to be used to define new scenarios or analysis. Thus, role play game events can be reused for more systematic experimentations using the "Analysis" mode.

B. Area of Evisa

Our first experimentations using our multi-agents model are focused on the study of a flow of visitors looking for catering and sightseeing in the Corsican village of Evisa, 17 km far from the sea by road, altitude 800 meters. Population varies from 100 inhabitants in winter to 900 residents during the touristic season. Each day, during the august peak of frequentation more than 4000 people can cross this village.

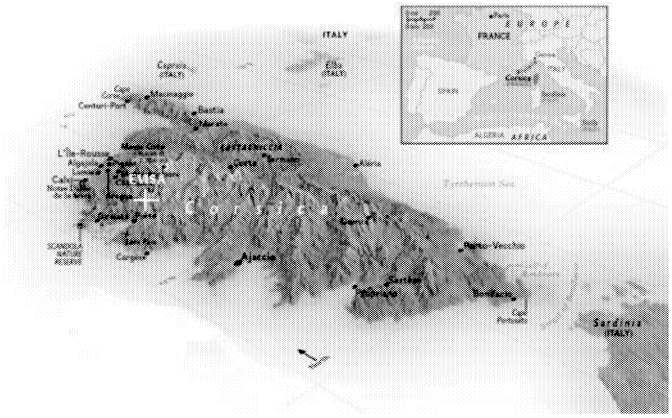


Figure 4. Map of Corsica Island

Figure 5 shows a CORMGIS interface's detail enabling to supervise agents during simulations in a 4x3 km area centered on Evisa. On this figure are located E_RestoBar agents (green points), NSpot sightseeing spot (brown points), T_Guide agents (big black triangles), and 125 T_Tourist agents (small colored triangles). The cells are GroundParcell agents represented using the land covert point of view.

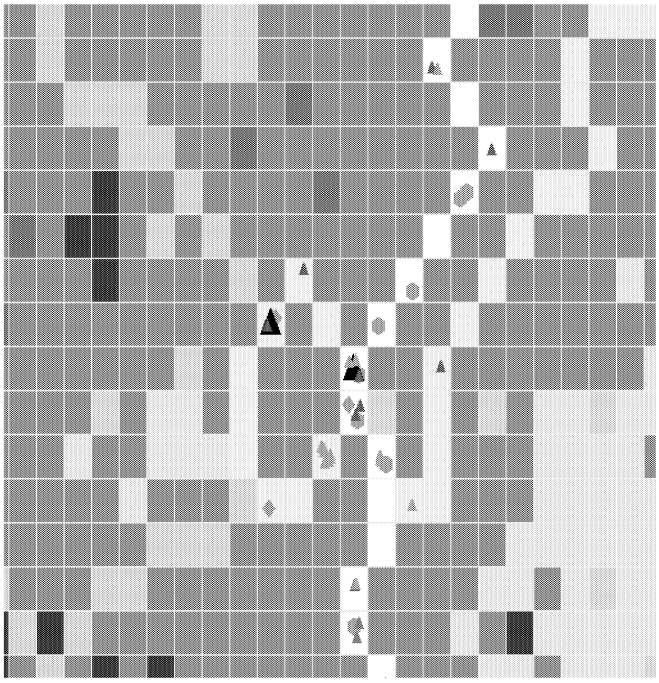


Figure 5: CORMGIS interface's detail of Evisa area

C. Warm up and equilibrium

Before our first experimentations we run the simulator forcing the tourist's demand to a constant. During this period catering agents modify their profile (price, time, authenticity) to maximize their sales and profits.

We select several scenarios and run simulations during 10000 steps until the system reaches a stationary state. Because agents are stochastic, we repeat 100 times each scenario, but even if trajectories are different, finals stationary states set up a small group of combinations.

Forcing tourists' demand and agents' initial state we obtain a set of data. These Warm up results, where the system is supposed to be stabilized and T_Restaurant agents didn't change anymore their services, are used as starting points for next experimentations and complex scenarios.

Using current ground data and official statistics to initialize some experimentations, we notice that the current state of the catering economy in the area of Evisa correspond with an equilibrium point. E_Bar agents don't change their service.

D. Free market vs. tourist trap

In this set of experimentations, decision makers (T_DM) decide to control the information delivered toward tourists through the T_Guide agents.

We start our experimentations from current field data and statistics, point of equilibrium, to foresee long term influence of decision makers' choices in the area of Evisa.

In scenario A, close to 2010's realty, information delivered to tourist is poor and not up to date. Simulations show that all the T_Restaurant tend to offer same services at the same prices. Because tourists are misinformed, they can't make a real choice and all T_Restaurant offer the same generic product liable to satisfy most of customer. Final state is an oligopoly with a unique offer toward travelers, a very poor competition between economic agents, and lastly the village of Evisa become a tourist trap: one single poor and expensive product offered.

In scenario B, information delivered to tourist is rich and near real time up to date. This scenario is based on decision makers' actions of promotion on Internet and the development of a sort of real time network informing travelers through Smartphone. Simulations show that all the T_Restaurant tend to offer different services and prices. Because tourists are well informed, they can make a choice and T_Restaurant offer products fitted to satisfy each clienteles. Final state is a free market with a rich offer toward travelers, a strong competition between economic agents. Lastly Evisa will become a healthy touristic destination: one product at the best price for each types of tourist.

These experimentations are helpful for decision makers to foresee future, explain their choice to economic players, and justify changes and investment. In small community similar to Evisa's, none change can occur against local economic players. Our results can convince Evisa's economic players to become stakeholder in the improvement of the information toward tourists.

E. Impact of restaurants' localization

From our experimentations we highlight that the importance of the localization for the T_Restaurant agents depends of both the structure of the touristic demand, the clientele degree of information and the restaurants' characteristic.

T_Restaurant fitted to a niche market are very sensitive to their location when the system of information is poor. Indeed, when they are misinformed, these customers find them with difficulty, and these agents will survive only if they are located in a place where the flow of tourists big enough to guarantee them a minimum of income.

Thanks to GIS data integration, we highlight by our experimentations the importance of others T_Restaurant agents in the near environment for niche market players when the information toward tourist is poor. Other restaurants attract more people and produce a domino effect: niche clientele is more disposed to consume when it surrounded by other people consuming.

F. Impact of new clientele

In this set of experimentations we explore the consequences induced by qualitative and quantitative changes in the flow of tourists.

In scenario A, qualitative changes, similar destinations where decision makers decide to improve the profile of the clientele: not more tourists but tourists with a better buying power. This strategy is used to face with the overcrowding and the collapse risk of the native communities under the tourism pressure (Pandey). Thus, since 1980 Seychelles (MTT) and Bhutan (NEC) limit the number of tourists.

In scenario B, quantitative changes, we explore effects indeed by additional arrivals. This case of study is similar to Corsica where decision makers decide to attract low cost airways that have just shyly landed for the first time in the 2009's summer. Due to political policy the number of travelers is expected to grow fast in the next years.

The explorations of scenarios A&B in the Evisa's system confirm our previous results about the importance of the localization and the information toward tourist in the speed of adaption of economic players.

V DISCUSSION

Even the multi-agent paradigm has been already used successfully many times to provide help to traveler and people looking to purchase travels, it's still difficult to use it to model the behavior of individual visiting a place due to the complexity of the human choice process.

Our results according to observations and theory are obtained by simplifying the factors implied in the tourist's decision making. Add more factors such as weather, temperature, view or mood makes the model more faithful while simulations become chaotic and results unexpoitable.

VI CONCLUSION AND PERSPECTIVES

The study and simulation of catering touristic market imply to take into account the spatiotemporal data and constraints. A generic multi-agent based behavioral modeling approach must be conducted jointly to a fine understanding of tourism economy. The use of the CORMGIS platform allows taking into account spatiotemporal data processing works to fit to each territory.

The tourism problematic is at the frontier of psychology, marketing, economy and identities of travelers and host communities. This system is emblematic of both the complex interactions between global political choices and individual consequences, and the tools' expectancies to find a way toward a controlled sustainable development. Following our methodology we build a model fitted to answer to the economic players' and the managers' expectancies. The first experimentations dedicated to the study of the Corsican village of Evisa according to macro economical theory and realty observations confirm the relevance and validity of our hybrid MAS GIS approach of decision support systems dedicated to tourism economy. Next we will study consequences of new opening restaurants and of enacting new rules and grants.

Furthermore we can use similar methodology to explore others fields with a strong spatial component and numerous stakeholders such as dynamic of population, marine ecosystems, epidemics and diseases and management of harbor.

REFERENCES

ArcGIS description http://www.esri.com/

Bousquet, F. and al. 1998. « Cormas: Common-Pool Resources and Multi-agent Systems », *Proceedings of the 11th International Conference on Industrial and Engineering Applications of Artificial In telligence and Expert Systems: Tasks and Methods in Applied Artificial Intelligence*.

Caccomo, J. and B. Solonandrasana. 2006. "*L'innovation dans l'industrie touristique: enjeux et stratégies*", L'Harmatan (ed), Paris.

CE. 2003. COUNCIL OF EUROPE - COMMITTEE OF MINISTERS Recommendation Rec(2003)1 of the Committee of Ministers to member states on the promotion of tourism to foster the cultural heritage as a factor for sustainable development .

CIRAD, French agricultural research centre working for international development, description http://www.cirad.fr/indexeng.htm

Fink , J. and al. 2002. "User Modeling for Personalized City Tours", Artificial Intelligence Review, Springer Netherlands, Volume 18, Number 1 / septembre 2002

Koch, A. 2000 "Linking multi agent systems and GIS, modeling and simulating spatial interactions", *Proceedings of 29th International Geographical Congress*, Seoul, South Korea

MTT. 2000."*Vision 21 : TOURISM DEVELOPMENT IN SEYCHELLES 2001-2010*", Ministry of Tourism & Transport, Republic of Seychelles

NEC. 1998. "*National Environmental Action Plan 1998 2001*", National Environment Commission (NEC) Government of Bhutan, pp 51-53

Pandey, R. and al. 1995. "*CASE STUDY ON THE EFFECTS OF TOURISM ON CULTURE AND THE ENVIRONMENT.NEPAL : Chitwan-Sauraha and Pokhara-Ghandruk*", Bangkok, UNESCO.

UNEP. 2005. "*Dossier on tourism and sustainable development in the Mediterranean*". UNEP. MAP Technical Reports series n° 159, Plan Bleu.

UNWTO. 2006. "*Tourism Market Trends*", United Nation World Tourism Organisation.

UNWTO. 2009. "*UNWTO Highlights 2009*", United Nation World Tourism Organisation.

Urbani, D. and M. Delhom. 2006a. "Water Management Using a New Hybrid Multi-Agents System – Geographic Information System Decision Support System Framework", *Proceedings of IEEE International Symposium on Environment Identities and Mediterranean area*, Corte, France.

Urbani, D. 2006b. "*Development of a new MAS GIS hybrid approach for Decision Support Systems; application to water management*", Ph. D. these, University of Corsica.

Urbani, D. and M. Delhom. 2008a. "Hybrid MAS GIS Coastal Systems Modeling Methodology"; *Proceedings of 21st International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2008*, Wroclaw, Poland. LNCS 5027 Springer 2008

Urbani, D. and M. Delhom. 2008b."Analyzing knowledge exchanges in hybrid MAS GIS decision support systems, toward a new DSS architecture", *Proceedings of the 2nd KES AMSTA'08 International conference on Agent and multi-agent systems*, Incheon, Korea

Urbani, D. and M. Delhom. 2009." "Hybrid MAS GIS Mediterranean Backcountry Tourism Economy Modeling" , *Proceedings of the 2nd International Conference on Simulation Tools*, Simutools 2009, Rome, Italy.

Westervelt, D. 2002. "Geographic Information Systems and Agent-Based Modeling", In Integrating Geographic Information Systems and Agent-Based Modeling Techniques, H. R. Gimblett (ed.), Oxford University Press, New York.

WTTC. 2010. "*Economic Impact Data and Forecast, world key facts 2010*", World Travel & Tourism Council

MULTI AGENT MODEL OF TRUST AFFECTION

Arnostka Netrvalova and Jiri Safarik
Department of Computer Science and Engineering
University of West Bohemia
Univerzitni 8, Plzen
Czech Republic
E-mail: netrvalo@kiv.zcu.cz

KEYWORDS
Trust, trust modeling, impersonal trust, trust affection.

ABSTRACT

The paper deals with construction of a platform for trust modeling extended by intentional affection of trust using a multi-agent system. Terms trust, phenomenal trust as a modification of impersonal trust, and trust representation are introduced, and model of trust affection is presented. Design of corresponding multi-agent system is described and applied to real data. These data deal with the public opinion poll of chosen ecological problems. Survey was acquired from websites articles of the Institute of Sociology of the Academy of Sciences of the Czech Republic.

INTRODUCTION AND RELATED WORKS

Many studies coming from psychological or social sciences describe the meaning and characteristics of trust (Luhmann 1979; Fukuyama 1995; Sztompka 1999; Gambetta 2000). Computational models for exploration of trust formation were created (Mui 2002; Lifen 2008). Wide-spreading of e-service, e-commerce (Zhang et al. 2008; Sathiyamoorthy et al. 2009), e-banking, etc., raise question of human machine trust. Further, trust plays an important role in peer-to-peer networks, ad hoc networks, grid computing, semantic web, and multi agent systems (Indiramma et al. 2008; Samek et al. 2009; Sathiyamoorthy et al. 2010), where humans and/or machines have to collaborate. The aim of our work is simulation of the trust evolution under intentional trust affection. This is common real situation, e.g. when bank affects the clients to trust it.

TRUST AND TRUST REPRESENTATION

The acceptation of the term trust is wide (Fetzer 1988). Based on Gambetta (Gambetta 2000), we interpret trust as a confidence in the ability or intention of a person to be of benefit to trust something or someone at sometime in future. Trust in our model is represented by a value from continuous interval $\langle 0, 1\rangle$. Value 0 represents complete distrust and value 1 means blind trust. Trust evolves not only within personal relations - personal trust, but person can trust to a phenomenon – phenomenal trust. In this case, trust is formed towards a phenomenon, e.g. to certain product from a set of products of some kind or to a political party.

Considering a set of m products, the distribution of person's trust can be described by trust values t^k , $0 \le t^k \le 1$, $k = 1, \ldots, m$, and their sum is one.

INTERVENTION MODEL

The general model of information intervention effect (Vavra F., University of West Bohemia, personal communication) will be applied. Let finite set of events X with the probability distribution mass function $P(x)$, $x \in X$ on the input represents the state before intervention, e.g. initial probability of specific product preferences from a set of products of some kind. Probability distribution $Q(x)$ on the output describes the state after intervention activity and the intervention is modelled by probability distribution $R(x)$. The simple method for joining initial probability and intervention probability is their mixture

$$Q(x) = (1-\lambda)P(x) + \lambda\, R(x) \qquad (1)$$

where $0 < \lambda \le 1$, represents intensity of the intervention.
Given probability mass functions $P(x)$, $R(x)$, $Q(x)$, the intensity λ can be found by the method of the least squares when all probability mass functions exist.

PHENOMENAL TRUST AFFECTION

Further, we will cope with an intentional trust intervention applying presented intervention technique to phenomenal trust. Consider a group of n subjects represented as the set $S = \{s_1, s_2, \ldots, s_n\}$, and a group of m exclusive products of some kind represented as a set $P = \{p_1, p_2, \ldots, p_m\}$ that constitutes the phenomenon. Trust of subject s_i, $i = 1, \ldots n$, to product p_k, $k = 1, \ldots, m$, is denoted as follows

$$t_i^k = t(s_i, p_k),\ t_i^k \in \langle 0,1\rangle,\ \text{and}\ \sum_{k=1}^{m} t_i^{\,k} = 1 \qquad (2)$$

The dominant product p_d is defined as product a subject $s \in S$ trusts mostly. This trust is called t_d, $t_d = \max t(s, p_k)$, $k = 1,\ldots, m$. The population S can be divided into the preferential classes according to the dominant product the individual subject trusts. Population trust to dominant product is denoted by T_d. The example of subject (t) and population (T) trust distributions to five products of a phenomenon is shown in the Figure 1. It illustrates a possible

situation, when the dominant product in whole population (p_3) differs from dominant product of specific individual (p_4).

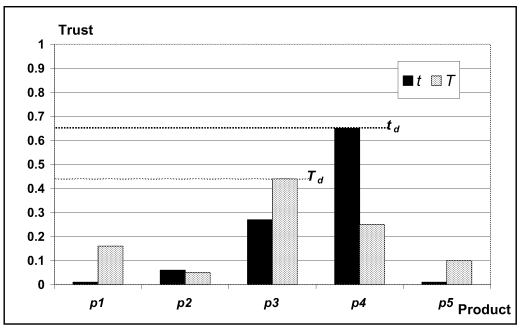


Figure 1: Population and Subject Trust Distributions Example

Now, consider affection of trust in favor of selected product in order to gain or even increase dominancy. This is modeled by mixture of intervention distribution I and current trust distribution to the products of individuals. Then, new trust probability distribution is given by values t'^{k}_{i}

$$t'^{k}_{i} = (1- \lambda^{k}_{i})\, t^{k}_{i} + \lambda^{k}_{i}\, I^{k}_{i}, \qquad (3)$$

where $0 \le t^{k}_{i} \le 1$, $0 \le I^{k}_{i} \le 1$, $\sum_{k=1}^{m} t^{k}_{i} = 1$, and $\sum_{k=1}^{m} I^{k}_{i} = 1$

The formula (3) means that some part of previous trust is transformed into intervened trust.

TRUST INTERVENTION MODEL FRAMEWORK

Agent model structure is hierarchical and covers four sets of subjects. The first set is called Consumers, the second Producers, next Analyzer (set of one or more agents), and the last is Dominator (set of one agent). The model hierarchy is shown in the Figure 2.

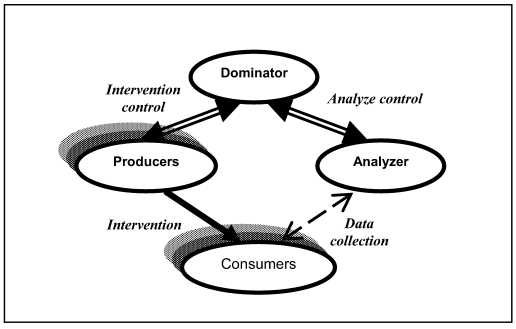


Figure 2: Model Hierarchy

Dominator is the highest element in the hierarchical structure, has the control function of the whole intervention process, sets the input parameters, and evaluates the impact of the intervention. Analyzer and Producer represent the next lower hierarchic level. Intervention is realized through chosen Producers on the whole set of the Consumers or its subset. Analyzer is advisory service agent, which requests and collects data on trust changes of the Consumers, analyzes the intervention process, and sends the results to Dominator. Producers authorized by Dominator are charged with the task to perform the intentional intervention on selected Consumers. Consumer is the lowest element in hierarchy that is able to change his phenomenal trust distribution to products depending on Producer's intervention, and sends the messages about trust changes to Analyzer. The phenomenal trust model is implemented (Hruska 2010) exploiting Java Agent DEvelopment Framework JADE (JADE 2010).

CASE STUDY

To illustrate trust evolution under affection, we took data obtained from the reports on the portal websites of the Institute of Sociology of the Academy of Sciences of the Czech Republic (IS 2009). The data deal with an opinion on growing genetically modified farming products (Global ecological problems by Czech public eyesight published in May 2009). The respondents answered the question: "Tell us, please, what is your view of growing the genetic modified farming products?", and 22% respondents considered growing the genetic modified farming products as big, 31% quite and 21% as small problem, 19% didn't know, and 7% saw no problem. For simplicity, data are reduced into two values. First three answers in "big problem" and the last two ones in "no problem". Then, dominant trust value of an individual needs to be higher than 0.5. The higher trust the stronger belief in dominantly trusted value. Choosing the mean of belief, we generate the population having dominant trust distribution approximately normal.

Intervention distribution adjustment

We start with the very neutral situation modeling the state before the discussion on growing genetically modified farming products started. The opinion in population is evenly distributed with rather low belief due to the lack of information. The May situation corresponds to intervention I_d between 0.7 and 0.8 in favor of big problem. We explored how the trust under these values will change depending on the intensity of intervention λ. Results of this study are shown in the Figure 3.

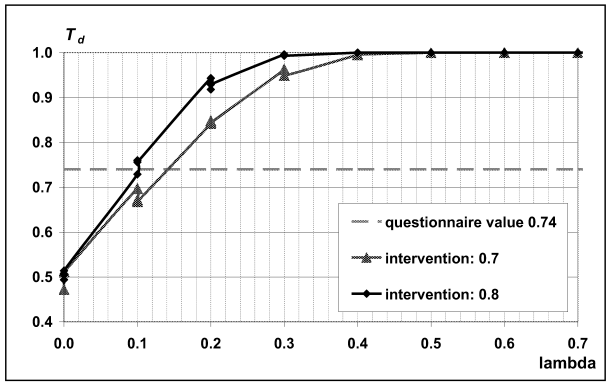


Figure 3: Study of Intervention Distribution Adjustment

In the graph, a curve connects the computed discrete values denoted by different marks. A vertical distance of the same mark represents the population trust dispersion value

computed in simulation runs. The value acquired from the questionnaire is depicted by dashed line. The values gained by the intersection points with the questionnaire value range from 0.1 to 0.14. According to expectation, the higher intervention value I_d the lower intensity λ is needed. The reason for relatively low intensity value can be overall situation in the society, when concerns grew slightly.

The new questionnaire results were published at the beginning of July 2010. The rate between "big problem" and "no problem" products changed only a little, from 74:26 to 73:27, which is in concordance with small media attention in the last year. Using formula (3) and according to small decrease of trust to dominant product, the required intervention distribution I_{dComp} was computed with trust value change from 0.74 to 0.73 using value of intensity $\lambda = 0.14$ (see Figure 3, I_d = 0.7). The computed value I_{dComp} = 0.67 corresponds to our simulation result for $I_d = 0.7$.

Expected value influence

Next, we studied how the intensity λ needed to reach today's state depends on the mean value μ of belief to the dominant value. Following values of mean values were chosen: $\mu = 0.6$, 0.7, 0.8, and 0.9 instead of neutral 0.5. The results of this study are shown in the Figure 4.

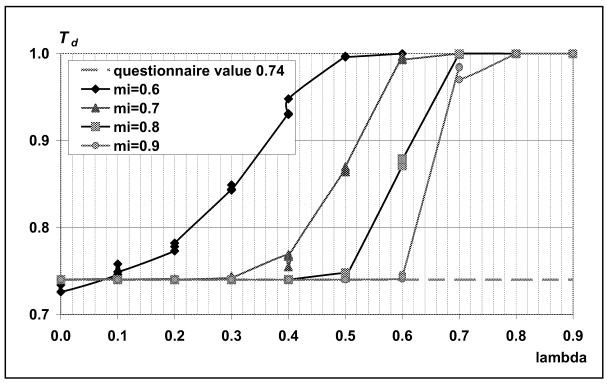


Figure 4: Study of Expected Value Influence

The value acquired from the questionnaire is depicted by dashed curve. The parameter λ increases with growing mean μ to accomplish the same trust. This result is in good concordance with human behavior, when we expect more effort to change somebody's opinion, in which he believes strongly.

CONCLUSION

We developed the phenomenal trust model integrating intentional affection of trust evolution. The model itself is deployed in the agent based trust management model. We demonstrated its application to the real data. The model confirmed expected sociological behavior, moreover some its aspects can be quantified. Upcoming model modification will allow covering the effect on benefit of more products in several time series.

ACKNOWLEDGEMENT

The work was granted by the Ministry of Education, Youth and Sport of the Czech Republic - University spec. research – 1311.

REFERENCES

Fetzer S., 1988. "The World Book Dictionary." World Book Inc., The World Book Encyclopaedia, Chicago, USA.

Fukuyama, F. 1995. "*Trust: The social virtues and the creation of prosperity*". New York, The Free Press.

Gambetta D., 2000. "*Can We Trust Trust?*" In Gambetta, Diego (ed.) Trust: Making and Breaking Cooperative Relations, electronic edition. Department of Sociology, University of Oxford, chapter 13, 213-237.

Hruska V., 2010. "Simulation trust by multi-agent technology". Diploma Thesis. Department of Computer Science, University of West Bohemia, Plzen, Czech Republic.

Indiramma M. and Anandakumar K., 2008. TCM: "A Trust Computation Model for collaborative decision making in Multi-agent System". *International Journal of Computer Science and Network Security (IJCSNS)*, vol.8 No.11, November 2008

Institute of Sociology, Academy of Sciences of the Czech Republic. 2009. Available at: http://www.soc.cas.cz/articles, [Cit. 2009-06-26, 2010-07-07]

JADE. Available at: http://jade.tilab.com, [Cit. 2010-06-19].

Lifen L., 2008. Trust Derivation and Recommendation Management in a Trust Model. In: Proc. *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 219-222, Harbin, China.

Luhmann N., 1979. *Trust and Power*. New York, John Wiley.

Mui L., 2002. "Computational Model of Trust and Reputation: Agents, Evolutionary Games, and Social Networks". Ph.D. Thesis, Electrical Engineering and Computer Science, Massachusetts Institute of Technology, USA.

Samek J., and Zboril F., 2009. "Agent Reasoning Based On Trust and Reputation". In: *Proc. MATHMOD' 09, ARGESIM* (Vienna, Austria), pp. 538-544.

Sathiyamoorthy E., Iyenger S., and Ramachandran V., 2009. "Agent Based Trust Management Model Based on Weight Value Model for Online Auctions". *International Journal of Network Security & Its Applications (IJNSA)*, vol. 1, No. 3.

Sathiyamoorthy E., Iyenger S., and Ramachandran V., 2010. "Agent Based Trust Management Framework in Distributed E-Business Environment", *International Journal of Computer Sciences & Information Technology (IJCSIT)*, vol. 2, No. 1.

Sztompka P., 1999. Trust. *A Sociological Theory*. Cambridge, Cambridge University Press.

Zhang Z., Zhou M., and Wang P., 2008. "An Improved Trust in Agent-mediated e-commerce." *International Journal of Intelligent Systems Technologies and Applications*, vol. 4, 271-284.

AUTHOR BIOGRAPHY

ARNOSTKA NETRVALOVA was born in Plzen, Czech Republic. She is senior lecturer in Department of Computer Science and Engineering at Faculty of Applied Sciences of University of West Bohemia. She received Ph.D. from this university in 2010. Her present research is focused on trust modeling and simulation.

MANAGEMENT SIMULATION

ON THE COMPLETION TIME OF A PROJECT WITH RANDOM ACTIVITY DURATIONS BASED ON A MODEL OF STOCHASTIC MARKED GRAPHS

Gerrit K. Janssens
Transportation Research Institute (IMOB)
Hasselt University - campus Diepenbeek
Wetenschapspark 5, 3590 Diepenbeek, Belgium
e-mail: gerrit.janssens@uhasselt.be

Kongkiti Phusavat and Pornthep Anussornnitisarn
IGP in Industrial Engineering, Kasetsart University
50 Phahon Yothin Road, Chatuchak, Bangkok 10900, Thailand
e-mail: {fengkpp,fengpta}@ku.ac.th

KEYWORDS

Project scheduling, random durations, Petri nets, stochastic marked graphs

ABSTRACT

Activities of a project often have random durations. This makes the estimation of the total project more difficult. Simulation, approximation and bouding techniques help the project manager in formulating these estimates. This paper focuses on formulating an upper bound on the project length based on the fact that an activity network can be represented as a stochastic marked graph. The bound, however, includes a part which requires that the distributions of the activity durations are fully specified. As many times, only incomplete information is available on these durations like the interval, the mean, the mode or the standard deviation, this research investigates how the bound should be computed including only incomplete information.

INTRODUCTION AND LITERATURE REVIEW

Scientifically it has become a challenge to develop a model guaranteeing a successful project. But the reality is different: project activities use all available time; activities are finished for 90% during 90% of their time; a carelessly planned project takes three times the time as planned, a carefully planned project only twice (Shtub et al., Chapter 1, 1994). These thoughts might sound humorous, but the fact is that, however, dedicated project management suffers from uncertainties. The effect is even stronger because a project manager cannot have everything under control. Finally, a project which should not have been started cannot be ended successfully by whatever excellent project manager .

Project network analyses concentrate mostly on the total project time (or makespan in a scheduling context). Management is interested in two types of questions: "What is the expected total project time?", and "Which activities are critical in obtaining this total project duration?" Research into finding out which activities 'most probably' lie on the critical path is done by Dodin and Elmaghraby (1985). Projects are confronted in an extended way with the following objectives: be completed by a certain date, for a certain amount of money, within some level of performance (Tuman, 1986). A project can be planned according to a minimum cost criterion, but it can take more time and maybe not reach the required quality or conformance to the specifications. Avots (1984) suggests that time planning is of greater importance in the earlier stages of the project; during the project cost becomes of more importance and, after the project, only the technical quality matters.

Risk in project management exists in all of the three building blocks: time, cost and quality. The project manager has to cope with risk in subsequent steps: first risk identification, then risk management and reduction, and finally risk evaluation (Ho, 1992). All steps make part of a decision-making process. The decision-maker has to judge whether a part of the inherent risk can be avoided, reduced or accepted.

The most widely studied type of risk is *time risk*. Time risk exists due to uncertainties in the duration of some activities. Most studies assume that all activities of the project are known, that the precedence relations are known, but their durations are not fully known. Less attention has been paid to *cost risk*. The reason is that cost risk is not at all specific for project management. The problem is easier than the time risk due to the additivity of the individual components. In the literature hardly any articles exist on the analysis of *quality risk*. However, it has been found that many projects fail because the technical contents of the project have not been controlled sufficiently or not enough at an early stage (Morris, 1988). Time risk is certainly of a more complex nature than cost risk, but still has the advantage that techniques exist to approximate or to simulate the uncertainties. In quality risk measures have to be compared which are not additive because they are expressed in different measuring units.

Duration of activities in a project network is mostly of random nature. Due to the fact that a project is unique, no historical data exist to provide information regarding the probability distribution of an activity duration. In the analyses it is common to estimate a number of moments of the duration of the individual activities. Hardly anyone tries to obtain a full distribution. An *a priori*-distribution is chosen: most authors choose for a Beta-distribution. However, in the context of simulation other distributions have been proposed such as the triangular distribution, because random numbers from these distributions can be drawn in a more efficient way. It is of greater importance whether the choice of the distribution is of any importance. However, MacCrimmon and Ryavec (1964) have shown that the choice of the distribution hardly influences the total project time, except for some very extreme types of distributions.

Bounding techniques are also useful in terms of approximation. Most textbooks today still use the approximation by the authors of the original formulation Malcolm et al. (1959). They estimated the expected project duration as the length of the longest path through the deterministic network obtained by replacing each random arc length by its expected value.

A next review (Adlakha and Kulkarni, 1989), covering the literature from 1966-1987, deals with stochastic PERT networks: they stress the risk aspect with subjects as estimates, errors, bias and Monte-Carlo simulation approximations.

The PERT approach has proposed to collect information from experts in the form of three estimates for the duration: an optimistic, most likely and pessimistic estimate. From these estimates formulae for the expected value and the variance are proposed. The formulae are based on the assumption that a Beta-distribution might be underlying the duration, with the optimistic and pessimistic as the upper and lower bound of the range on which the distribution is defined. The most likely estimate corresponds to the modal value of the distribution (in case it is in fact unimodal).

Some criticism has been formulated on this approach as some combinations of the three estimates with the proposed formulae may lead to bimodal types of the Beta-distribution (Pagnoni, Chapter 4, 1990). The triangular distribution might be a valid alternative with the same interpretation of the three estimates. Any distribution on a finite range is valid as long as the interpretation of the three estimates leads to a unimodal distribution. However the largest type of criticism relates to the determination of the project length distribution in terms of its expected values, its variance or its tail probabilities. To simplify the computation of the expected value of the total project time and its variance, some additional assumptions are made within the PERT-approach. They are: (1) one single path dominates all others. This means that the probability is very low that another path becomes the critical path; and (2) the activity durations are independent random variables.

In the PERT approach the path with the longest (approximate) expected value is chosen as the critical path. It is assumed that the activity durations are independent of each other meaning that the expected total project time and its variance can be obtained by summing the expected values resp. variances of the activities on the critical path. On basis of the central limit theorem it is assumed that the total project length follows a Normal distribution. These assumptions are used to make statements on the completion of the total project within a required deadline. The PERT approach leads to an optimistic value (a lower bound) on the expected value of the project length. Also the normal character of the project length can be questioned. Even if the lengths of the individual paths follow a Normal distribution, the project length is Normally distributed only if the dominance assumption is valid (Elmaghraby, 1977). Therefore a lot of research has been spent on finding approximations which are more realistic or to bounds on the project length's expected value.

In this paper we determine bounds on the expected value of the project length. The lower bound is quite trivial. In the next section it is shown how a project network can be modeled as a stochastic marked graph (SMG). Through an optimisation problem on the SMG, an upper bound on the expected project length is obtained without assuming independence of paths or neglecting some paths. It however does not provide a measure of variability in order to evaluate the risk on the project duration.

A PROJECT NETWORK AS A STOCHASTIC MARKED GRAPH

The activity-on-arc and activity-on-node network representations are two classical representations of a project network.

Petri nets are an established model to represent and analyze concurrent systems. A *Petri net* is a collection of directed arcs connecting places and transitions. These arcs have a default capacity of one unless stated otherwise. Places can contain tokens, and the assignment of tokens to places is called the state or marking of the net. Arcs can only connect places to transitions and vice versa. A transition is said to be enabled if the number of tokens in its input places is at least equal to the arc weight going from the input places to the transition. Once enabled, a transition can fire. When fired, the tokens in the input places are moved to output places, according to the arc weights and place capacities.

A *marked graph* is a Petri net in which each place has at most one input transition and one output transition. Marked graphs constitute a good formalism to model manufacturing systems containing parallel tasks and synchronization or to order activities like in PERT. They are more general than PERT graphs in the sense that places can contain several tokens. Marked graphs have been studied extensively either in a deterministic or in a stochastic context. One of the main problems of timed and stochastic Petri net models for large systems is the explosion of computational complexity

algorithms to analyze performance measures (such as the cycle time) of marked graphs. Campos et al. (1992) determine upper and lower bounds on the steady-state performance of marked graphs to evaluate performance in an efficient way. In this paper we develop a tight upper bound for the cycle time of a stochastic marked graph.

In case the SMG is cyclic, an upper bound on the cycle time can be computed through an optimisation problem in which some function of the stochastic transition times needs to be determined. By connecting the start transition of the network with the finish transition a cycle is created. For the required function an upper bound can be computed depending on available information for the stochastic transition times, for example the range, mean, variance and/or mode (Janssens et al., 2009). In this research, the available information consists of the range and mode, and – in case of a specific type of distribution for the transition times is assumed – also a first and second moment of the distribution.

A project can be modelled as a safe marked graph, which means that each place can hold at maximum one token. In a safe marked graph every transition must have some input place, which means that the graph G is covered with directed circuits. As the number of tokens on a directed circuit is invariant under firing, all transitions on token-free circuits are dead. Therefore, let us assume that the initial marking M_0 places at least one token on each directed circuit in G. In this case, the marked graph is live and, since it is safe, M_0 places exactly one token on each directed circuit in G.

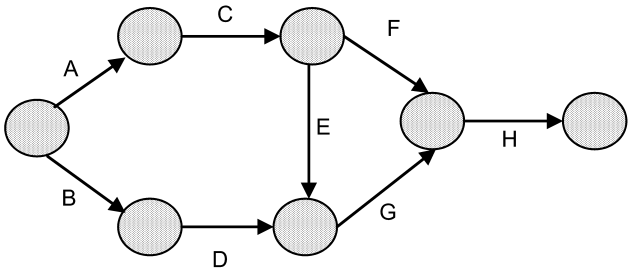


Figure 1: An activity-on-arc network

During the explanation of the theoretical development, the results will be illustrated by means of a project network example, taken from Heizer and Render (2001, Chapter 16, p. 674). An activity-on-arc representation of the project is shown in Figure 1.

Its representation as an SMG is shown in Figure 2. To each activity is associated a transition. One transition (t_Z) is added to produce a cyclic SMG. The concept of immediate predecessors is represented by forks and joins in the SMG. To each transition a stochastic firing time distribution is associated (the firing time of $t_Z = 0$).

Typically in project scheduling, the stochastic analysis is based on three estimates: optimistic, most probable, and pessimistic. The time estimates of the Heizer and Render example are given in Table 1.

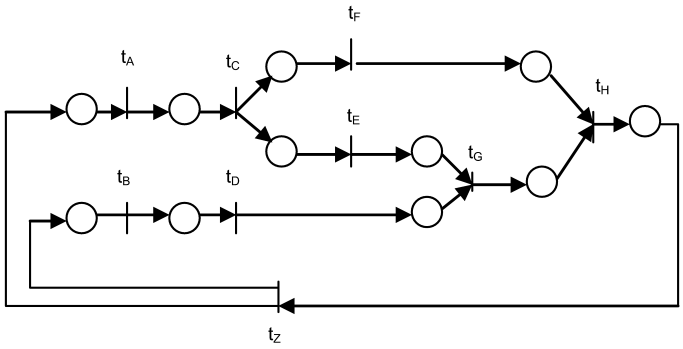


Figure 2: Stochastic marked graph related to Figure 1

Activity	*Optimistic*	*Most probable*	*Pessimistic*
A	1	2	3
B	2	3	4
C	1	2	3
D	2	4	6
E	1	4	7
F	1	2	9
G	3	4	11
H	1	2	3

Table 1: Time estimates

Bounding techniques obtain lower and upper bounds for the expected project length. The lower bound is of less interest but is required to discuss as it appears in the formulation of the upper bound. Prior to determining the lower bound for a stochastic marked graph, the bound is determined for its deterministic counterpart. Following Magott (1984) the minimal cycle time can be found as the solution of a linear program (LP).

When switching from deterministic marked graphs towards SMG, a similar linear optimization program has been formulated by Campos et al. (1992, p. 390). A lower bound for the mean cycle time for live strongly connected marked graphs can be obtained by solving such a linear program. As deterministic timed graphs are a special case of SMG with the mean transition firing time equal to the deterministic firing time, both types of linear programs give the same results in case of deterministic marked graphs.

Campos et al. (1992) prove that for strongly connected marked graphs with arbitrary values of mean and variance for transition firing times, the lower bound for the mean cycle time obtained in their LP cannot be improved. If both mean and variance of the firing time of each transition are known, the lower bound as obtained by the LP cannot be reached (unless all variances are equal to zero).

The asymmetric three-parameter triangular distribution appears in the project scheduling literature already for a long time. The three parameters are in a one-to-one correspondence with the optimistic, most probable and

pessimistic estimates. This leads to an intuitive appeal to this distribution, while in reality the shape of the distribution is unknown. The expected value and the variance of the activity durations, assuming a triangular distribution, are given in Table 2.

Activity	*Expected value*	*Variance*
A	2.0	0.167
B	3.0	0.167
C	2.0	0.167
D	4.0	0.667
E	4.0	1.500
F	4.0	3.167
G	6.0	3.167
H	2.0	0.167

Table 2: Expected value and variance using the triangular distribution

The Critical Path Method, making use of the expected values from table 2 as deterministic values, a project length CT = 16 is obtained. The same value appears as a solution of Magott's linear program for the marked graph:

Min CT
subject to
$S(t_A) - S(t_Z) \geq 2$
$S(t_B) - S(t_Z) \geq 3$
$S(t_C) - S(t_A) \geq 2$
$S(t_D) - S(t_B) \geq 4$
$S(t_F) - S(t_C) \geq 4$
$S(t_E) - S(t_C) \geq 4$
$S(t_G) - S(t_E) \geq 6$
$S(t_G) - S(t_D) \geq 6$
$S(t_H) - S(t_F) \geq 2$
$S(t_H) - S(t_G) \geq 2$
$S(t_Z) - S(t_H) + CT \geq 0$

AN UPPER BOUND ON THE PROJECT NETWORK LENGTH

Let $\pi(M_0)$ be the cycle time of the SMG and $\pi^D(M_0)$ be the cycle time of its deterministic equivalent, discussed in the previous section. Sauer and Xie (1993) prove that the following bound holds:

$$\pi(M_0) \leq \pi^D(M_0) + \inf_{z \in E} \left\{ \sum_{t \in T} E\left[(X_t - z_t)^+ \right] \right\} \qquad (1)$$

where the infimum needs to be found in the set E defined as:

$$E = \left\{ z \,\middle|\, z_t \geq m_t, \forall t \in T \text{ and } \sum_{t \in \gamma} z_t \leq \pi^D(M_0).M_0(\gamma), \forall \gamma \in \Gamma \right\},$$
$$z = [z_1, z_2, \ldots, z_m]$$

The second term in the right hand side of the inequality is an optimisation problem where a vector z has to be determined within a set of linear constraints E. The objective function however consists of a sum of nonlinear functions in the elements of the vector z. In most cases the partial expectation function $E[(X_t-z_t)^+]$ cannot be expressed analytically in z_t, so the objective function cannot be formulated. In other cases, like the case when the X_t are distributed according to a triangular distribution, the function can be expressed analytically but is difficult to solve because of the sum of nonlinear terms (Janssens et al., 2009). In cases, in which the distribution is not completely specified, also no objective function can be formulated.

A practical solution to this problem might be found in looking for distributions which have an upper bound on the the partial expectation function $E[(X_t-z_t)^+]$, given a number of constraints like the knowledge on the interval on which the variable is defined, the mode, the expected value of higher moments, or any combination of these characteristics. It is of interests to investigate whether these distributions lead to a feasible and/or easy way to formulate the objective function. Such an upper bound, of course, is only of practical value if the bound is tight.

There exists a similarity between the function of our interest and a function, called the stop-loss premium in insurance mathematics. In insurance mathematics, an insurance company using the option of re-insurance is confronted with a stop-loss premium. A stop-loss premium limits the risk X of an insurance company to a certain amount t. If the claim size is higher than t the re-insurance company takes over the risk X-t. The stop-loss premium is based on the expected value of X-t, which in case of a known claim size distribution may be defined as:

$$\int_0^\infty (x-t)^+ \, dF(x) \qquad (2)$$

where $F(x)$ represents the claim size distribution (Goovaerts, De Vylder, and Haezendonck 1984). A single term of our objective function is the same as this integral.

As the optimistic and pessimistic estimates are finite numbers, the distribution of the duration of an activity t can be represented as defined on an interval $[a_t, b_t]$ with $0 \leq a_t \leq b_t < +\infty$. In case only this knowledge is used, the worst distribution would put all its mass in b_t to obtain an upper bound:

$$E\left[(X_t - z_t)^+ \right] = b_t - z_t . \qquad (3)$$

This makes the second term in inequality (1) a linear objective, but uses minimal information on the activity duration.

The additional use of the most probable estimate (c_t) disturbs the linear character of the objective function. The upper bound is given by:

$$E\left[(X_t - z_t)^+ \right] = \frac{1}{2} \frac{(b_t - z_t)^2}{b - c_t} \; if \; c_t \leq z_t$$
$$E\left[(X_t - z_t)^+ \right] = \frac{1}{2}(b_t + c_t - 2z_t) \; if \; c_t \geq z_t \qquad (4)$$

The function contains a quadratic component in the decision variables z_t. Assuming unimodality, without specific knowledge of the mode c_t, will lead to the case in which the

upper bound puts all mass in b_t which becomes also the mode and reduces the bound to equation (3).

In case one has symmetric distributions in mind, the extreme case assigns as much as possible mass to b_t while satisfying the constraint of symmetry (the same mass to a_t). This assigns 50% mass to b_t and leads to the upper bound:

$$E\left[(X_t - z_t)^+\right] = \frac{1}{2}(b_t - z_t). \quad (5)$$

This leads again to a linear term and makes the optimization problem a linear programming problem.

The knowledge of the expected value m_t is also of interest. In that case the upper bound is found as:

$$E\left[(X_t - z_t)^+\right] = (b_t - z_t)\frac{m_t - a_t}{b_t - a_t}. \quad (6)$$

Again this leads to a linear term but, using the three estimates, the problem is that the expected values is not known. However, one might assume that the expected values corresponds to the one when assuming a triangular distribution (leaving open all distributions with variances different from the triangular). If one feels too restricted, the linear program can be run for several feasible values of m_t as the knowledge of the mode limits its range as:

$$\frac{1}{2}(a_t + c_t) \le m_t \le \frac{1}{2}(b_t + c_t). \quad (7)$$

If the project contains many activities with random durations, the latter option is not efficient.

In case a distribution on a finite interval with known expected value and variance are considered, Janssens et al. (2009) have also shown how to find the upper bound by means of a linear program. The objective function can be formulated as:

$$\inf_{z \in E} \sum_t (b_t - z_t)\frac{\sigma_t^2}{\sigma_t^2 + (b_t - m_t)^2} \quad (8)$$

where E is defined as in equation (1). It leads to the following inequality for the mean cycle time:

$$\pi(M_0) \le \pi^D(M_0) + \inf_{z \in E} \sum_t (b_t - z_t)\frac{\sigma_t^2}{\sigma_t^2 + (b_t - m_t)^2} \quad (9)$$

It can be seen that the second term in the r.h.s. is a linear objective function and that the constraints set E also contains only linear constraints. This second term can be written as a constant plus a sum of terms each including one decision variable z_t. The linear program is illustrated by means of the example of Figure 2:

Min 5.962806 – 0.142857 z_A - 0.142857 z_B - 0.142857 z_C - 0.142857 z_D - 0.142857 z_E – 0.112426 z_F – 0.112426 z_G – 0.142857 z_H

subject to

$z_A \ge 2$
$z_B \ge 3$
$z_C \ge 2$
$z_D \ge 4$
$z_E \ge 4$
$z_F \ge 4$
$z_G \ge 6$
$z_H \ge 2$
$z_A \le 3$
$z_B \le 4$
$z_C \le 3$
$z_D \le 6$
$z_E \le 7$
$z_F \le 9$
$z_G \le 11$
$z_H \le 3$
$z_z + z_A + z_C + z_F + z_H \le 16$
$z_z + z_A + z_C + z_E + z_G + z_H \le 16$
$z_z + z_B + z_D + z_G + z_H \le 16$

The model leads to the following solution: $z_A = 2$, $z_B = 3$, $z_C = 2$, $z_D = 5$, $z_E = 4$, $z_F = 9$, $z_G = 6$, $z_H = 2$, $z_Z = 0$. The objective value is equal to 1.70499, which makes the upper bound equal to 17.70499.

CONCLUSIONS

An optimization model has been formulated to compute a tight upper bound for the expected project time in case of activities with random durations. The model is useful both in the case when the distributions of the ectivity durations are fully specified or partially specified. In the former case not all information is used, but the model offers a tractable method of finding a good upper bound. In the latter case the model offers the best solution to obtain this bound. It has been shown that in some cases the model leads to a linear program. In other cases it leads to a mathematical programming model with a non-linear objective function but with a set of linear constraints.

REFERENCES

Adlakha, V.G. en V.G. Kulkarni, 1989, A classified bibliography on research on stochastic PERT networks: 1966-1987, *INFOR*, vol. 27, pp. 272-296.

Avots, I., 1984, Information systems for matrix organisations, in: D.I. Cleland (ed.), *Matrix Management Systems Handbook,* Van Nostrand Reinhold, New York.

Campos, J., G. Chiola, J.M. Colom and M. Silva, 1992, Propoerties and performance bounds for timed marked graphs, *IEEE Transactions on Circuits and Systems,* CS-39(5), pp. 386-401.

Dodin, B. and S.E. Elmaghraby, 1985, Approximating the criticality indices of the activities in PERT networks, *Management Science*, vol. 31, pp. 207-223.

Elmaghraby, S.E.,1977, *Activity Networks: Project Planning and Control by Network Models*, Wiley, New York.

Goovaerts, M.J., De Vylder, F. and Haezendonck, J., 1984, *Insurance Premiums.* Amsterdam: North-Holland.

Ho, S.S.M., 1992, The use of risk analysis techniques in capital investment appraisal, in J. Ansell en F. Wharton (eds.), *Risk: Analysis, Assessment and Management*, Wiley & Sons, Chichester, blz. 71-94.

Janssens, G.K., K. Sörensen and W. Dullaert, 2009, An upper bound of the cycle time of a stochastic marked graph using incomplete information on the transition firing times, *Mathematical and Computer Modelling,* vol. 40, pp. 563-572.

MacCrimmon, K.R. and C.A. Ryavec, 1964, An analytical study of the PERT assumptions, *Operations Research*, vol. 12, pp. 16-27.

Magott, J., 1984, Performance evaluation of concurrent systems using Petri nets, *Information Processing Letters,* vol. 18, p. 7-13.

Malcolm, D.G., J.H. Roseboom, C.E. Clark and W. Fazar, 1959, Applications of a technique for R & D program evaluation, *Operations Research*, vol. 7, pp. 646-669.

Morris, P.W.G., 1988, Managing project interfaces - Key points for project success, in D.I. Cleland en W.R. King (eds.), *Project Management Handbook*, Van Nostrand Reinhold, New York, pp. 16-55.

Pagnoni A., 1990, *Project Engineering: Computer-Oriented Planning and Operational Decision Making,* Springer Verlag, Berlin, 239 pp.

Sauer N. and X. Xie, 1993, Marking optimization of stochastic timed event graphs, in Application and Theory of Petri Nets 1993, *Lecture Notes in Computer Science* 691, pp. 357-376.

Shtub A., J.F. Bard and S. Globerson, 1994, *Project Management: Engineering, Technolohy and Implementation,* Prentice-Hall, Englewood Cliffs.

Tuman, J., 1988, Development and implementation of project management systems, in: D.I. Cleland en W.R. King (eds.), *Project Management Handbook*, (2nd ed.), Van Nostrand Reinhold, New York, pp. 652-691.

BIOGRAPHY

GERRIT K. JANSSENS received degrees of M.Sc. in Engineering with Economy from the University of Antwerp (RUCA), Belgium, M.Sc. in Computer Science from the University of Ghent (RUG), Belgium, and Ph.D. from the Free University of Brussels (VUB), Belgium. After some years of work at General Motors Continental, Antwerp, he joined the University of Antwerp. Currently he is Professor of Operations Management and Logistics at Hasselt University (UHasselt) within the Faculty of Business Administration. He has been president of the Belgian Operations Research Society (ORBEL) in 2006-2007. During the last twenty years he has been several times visiting faculty in universities in South-East Asia and Africa. His main research interests include the development and application of operations research models in production and distribution logistics.

KONGKITI PHUSAVAT is Associate Professor and Director of the International Graduate Program in Industrial Engineering, Department of Industrial Engineering at Kasetsart University, Bangkok, Thailand. He received his doctoral and master degrees in Industrial and Systems Engineering from Virginia Tech, USA in 1995. His undergraduate degree is also in Industrial Engineering from Texas Tech, USA. He has worked with several organizations in the areas of management system analysis, productivity management, performance measurement, acquisition logistics, and supply-chain management. He is a member of several editorial boards such as Industrial Management and Data Systems, International Journal of Management and Enterprise Development, International Journal of Services and Standards, International of Innovation and Learning, and International Journal of Sustainable Economy.

PORNTHEP ANUSSORNNITISARN is a Lecturer and the Deputy Director of the International Graduate Program in Industrial Engineering, Department of Industrial Engineering at Kasetsart University, Bangkok, Thailand. He received his Doctoral degree in Industrial Engineering from Purdue University in 2003. His research includes logistics, supply chain management, and applied ICT to improve organizational learning an development.

An Overview of Negotiation Models for Activity-Travel Applications

Huiye Ma
Nicole Ronald
Theo Arentze
Harry Timmermans
Urban Planning Group, Eindhoven University of Technology
5612 WB, Eindhoven, The Netherlands
E-mail: h.ma@tue.nl

KEYWORDS
Decision-making, Modelling, Multi-agent simulation, Negotiation, Joint activity travel schedule.

ABSTRACT

In this paper, we discuss the development of the research on interactions from within household members to social network for activity-travel applications in the literature. Along with the multi-agent simulation approach, there has been more work to study individual decision making and interactions among agents in the dynamic environment. Therefore negotiation becomes the key to solve any conflict among agents. Finally an intentative agent mediated negotiation model is given.

INTRODUCTION

The activity-travel patterns of individuals often include interaction with other household members, which we observe in travel surveys as joint activity participation and shared travels (Scott and Kanaraglou 2002). For example, Vovsha et al. (2003) reported that, one-third to one-half of observed weekday tours involved some form of intra-household joint travel in late 1990s regional household travel surveys for New York and Columbus, Ohio. Therefore failure to take into account such linkages among household members will lead to model system mis-specification.

In view of a huge amount of papers focusing on the decision making of household level, interactions among household members have been slightly investigated in terms of their activity-travel behavior. Among those that have, the importance of such interactions is clearly demonstrated (Gliebe and Koppelman 2001).

Despite the paucity of empirical research to date, there is convincing evidence which suggests that interactions among household members can no longer be ignored in the development of behaviorally sound, activity-based forecasting models (Gliebe and Koppelman 2001; Scott and Kanaraglou 2002). Moreover, such household-level models (i.e. models that capture interactions among household members) are superior to those developed for individuals in that they offer potentially more accurate and reliable forecasts.

Besides household activities, social activities account for a large amount of travel, yet due to their irregularity and the number of options regarding location, participants, and timing, they are difficult to model and predict. Hence interactions are highly required to be included in the process of achieving social activities (Gilbert 2008; Ronald et al. 2009; Ronald et al. 2010a; Ronald et al. 2010b).

Obviously, as highlighted by the discussion above, there remains a need to incorporate interactions among members in the development of activity-travel applications with behaviorally rational participants and considerable dynamics.

Multi-agent simulation is therefore becoming increasingly important in the activity travel simulation, travel analysis, and travel forecasting, in particular due to its possibilities to model explicitly the individuals' decision making processes and on the other hand one's interaction with others and the dynamic environment. For examples, travels and activities can be seen as a result of individual decisions (Balmer 2007; Ronald et al. 2010b). Moreover, agents' interactions to unforeseen events in the environement are taken into account by enabling agents to reconsider an existing schedule and to adapt their expectations about traffic conditions for subsequent days (Arentze et al. 2010).

Perhaps the most fundamental and powerful mechanism for managing inter-agent dependencies at run-time is negotiation – the process by which a group of agents come to a mutually acceptable agreement on some issues (Jennings et al. 2001). Negotiation handles attempts to cooperate and coordinate and is required both when the agents are self interested and when they are cooperative. The means of achieving a stable state are to make self proposals, offer concessions, and hopefully come to a mutually acceptable agreement. In short, to negotiate.

Given its ubiquity and importance in many different contexts, negotiation theory incorporates a broad range of phenomena and makes use of many different approaches from AI, Social Psychology and Game Theory, etc. Automated negotiation research can be considered to deal with three broad topics (Jennings et al. 2001):

- Negotiation Protocols: the set of rules that govern the interaction.
- Negotiation Objects: the range of issues over which agreement must be reached.
- Agents' Decision Making Models: the decision making tools that the participants employ to act in line with the negotiation protocol in order to achieve their objectives.

It should be clear that there is no universally best approach or technique for automated negotiation in various applications. The aim of this paper is to search negotiation opportunities for autonomous agents in activity travel applications, and to point out the future automated negotiation research by means of an illustration.

The remainder of this paper is organized as follows. In Section 2, we give a brief overview of related works focusing on the interactions within a household level in transportation literature. In Section 3, we go further to discuss agent-mediated interactions. In Section 4, agent mediated negotiation models are surveyed shortly. In Section 5, an illustration of the agent mediated negotiation model is given. Section 6 concludes the paper.

INTERACTION WITHIN A HOUSEHOLD

In activity-travel modeling, joint activities have mostly been studied within households (Kostyniuk and Kitamura 1982; Golob and McNally 1997; Gliebe and Koppelman 2001). Some high-level models consider the number of joint outside-home activities (Wen and Koppelman 2001; Scott and Kanaraglou 2002), the time spent in joint activities (Gliebe and Koppelman 2002), and the development of a household task allocation and time use model (Zhang et al. 2005). Activities with non-household members are not covered, nor are the activities explicitly modeled, that is, scheduled for a particular location or time.

For example, the findings of Kostyniuk and Kitamura (1982) indicate that participation in joint out-of-home activities during the evening is quite common for some households. Specifically, young couples without children and dual-earner couples in which both husband and wife work on a given day are oriented toward such activities.

Golob and McNally (1997) employ a structural equation's model to capture interactions between male and female household heads in terms of their participation in work, maintenance and discretionary activities. They find that male work activity governs interactions. Their study shows that an increase in such activity leads to an increase in female maintenance activity and travel and to a decrease in female discretionary activity and travel. Furthermore, the number of young children in the household is related to the substitution of work and maintenance activities between male and female household heads.

Gliebe and Koppelman (2001) estimate a share model of daily time use in two-adult households that distinguishes between independent and joint activity participation. Their findings demonstrate the importance of employment levels and the presence of children on time allocation to such activities.

Wen and Koppelman (2001) postulate a decision-making process whereby households first generate out-of-home maintenance activities and then assign them to specific members for execution. They incorporate the number of such activity episodes in their modeling framework, which is a necessary prerequisite for forecasting models. A three-tier nested logit model is used to implement the modeling framework, which, despite its advances over previous research, still has two shortcomings. First, it is limited to couples that do not engage in out-of-home maintenance activities together, thereby ignoring joint activities. Second, although the nested logit model allows the household decisions to be estimated simultaneously, it does not deal with the ordinal nature of the first decision – that is, the number of out-of-home maintenance episodes.

Scott and Kanaraglou (2002) propose an approach to modeling activity generation which captures interactions between household heads in terms of all non-work, out-of-home activity episodes that they participate in on a daily basis. Although their approach does not distinguish between specific activity types, it has several strengths, in addition to capturing interactions, which define its usefulness for practical forecasting purposes. These strengths include explicit recognition of membership identity within a household, explicit recognition of activity setting (i.e. independent and joint activities) and explicit recognition of activity episodes as units of analysis. Although this approach to modeling activity generation within households is not necessarily a panacea, it does provide a robust alternative to other methods that are capable of capturing interactions between household heads.

Zhang et al. (2005) point out that although it has been realized that activities in multiple person households need to be coordinated and sometimes synchronized in time and space, there is still a lack of household-level models of activity-travel demand. The authors report on the development of a household task allocation and time use model based on a multi-linear group utility function. Using activity-travel diary data, a model of household task allocation and time use, incorporating the influence of travel time, was estimated. The model also allows quantifying the relative influence of the household members. The results indicate that, on weekdays, for nearly half of the households the husband mostly influences task allocation and time use, for one-fifth of the households it is the wife and the remaining households show an equal relative influence for the husband and wife.

Gliebe and Koppelman (2002) identify several spatially defined tour patterns found in weekday household survey data that describe this form of interpersonal decision-making. Using pairs of household decision makers as their subjects, they develop a structural discrete choice model that predicts the separate, parallel choices of full-day tour patterns by both persons, subject to the higher level constraint imposed by their joint selection of one of several spatial interaction patterns. They apply this model to the household survey data, drawing inferences from the household and person attributes that prove to be significant predictors of pattern choices, such as commitment to work schedules, auto availability, commuting distance and the presence of children in the household. Parameterization of an importance function in the models shows that in making joint activity-travel decisions significantly greater emphasis is placed on the individual

utilities of workers relative to non-workers and on the utilities of women in households with very young children.

AGENT-MEDIATED INTERACTION

Hackney and Marchal (2009) use the approach in an activity-travel context to model schedules individually and then reward individuals for being in the same place at the same time as their friends. As a consequence, joint activities are encouraged and occur more and more. A simulation tool is presented to let the experimenter construct and test hypothetical interdependencies between geography, socially-linked travellers, and activity-travel choices. Initially, any social network can be constructed and embedded in geography. It can remain fixed, or be adapted to the travel patterns of the agents. The interactions and exchanges between agents influencing socializing and/or travel behavior can be defined in substance and in time/space. The reward for socializing or being socially linked can be varied. Finally, the co-dependence of social factors and travel behavior can be studied. In this case, the schedules are iteratively changed to find an optimal schedule, however the activities are not explicitly modeled with joint intention.

People frequently interact face-to-face with each other. This could fulfill several needs: to gather information, to share an experience, to help one another, or for relaxation. Face-to-face interaction is sometimes crucial for relationships to continue. In order to model these activities, the transport modeling field is experiencing a shift from understanding "where are people going" and "what activity are they doing" towards "who are they interacting with". Ronald et al. (2010b) focus on social face-to-face activities by means of an agent based model to describe social activities between two people over time. The results show that the overall social network has an effect on the number of activities generated in the entire system and also between pairs of friends.

AGENT-MEDIATED NEGOTIATION MODELS

If participants have private information and modelling of agreements among them is required, then negotiation is especially useful. "Negotiation can be viewed as a distributed search through a space of potential agreements" (Jennings et al. 2001).

Given the dynamic environment in activity travel applications (Arentze and Timmermans 2007), rational agents obtain incomplete information from each other and the environment. Hence negotiation is applicable for joint activities among all those activities and travels because an agreement needs to be made regarding various items, such as day, time, location, and participants (Ma et al. 2010). The items to be agreed upon are referred to as issues. The simplest case involves two people and one issue.

One extension is two players and multiple issues, such as described by Fatima et al. (2009). Specifically, they consider the case where issues are divisible, there are time constraints in the form of deadlines and discount factors, and the agents have different preferences over the issues. Given these differing preferences, it is possible to reach Pareto-optimal agreements by negotiating all the issues together using a package deal procedure (PDP). However, finding equilibrium strategies for this procedure is not always computationally easy. In particular, if the agents' utility functions are nonlinear, then equilibrium strategies may be hard to compute. In order to overcome this complexity, they explore two different solutions. The first is to use the PDP for linear approximations of the given nonlinear utilities. The second solution is to use a simultaneous procedure (SP) where the issues are discussed in parallel but independently of each other. Eventually, they show that an approximate equilibrium for the PDP and the SP can be found in polynomial time; in some cases, the SP is better for one of the two agents and also increases the social welfare.

Another extension is to consider multiple players and one issue. Wainer et al. (2007) describe several protocols for scheduling a meeting with many participants. The only issue under discussion is the time of the meeting. presents a set of protocols for scheduling a meeting among agents that represent their respective user's interests. Four protocols are discussed: a) the full information protocol when all agents are comfortable with sharing their preference profile and free times; b) the approval protocol when only the preference profile can be shared; c) the voting protocol when only free time can be shared; and d) the suggestion protocol if neither preference nor free time can be shared. Results show that the voting protocol achieves the best solution 88% of the time. Simulation results for the suggestion protocol with different numbers of agents, different numbers of solutions, and different strategies are presented. The suggestion protocol is shown to be coalition-free. However it is limited by the number of issues under discussion.

Ronald et al. (2009) describe a protocol for scheduling joint social activities in which the activity host negotiates with the participants in parallel, combining their suggestions and preferences, however the participants do not have private information and rely on the host to make a decision.

Combining both dimensions of players and issues, true multiplayer negotiation protocols for multiple issues have been developed. Ito et al. (2007) present a protocol for interdependent multiple issues with non-linear utility. Their process involves sampling the issue space and then adjusting to find local maxima. Agents then submit their best options and a mediator determines the best solution.

Wu et al. (2009) describe a negotiation protocol for multi-player multi-issue negotiations with incomplete information and present results for a simulation with three agents and two issues. Again, the issues are assumed to be continuously-valued, and the utility function is assumed to be monotonically increasing, which are not enough for our current applications. In the current applications, there are discrete issues and continuous issues as well. The utility functions are not only monotonically increasing or decreasing but also convex or concave or circle like, etc. These require to extend and design new model applicable for more general cases.

Rindt et al. (2003) reports on the development of a simulation kernel for agent-based activity microsimulation based on the re-characterization of human activity as interaction between autonomous entities. They started from the idea that "human activity is the negotiated interaction of socially and physically situated individuals and settings" and as a result their kernel assumes that behaviour is adaptive. People, groups and resources (such as buildings) were represented as agents and used a variant of the contract net protocol to organise activities. The framework described was flexible and did not impose many restrictions on negotiations, but was not a complete model in itself.

AN ILLUSTRATION OF AGENT-MEDIATED NEGOTIATION MODELS

Issues

It is necessary to define the issues that are under discussion. In an activity-travel context, there are several possibilities. We limit our model to four issues: start time of the activity *ST*, location of the activity *LOC*, participants of the activity *PAR*, and the duration of the activity *DUR*. Among these issues, *ST*, *LOC*, and *PAR* are discrete issues while *DUR* is continuous. Other issues such as the day and the type of the activity are already determined before the negotiation begins.

The issues combine to define an activity. Each activity is evaluated using a non-linear utility function:

$$U_i = \beta(ST_i, LOC_i, PAR_i)\ln(DUR_i) - cost(DUR_i) \quad (1)$$

where $cost(DUR_i)$ represents the cost caused by taking this activity instead of other alternatives.

The beta function represents preferences for the discrete variables (start time, location, and participants) in Equation 2. But please note that the beta function will be different upon different variables.

$$\beta(ST_i, LOC_i, PAR_i) = \beta_1(ST_i)\,\beta_2(LOC_i)\,\beta_3(PAR_i) \quad (2)$$

The location and the participants in the activity are clearly discrete variables. Although the start time can be represented as continuous, it can be problematic. For example, the preference for starting an activity at 12pm or 9pm could be radically different, but a choice between 5pm or 5:10pm is relatively unimportant for scheduling.

The cost of the activity is represented by a factor ω multiplied by the duration:

$$cost(DUR_i) = \omega\, DUR_i \quad (3)$$

Combining the components, an overall utility curve is developed as shown in Figure 1 where several dots are used to indicate more curves exist there but not possible to exactly draw for lack of the concrete values of the variables.

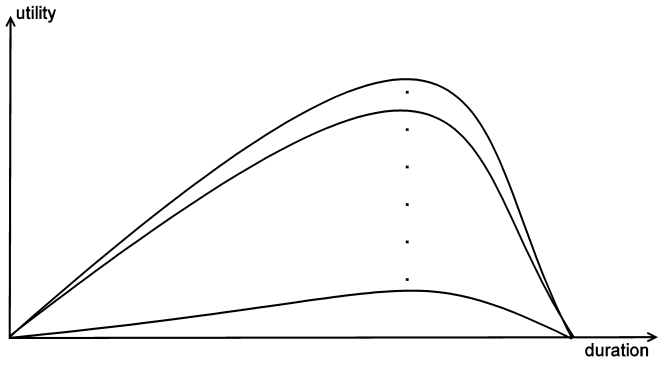


Figure 1. The overall utility curve for an agent.

Rules

Before the negotiation begins, each agent determines his personal utility function and negotiation deadline.

The negotiation takes place round by round. A round is a time duration when all the agents send their proposals sequentially. The order is generated randomly before every round. When it is an agent's turn, he calculates his best proposal according to his utility function, the current indifferent utility level, and the previous proposals from other agents, if any.

The negotiation will end when an agreement is reached or when one of the agents reaches the deadline. An agreement is reached when the same activity has been suggested by all agents, which means the values of ST_i, LOC_i, PAR_i, and DUR_i for all the agents are the same. If there is no agreement, every agent receives zero utility.

At the end of each round, if the negotiation has not terminated, then the agents will calculate the concessions that they are prepared to make and a new round will begin.

Strategy

In order to reach an agreement, agents sometimes have to concede to each other. A concession strategy is essential for an agent to decide the utility level that he is going to reach in the coming round.

Concession depends on the information that an agent has at that moment. If there is no proposal(s) in the history, then it is the first proposal in the first round of the negotiation. In case that there is no time pressure from the deadline, an agent determines the proposal which returns the highest utility.

If the agent is not the first one to give his proposal in the negotiation, then he will choose whether to follow the offer from the other agents or choose to determine his proposal by himself. There are two options available here:

1. The agent bases their next move on previous proposals by others by moving towards their proposals.
2. The agent concedes a fixed amount of utility. In this case, multiple offers can be made.

A rational agent will compare both techniques and select the one which provides the most utility. The concession processes until an agreement or the deadline is reached.

CONCLUSIONS

This paper has arguesd that negotiation is the key element for multi-agent simulations in activity travel applications. To this end, this paper has sought to capture the development of the simulations for activity travel application from where interaction is known within household, then where agents are adopted to behave for the owners and interact with each other, to where negotiation is acknowledged as the major concern. This process is also crucial to our understanding of the impacts of agent interaction and multi-agent simulations.

Although the overview presented in this paper focuses primarily on the evolution of negotiation models, there is still a considerable challenge to activity schedule models if negotiation will be included. For example, how to collect data to reflect the effect of the negotiation? How to integrate the negotiation into activity-travel scheduling process? Thereafter there is much opportunity to address these issues in the near future.

Finally the illustration presented here suggests a negotiation model to be designed according to the utility functions and issues which have been defined in activity travel applications. The negotiation model could also be combined with other techniques in order to investigate underlying decision rules and more complex/diverse behaviours (Hannes et al. 2010), and to enhance multi-agent simulations. Of course, many logistical and operational problems would need to be overcome and the comparison with other models is highly required, however, the development of this computerized approach clearly has the potential to transform the way of current activity-travel simuations and how we further study the observations obtained from data collections.

REFERENCES

Arentze, Theo, Claudia Pelizaro, and H.J.P. Timmermans. 2010. "An Agent-based Micor-simulation Framework for Modeling of Dynamic Activity-travel Rescheduling Decisions". *International Journal of Geographical Information Science*, 24:8, 1149-1170.

Arentze, T.A. and H.J.P. Timmermans. 2007. "Modelling Dynamics of Activity-travel Behavior". In *Proceedings 12th HKSTS Conference*, Hong Kong, China (CD-Rom: 30 pp.).

Balmer, M. 2007. "Travel Demand Modeling for Multi-agent Transport Simulations: Algorithms and Systems". *Ph.D. dissertation*, ETH Zurich.

Fatima, S. S.; M. Wooldridge; and N. Jennings. 2009. "An Analysis of Feasible Solutions for Multi-issue Negotiation Involving Non-linear Utility Functions". In *Proceedings of AAMAS*, 1041-1048.

Gilbert, Nigel. 2008. "Agent-Based Models (Quantitative Applications in the Social Sciences)", Sage Publications.

Gliebe, J.P. and F.S. Koppelman. 2001. "A Model of Joint Activity Participation". In *Proceedings of the TRB Annual Meeting*, Washington, DC.

Gliebe, J.P. and F.S. Koppelman. 2002. "A Model of Joint Activity Participation between Household Members". *Transportation*, Vol. 29, 49–72.

Golob, T.F. and M.G. McNally. 1997. "A Model of Activity Participation and Travel Interactions between Household Heads". *Transportation Research Part B: Methodological*, Vol. 31: 3, 177–194.

Hackney, J. and F. Marchal. 2009. "A Model for Coupling Multi-Agent Social Interactions and Traffic Simulation". In *Proceedings of the TRB Annual Meeting*, Washington, DC.

Hannes, E.; F. Liu; M. Vanhulsel; D. Janssens; T. Bellemans; K. Vanhoof; and G. Wets. 2010. :Tracking Household Routines Using Scheduling Hypothesis Embedded in Skeletons (THRUSHES)". *Transportmetrica, Special Issue "Universal Design"*, accepted for publication.

Ito, T.; H. Hattori; and M. Klein. 2007. "Multi-issue Negotiation Protocol for Agents: Exploring Nonlinear Utility Spaces". In *Proceedings of IJCAI*, 1347-1352.

Jennings, N.R.; P. Faratin; A.R. Lomuscio; S. Parsons; M.J. Wooldridge; and C. Sierra. 2001. "Automated Negotiation: Prospects, Methods and Challenges". *Group Decision and Negotiation*, Vol. 10. No. 2, 199-215.

Kostyniuk, L.P. and R. Kitamura. 1982. "Life Cycle and Household Time-space Paths: Empirical Investigation". *Transportation Research Record,* 879, 28–37.

Ma, Huiye; N. Ronald; Mengxiao Wu; T.A. Arentze; and H.J.P. Timmermans. 2010. "Multi-player Multi-issue Negotiation with Incomplete Information in Agent-based Activity-travel Scheduling". In *Proceedings of DDSS*, The Netherlands.

Rindt , Craig R.; James E. Marca; and Michael G. McNally. 2003. "An Agent-based Activity Microsimulation Kernel using a Negotiation Metaphor". In *Proceedings of the TRB Annual Meeting*, Washington, DC., US.

Ronald, N.; T.A. Arentze; and H.J.P. Timmermans. 2009. "Modelling Social Interactions between Individuals for Joint Activity-travel Scheduling". In Proceedings of IATBR, India.

Ronald, N.; H. Ma; T. Arentze; and H. Timmermans. 2010a. "Incorporating Power into Joint Social Activity Generation". In *Proceedings of the 3rd World Congress on Social Simulation*, Germany.

Ronald, N.; Virginia Dignum; and Catholijn Jonker. 2010b. "When will I See You again: Modeling the Influence of Social Networks on Social Activities". In *Proceedings of the Multi-Agent Logics, Languages, and Organisations Federated Workshhops*, France.

Scott, D.M. and P.S. Kanaroglou. 2002. "An Activity-episode Generation Model that Captures Interactions between Household Heads: Development and Empirical Analysis". *Transportation Research Part B: Methodological*, Vol. 36, 875–896.

Vovsha, P.; E. Petersen; and E. Donnelly. 2003. "Explicit Modeling of Joint Travel by Household Members: Statistical Evidence and Applied Approach". *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1831. 1-10.

Wainer, J.; P.R. Ferreira Jr.; and E.R. Constantino. 2007. "Scheduling Meetings through Multi-agent Negotiations". *Decision Support Systems*, Vol. 44, pp. 285–297.

Wen, Chieh-Hua and Frank S. Koppelman. 2001. "The Generalized Nested Logit Model". *Transportation Research Part B: Methodological*, 35:7, 627-641.

Wu, M.; M. de Weerdt; and H. La Poutre. 2009. "Efficient Methods for Multi-agent Multi-issue Negotiation: Allocating Resources". In *Proceedings of PRIMA (Lecture Notes in Computer Science 5925)*, pp. 97-112.

Zhang, Junyi; H.J.P. Timmermans; and A. Borgers. 2005. "A Model of Household Task Allocation and Time Use". *Transportation Research Part B: Methodological*, Vol. 39, pp. 81–95.

HEALTH CARE MANAGEMENT

Interoperabity in Healthcare

Miguel Miranda[1], Júlio Duarte[1], António Abelha[1], José Machado[1], José Neves[1] and João Neves[2]
[1]Universidade do Minho, CCTC, Departamento de Informática, Braga, Portugal
{miranda,jduarte,abelha,jmac,jneves}@di.uminho.pt
[2]Centro Hospitalar de Vila Nova de Gaia e Espinho, Portugal
j_neves@hotmail.com

ABSTRACT

Hospital Information Systems need to communicate in order to share information and to make it available at anyplace at whatever time. Indeed, these systems have to go along with the fundamentals of ubiquity and quality-of-care, being embedded in some forms of intelligent mechanisms in order to be useful for medical, clinical and administrative staff. In fact, to fulfill this goal, the information available must be judge in terms of its quality, acquired via a process of quantification of the extensions of the predicates that make their realm, i.e., speaking for a high degree of confidence on it on the part of the users. Admittedly, centralized systems are not a solution, they speak for themselves. The answer, once one must be able to exchange and make use of information, is interoperability.

KEYWORDS
healthcare interoperability

Introduction

The possibility of communication is one of the main characteristics of the human beings that opened diverse and valuable possibilities for mankind, ranging from the creation of languages to the cooperation in otherwise impossible intents. Cooperation and exchange of information is indeed one of the most relevant features in any context, in which different entities, virtual or physical, coexist. This aphorism is also valid in the case of information systems that exist at the same time or in the same place. Somehow, secluded information systems have little impact on their environment and create isolated information islands, that limit the flow of information, a key factor in systems applied to intersectional problematics.

Indeed, once it is set that independent information systems need to communicate and cooperate in order to enhance their overall performance and usefulness, the perception of integration and interoperation must be introduced at different conceptual levels with distinct objectives, but with intersecting principles. Although these fundamental truths are of great significance and value for cooperation and the flow of information within and among organizational units, they are based on disconnected behaviors and attitudes. In simpler terms, integration aims to gather and acquire information of distinct systems in order to reinforce or strengthen them, while interoperation concentrates on the continuous communication and exchange of information across cooperative systems.

The Institute of Electrical and Electronics Engineers (IEEE) defines interoperability as the "ability of a system or a product to work with other systems or products without special effort on the part of the customer". It also states that interoperability is made possible by the implementation of standards. In a similar but more complete manner, according to the International Organization for Standardization (ISO), interoperability is the ability of independent systems to exchange meaningful information and initiate actions from each other, in order to operate together to mutual benefit. In particular, it envisages the ability for loosely-coupled independent systems to be able to collaborate and communicate.

Principles of Interoperability

The intrinsic characteristics and methodologies for problem solving under an interoperable setting are complex to be determined in one single and absolute model, as numerous models can be found that are both valid and sound. These models attempt to classify interoperation approaches and activities using one or more sets of attributes, which define the exchanged data abstraction level, technological implementation, interoperation viewpoint and underlying purpose.

The Dublin Core Metadata Initiative (DCMI), an organization that aims to provide simple standards to facilitate the finding, sharing and management of information, created an interoperable framework at the system level, which is mainly based on the abstraction level of the exchangeable data or records, which is to be understood in terms of 4 (four) tiers, namely, Level 1 - Shared Terms Definitions - data components with shared natural language definitions; Level 2 - Formal Semantics of Interoperability - data is based on formal-semantics; Level 3 - Description Set of Syntactic Interoperability - data is structured according to shared formal vocabularies in exchangeable records; and Level 4 -

Description Set Profile Interoperability - data content is structured according to shared formal vocabularies, being bounded by a set of invariants on the exchangeable data or records.
These levels are oriented to the Dublin Core environment and its reference model, but can be extrapolated to other metadata ones.
Another model proposed by Tolk an Muguira aims to divide the conceptual layers of the interoperable factors according to the exchanged data abstraction levels, technological implementation and underlying purpose/focus. These 7 (seven) layers, as it is depicted in Table 1, ranging from level L0 to L6, define a scenario from no interoperability to a scenario with conceptual interoperability. In this model not only abstract concepts but also methodologies for problem solving are grouped according to their impact and potential to the interoperation process Wang et al. (2009).
A distinctive description adapted by Mykknen and Tuomainen (Table 2) for evaluating and classifying interoperation standards and models, is also of interest to the classification of the interoperable levels in itself. Although for more oriented approaches towards the identification of which interoperable aspects are not covered by a standard (i.e. from level 1 to 7), due to this underlying capability, it describes a model that put under a different perspective the interoperability process Mykkänen and Tuomainen (2008).
Far from comparing these models, several tendencies in the abstraction level of the exchanged information and the usages that are made of such information may be detected. One distinct borderline is the notion of syntactic and semantic level of information. In all models a clear difference there exists in the definition of shared syntactic rules, and shared semantic meaning and relations of the exchanged information. By setting shared syntactic rules, the content of all information within the exchanged data is normalized in such a way that it is possible to determine the nature of its content. However, only with shared semantic meaning and relations can the content be understood by the existing systems, enabling them to be connected with their equals, and at the same time to validate it according to the semantic relationships already defined.
The validation of information conforming to shared constraints is presented in the final levels of both the LMCIM and the DCMI, being given as the main characteristic defining the final level for interoperability. Indeed, the validation of exchanged data and rectification of incoherences is an important feature. Moreover, the use of shared constraints and formal vocabulary requires prior shared semantic models, something clearly stated in both models.
These concepts of interoperability are essential to understand and explore the particularities of interoperability in healthcare, where high levels of interoperation and exchange of information are required. These requirements are defined either by this environment or tools, such as the Electronic Health Record and the Decision Support Systems that are being integrated into it.

Interoperability in Healthcare

Although interoperability had been studied and its implications to care delivery had been considered, the level of interoperability among systems in most healthcare institutions remains frustratingly low Carr and Moore (2003).
In 2005 ISO released the ISO Health Informatics Profiling Framework (ISO TR 17119:2005) ISO/TC215 (2005), a vehicle tailored to describe standard artifacts, i.e. one of many kinds of tangible byproducts produced during the development of software in a healthcare environment. Similarly to other basic structures underlying a system, it aims to detail, classify and create relations among items within the domain area. This framework place such instruments under six perspectives, namely what, how, where, who, when and why, and three levels of specificity, that is to say conceptual, logical and physical design. This framework is oriented to organize and direct the development of a level of quality or attainment in the area, however it seems not to consider the existent norms such as Health Level Seven (HL7) and Digital Imaging and Communications in Medicine, which are decisive in most of existing interoperation processes in healthcare.
An intensive effort to develop standards adapted and optimized towards healthcare delivery had been developed before the ISO framework, and apparently resulted in more than two but not many consensually models in comparative evaluations. These standards have been able to give a definite structure or shape to low level interoperability in healthcare, in a firmly established and modular manner. Among these patterns HL7 is considered the most adaptable one in healthcare interoperability.
HL7 started as a mainly syntactic healthcare oriented communication protocol at the application layer, the seventh layer of the OSI communication model. This protocol defined the message structure to be exchanged by loosely connected healthcare applications by classifying the different types of messages involved in this environment with the aggregation of standardized segments.
The structuring and design of this standard, defining which artefacts of data should be transferred by a certain message, enabled and potentiated the application of HL7 in client-server architecture Ohe and Kaihara (1996). The most common implementation of this architecture using HL7 is based on distinct socket communication clients and servers, in which the client sends an HL7 structured message to the server, that upon processing sends an acknowledgement HL7 standardized message. The HL7 standard is not bound to this archi-

Table 1: Implication of LMCIM (adapted from Wang et al. (2009))

Level	Designation	Information abstraction level	Information defined	Contents clearly defined	Domain	Focus
L6	Conceptual	Common conceptual model	Assumptions or constraints	Documented conceptual models	Modelling abstraction	Compositionally
L5	Dynamic	Common execution model	Effect of data	Effects of information exchange		
L4	Pragmatic	Common workflow model	Use of data	Context of information exchange	Simulation implementation	Interoperability
L3	Semantic	Common reference model	Meaning of data	Content of information exchange		
L2	Syntactic	Common data structure	Structured data	Format of information exchanged		
L1	Technical	Common communication protocol	Binary data	Symbols of information exchanged	Network connectivity	Integrability
L0	No	No connection	NA	NA		

Table 2: Interoperability levels (adapted from Mykkänen and Tuomainen (2008))

Level	Designation	Meaning
7	Application life cycle interfaces	The life cycle of the application, including integration and development methods
6	Functional reference model	The domain-specific information or functional model or assumptions about the used methods
5	Semantics	The meaning of the defined interface elements
4	Functional interfaces	The defined functionality and information
3	Application infrastructure	The integration points in the distribution architecture of the participating applications
2	Technical infrastructure	The infrastructure for supporting the interface and communication technologies
1	Technical interfaces	The technologies used in the interfaces and implementations

tecture, but it is the most widely used in healthcare interoperability.

The initial versions of HL7 were uniquely syntactic, and according to the general models of interoperation are one of the lowest levels of this process. The current version 3 is opening the HL7 scope towards semantic interoperability, including the appropriate use of exchanged information in the sense of the communicating applications behaviour. This model presented in version 3 contains relations and metadata in a abstract level that may enable far higher levels of integration, namely by semantic interoperability and validation of exchanged information, using the relational mapping of each artefact. The Message Development Framework (MDF) is currently moving towards the HL7 Development Framework (HDF), therefore shifting the HL7 paradigm from message to architecture. Newer HL7 developments such as the EHR-S Functional Model and the SOA Project Group activities have been pushing this move Lopez and Blobel (2009).

The metadata and archetypes defined in HL7 allow it to organize both production and clinical data in clearly defined and connected segments and fields, which can be validated among artefacts. However, the implementation of version 3 is still rather limited as few service providers and institutions migrated already to this version.

Interoperability towards an Unified Electronic Medical Record

The Electronic Medical Record (EMR) is a core application which covers horizontally the health care unit and makes possible a transverse analysis of medical records along the services, units or treated pathologies, bringing to the healthcare arena new methodologies for problem solving, computational models, technologies and tools. This move must be centered on the patient, but considering the different perspectives of the medical staff.

The healthcare arena configures an environment where numerous specific solutions store in independent data structures the information of the patient, production and other significant data. Due to the complexity of each of the inner grounds of healthcare delivery, the possibility of a global information systems emerges as something complex and incomplete. However, the need

to gather significant information to be shared to other services and to communicate all relevant data related to the patient and the executed procedures, is not only of high value to the institutions, but also to the patient.
In order to aggregate and consolidate all significant information, a solid and efficient process of interoperation or integration must be developed. This process must take into consideration scalability, flexibility, portability and security when applied to EHR. The underlying EHR architecture must be component and model driven, forming or viewed as an unit apart platform-independent and platform-specific models.Blobel (2006) The complexity and sensibility of the exchanged information requires more than technological efficiency and pragmatic exchange of information. The dissemination of incoherent information and its introduction into the EHR may cause more than inconsistent records, they may give raise to a misdiagnose of bad choice of medical practices. In order to avoid this moral and ethical drawback a thorough validation of the exchanged and integrated information must be performed. The development of top level interoperability frameworks is henceforth of an intrinsic nature or indispensable quality for the healthcare environment. The multitude and intricacy of services that must be performed by the EHR and Group Decision Support Systems (GDSS), require such a framework or otherwise would be inefficiently intertwined with other essential solutions Miranda et al. (2008) Duarte et al. (2009).

Agency for Information Dissemination and Archive

The AIDA (an Agency for the Integration, Diffusion and Archive of Information) platform was developed in order to support the diffusion and integration of information generated in a healthcare environment. This platform imbues many different integration features, using mainly Service Oriented Architectures (SOA) and MultiAgent Systems (MAS) to implement interoperation in a distributed, specific and conform to a standard manner, comprising all the service providers within the healthcare institution Machado et al. (2008) Machado et al. (2007). Being MAS a field of research in Distributed Artificial Intelligence, this technology is intrinsically related with distributed problem solving, while being distinct in the intrinsic definition of an agent versus the properties of the general midlewares of the architecture called in its support Weiss (1999). Indeed, under this approach, a MAS subsumes a distributed architecture.

Case study

In table 3, it is presented the HL7 activity of AIDA-SOA in one portuguese major hospital, relating to the Emergency Department. Interoperation between different applications is made with the admission and outcome processes, as well as drug prescription and medical imaging and laboratory exam requests and results. 4 (four) channels are available for receiving HL7 messages and 4 (four) others channels are available for sending HL7 messages. Each message here corresponds to two ones because the acknowledgement procedure is implemented. The average number of messages processed by hour is 85 in sending channels and 83 in receiving ones. The number of messages processed by month is close to 500 000.
In table 4 it is presented the HL7 activity of AIDA-SOA in another portuguese hospital, relating to the Imaging Department and the Emergency Laboratory. Interoperation between different applications is made with the admission process, exams requests and results. For the Imaging Department, it was implemented the DICOM worklist procedure. The average number of messages processed by hour is 108 in the Imaging Department (one channel) and 291 in the Emergency Laboratory. The number of messages processed by month is close to 300 000.
In table 5 it is presented the SOA Activity of AIDA-SOA using web services. The data are related to a Laboratory Department using AIDA-SOA. Web services are used to publish laboratory reports and data for hospital accounting. Only in this department an average number of 166 343 times web services are executed. It means an average number of 291 runs by hour. In Figure 1(a), it is possible to see the distribution of web services running by hour. The interoperation process is not constant during the day.
In table 6 it is presented the SOA Activity of AIDA-SOA using web services. The data are related to the procedure of viewing exam results from AIDA-EHR, using AIDA-SOA. Web services are used to access laboratory reports and sending the report to be viewed in AIDA-EHR. There is an average number of 198 677 accesses. It means an average number of 276 runs by hour. In some day periods the number of accesses by hour is 763. In Figure 1(b), it is possible to see the distribution of web services running by hour. This interoperation implementation is a success of how to reach a free-paper hospital, eliminating paper circulation and avoiding the requests of exams because reports are not easy accessible.
On the other hand, it is obvious to conclude that with interoperation systems may be available 7 (seven)days by week and 24 (twenty four) hours by day. Upgrade procedures and software installation must be programmed in advance and the better day period to program these procedures is between 0:00 and 6:00 am.

Conclusions

In this paper it is presented interoperability as the path to follow for the next generation of e-Health systems.
The next step is to include embedded ontologies, follow-

Table 3: HL7 Activity - Emergency Department

	By Month	HL7 Channels	By Hour	By Channel
Sent	243 487	4	338	85
Received	239 248	4	332	83

Table 4: HL7 Activity - Imaging Department and Emergency Lab

	By Month	HL7 Channels	By Hour	By Channel
Imaging Dept	77 729	1	108	108
Emergency Lab	239 248	1	291	291

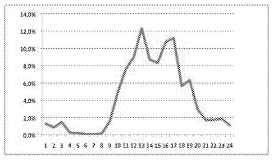
Table 5: SOA Activity - Lab Department

	By Month	By Hour	Max per Hour
Received	166 343	231	681

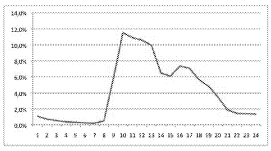
Table 6: SOA Activity - Exam Results Accesses

	By month	By hour	Max per Hour
0 to 6	6 755	38	73
6 to 12	78 676	437	763
12 to 18	84 636	470	659
18 to 24	28 609	159	322
Received Total	198 677	276	763

ing vocabulary, semantics and conceptual standards in Medicine, in order to achieve conceptual interoperability.



(a) Laboratory Department - By Hour



(b) Exam Results Accesses - By Hour

Figure 1: SOA Activity

REFERENCES

Blobel B., 2006. *Advanced and secure architectural EHR approaches. International Journal of Medical Informatics*, 75, no. 3-4, 185 – 190.

Carr C.D. and Moore S.M., 2003. *IHE: a model for driving adoption of standards. Computerized Medical Imaging and Graphics*, 27, no. 2-3, 137 – 146. ISSN 0895-6111. doi:DOI:10.1016/S0895-6111(02)00087-3. URL `http://www.sciencedirect.com/science/article/B6T5K-47VYGSX-C/2/f3a0fa01c1f2d10bab28817d09c7da9f`.

Duarte J.; Miranda M.F.M.; Abelha A.; Santos M.; Machado J.; Neves J.; Alberto C.; Salazar M.; Quintas C.; Ferreira A.M.S.F.; and Neves J., 2009. *Agent-Based Group Decision Support in Medicine*. In H.R. Arabnia; D. de la Fuente; and J.A. Olivas (Eds.), *ICAI*. CSREA Press. ISBN 1-60132-109-0, 115–121.

ISO/TC215, 2005. *Health informatics profiling framework*. Tech. Rep. ISO/TR 17119:2005.

Lopez D.M. and Blobel B.G., 2009. *A development framework for semantically interoperable health information systems. International Journal of Medical Informatics*, 78, no. 2, 83 – 103.

Machado J.; Abelha A.; Novais P.; and Neves J., 2008. *Quality of Service in Healthcare Units. European Simulation and Modelling Conference 2008*, 291–298. Bertelle, C Ayesh, A European Simulation and Modelling Conference OCT 27-29, 2008 European Technol Inst, Havre, FRANCE.

Machado J.; Alves V.; Abelha A.; and Neves J., 2007. *Ambient intelligence via multiagent systems in the medical arena. ENGINEERING INTELLIGENT SYSTEMS FOR ELECTRICAL ENGINEERING AND COMMUNICATIONS*, 15, no. 3, 151–157. ISSN 1472-8915.

Miranda M.F.M.; Abelha A.; Santos M.; Machado J.; and Neves J., 2008. *A Group Decision Support System for Staging of Cancer*. In D. Weerasinghe (Ed.), *eHealth*. Springer, *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 1, 114–121.

Mykkänen J. and Tuomainen M., 2008. *An evaluation and selection framework for interoperability standards. Information and Software Technology*, 50, no. 3, 176 – 197.

Ohe K. and Kaihara S., 1996. *Implementation of HL7 to client-server Hospital Information System (HIS) in the University of Tokyo Hospital. Journal of Medical Systems*, 20, no. 4, 197–205. URL `http://dx.doi.org/10.1007/BF02263391`.

Wang W.; Tolk A.; and 0002 W.W., 2009. *The levels of conceptual interoperability model: applying systems engineering principles to M&S*. In G.A. Wainer; C.A. Shaffer; R.M. McGraw; and M.J. Chinni (Eds.), *SpringSim*. SCS/ACM.

Weiss (Ed.), 1999. *Multiagent Systems – A Modern Approach to Distributed Modern Approach to Artificial Intelligence*. MIT Press, Cambridge, Massachusetts - London, England.

Interoperability Performance in a Healthcare Environment

Frederico Alves[1], António Abelha[1], José Machado[1], José Neves[1] and João Neves[2]
[1]Departamento de Informática, Universidade do Minho, Braga, Portugal
a44009@alunos.uminho.pt, {abelha,jmac,jneves}@di.uminho.pt
[2]Centro Hospitalar de Vila Nova de Gaia e Espinho, Portugal
j_neves@hotmail.com

KEYWORDS

Health Level 7, Server-Client Architecture, Performance, Mirth Connect

ABSTRACT

HL7(*Health Level 7*) provides interoperability standards that improve patient treatment, optimize the work flow, reduce the ambiguity and improve the knowledge transfer between healthcare facilities, governmental agencies and the provider community, i.e., all these processes must be handled with scientific accuracy and technical ability, with compromising transparency, responsibility and practicability. Indeed, any healthcare application need to communicate with its peers, once their added value is based on integration. In order to accomplish this goal we will consider performance studies of HL7 messages on a client-server architecture built on Mirth Connect and tested with client and server running on localhost, client and server on separate machines but in the same network and client and server on different machines and different networks.

INTRODUCTION

The HL7 protocol is one of several accredited standards of the *American National Standards Institute* (ANSI). Most standards are created for a particular area of interest, such as pharmacy, medical equipment, or medical imaging. HL7 operates with clinical and administrative data as a means to gather information in a uniform way, aiming at the development of standards, i.e., it defines a level of quality or attainment for transmitting information on the healthcare arena. Examples of HL7 messages include patient records, records of laboratory results and billing information, just to name a few. We intend to test the behavior of these messages in a client-server interface created through *Mirth Connect* software. The tests will be based on *Mirth Connect* and will be done locally, within the same network but on different machines and also on different networks (Miranda et al. 2009, Litwin et al. 1990).

CONFIGURATION AND SPECIFICATIONS

HL7 has established itself as a standard for exchanging information in a healthcare setting (Orgun and Vu 2005, Huang et al. 2003). *Mirth* allows messages to be filtered, processed, and routed based on user-defined rules. The web-based interface and the channel creation associate applications created with the components of the software engine *Mirth Connect*. *Mirth* uses a channel architecture to connect with other HL7 systems. These channels denote endpoints (*inbound* and *outbound*), filters and transformers. Multiple filters and a chain of processors may be associated with a channel. The *Mirth* web interface allows the reuse of filters and processors across multiple channels. The *endpoints* are used to configure connections and their particularities. The input parameters are used to designate the type of *listener* to use for incoming messages, such as TCP / IP or a web service. The output terminals are used to designate the destination of sent messages, such as an application server, a *JMS queue* or a database. In this study it will be used three channels(Hyun et al. 2009, Ohe and Kaihara 1996, Kim and Yun 2003), namely:

The Server

This channel has as a LLP Listener(*Lower Layer Protocol*) input parameter, setting up a server with an IP address that will be used to respond to The Client with an ACK message(*Acknowledgement*) saying that the message HL7 sent by the client was correctly or incorrectly received, in order to proceed to the next message. As output terminal, this channel has a *File Writer* whose function is to set a backup (in a hard drive) of the network messages exchanged between The Client and The Server.

The Client

The Client has as input parameter a *File Reader* that reads the HL7 messages present on the hard drive, once they had been sent to The Server. The Client receives its ACK, checks if everything is true and correct as a fact and moves forward to the

next message. The Client output terminal operates with a *LLP Sender* in order to be connected to The Server channel to exchange messages, returning the answers to The ACK Generator, so that it may store the ACK messages from The Server on the hard drive.

The ACK Generator

This channel has a simple *Channel Reader* as an input parameter, whose function is to read the The Client messages and to provide a backup of the ACK messages. In relation to the output terminal, The ACK Generator has a *File Writer*, but it works only for ACK messages, that must not be confused with others types of HL7 messages.

The *File Writers* present in both The Server and The ACK Generator, are configured to save messages in the hard drive with the corresponding *Timestamp*, that sets the messages creation times, that will allow as to conduct the performance tests that are described below (Figure 1).

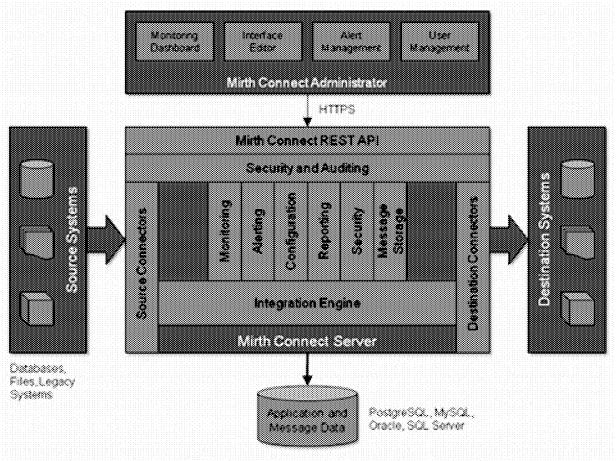


Figure 1: The Mirth Connecting Architecture

PERFORMANCE TESTS

In this section it will be referred not only the various *Mirth Connect* software tests in terms of performance, but also those in which The Client, The Server and The ACK Generator are all on the same machine. It will be also mentioned a test in which The Server is on a separate machine apart from The Client and The ACK Generator, but on the same network, as well as considered the situation when The Server is down for a given period of time, in order to perceive if The Client attempts to send the pending messages back to The Server, to test if the *Mirth Connect* software is fault tolerant.

SERVER AND CLIENT ON THE SAME MACHINE

In **Table 1**, **Table 2** and **Fig. 2** are presented the tests results. These tests were made in a machine running Windows, with all the channels (i.e. The Server, The Client, and The ACK Generator) running in parallel.

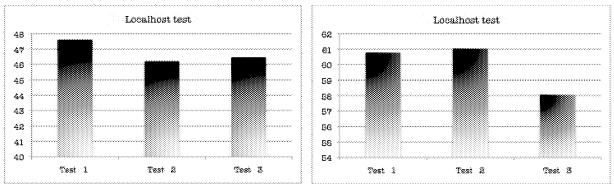


Figure 2: The HL7 messages performance results in seconds

Table 1: The HL7 messages details about the localhost test

Messages	Test 1	Test 2	Test 3
Total time(seconds)	47,615	46,217	46,457
Number of messages	2000	2000	2000
Time per message(seconds)	0,0238	0,0231	0,0232

Table 2: The ACK messages details about the localhost test

ACK's	Test 1	Test 2	Test 3
Total Time(seconds)	60,781	61,038	58,052
Number of messages	2000	2000	2000
Time per message(seconds)	0,0304	0,0305	0,0290

THE CLIENT AND THE SERVER RUNNING ON DIFFERENT MACHINES ON THE SAME NETWORK

In **Table 3**, **Table 4** and **Fig. 3** are presented the tests results. These tests were made in a machine running Windows, with the channels The Client and The ACK Generator running on a machine and The Server channel running on another machine, but on the same network.

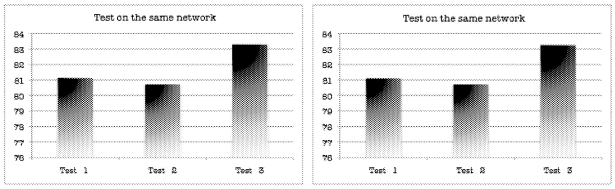


Figure 3: The HL7 messages performance results in seconds

Table 3: The HL7 messages details with The Client and The Server running on separate machines but on the same network

Messages	Test 1	Test 2	Test 3
Total time(seconds)	81,120	80,714	83,288
Number of messages	2000	2000	2000
Time per message(seconds)	0,0406	0,0404	0,0416

Table 4: The ACK messages details with The Client and The Server running on separate machines but on the same network

ACK's	Test 1	Test 2	Test 3
Total time(seconds)	81,103	80,713	83,249
Number of messages	2000	2000	2000
Time per message(seconds)	0,0406	0,0404	0,0416

THE ALERT SYSTEM

The Mirth Connect alert system provides the administrator with email or sms alerts when an error situation occurs. This alert system can be applied to any channel running on any Mirth Connect instance on any machine, sending the emails or sms to a pre-defined reported list. Some of the errors are now listed in table 5 :

Table 5: Errors

200	Filter	300	Transformer
301	Transformer conversion	302	Custom transformer
400	Connector	401	Document connector
402	SMTP connector	403	File connector
404	HTTP connector	405	FTP connector
406	JDBC connector	407	JMS connector
408	MLLP connector	409	SFTP connector
410	SOAP connector	411	TCP connector
412	VM connector	413	Email connector

The error message in the email only refers to the error number, such that only experienced users of Mirth Connect may understand it. However, in the future, Alert System implementations will look at:

Global Monitoring:

- Channel state monitoring (Started, Stopped, Paused).
- Channel processing monitoring (Is the channel locked up?).
- Connector state monitoring (Busy, Disconnected, Initialized).

Monitoring Metrics taken Globally and by Channel:

- No messages received in a given period of time.
- Too many messages received in a given of time.
- Number of queued messages that have been reported in a given period of time.
- Number of error messages that have been reported in a given period of time.
- Number of error messages received in a given of time.
- Number of filtered messages received in a given period of time.

Message Exception Monitoring:

- Severity of exceptions definable by the user.
- Severity threshold for a given alert.
- Setting what exceptions to alert on.

Advanced Scheduling of Alerts:

- Time of the day.
- Days of the week.
- Dynamically assign alertee's based on time

Alert Dispatch Timing Threshold:

- Prevents to dispatch the same alert in succession.
- Specify a threshold in which a channel can send the same alert.

Dispatching Methods:

- Keep simple dispatching method (email and SMS).
- Add ability to route an alert to a channel to proceed to other endpoints.

Transformer Alerts:

- Beef up the API to include these features, so that the same functionality is available within a script.

Template:

- Use values from rules defined in the alert in the error template.

Escalation:

- Define a period of time in which the alert has to be "deactivated" before it is escalated and sent to another recipient.

SIMULATION

First of all, The Server channel summary is filled with an incoming data type of HL7 v2.x and its initialized state is set as stopped, i.e., when the channels are deployed, they are in a stop state, and the administrator must start them other by hand or by changing its initial state to resume the job. The Server source connector type is a LLP Listener, being the administrator able to set the IP address of the machine where to run The Server and the respective port, the received Timeout (in ms), and the buffer size (in this case with 65536 bytes). It can also state if the batch is processed or not, and define the LLP frame encoding (can be ASCII or Hexadecimal), the message start (0x0B) and end (0x1C) character, as well as the record separator and end of segment character, both 0x0D. It may also set the message encoding and personalize the ACK

messages. Successful ACK code is AA and it means that the message was received successfully. Error ACK code is AE and it means that an error occurred when the message has been processed. At last, rejected ACK code, AR, means that the message was rejected.

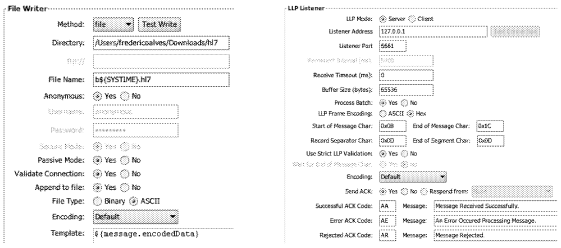


Figure 4: Server channel source and Server Channel Destination.

The Server channel has a File Writer as a destination. This destination writes files anonymously to the hard drive in a passive mode with an ASCII file type which contains the encoded data of the HL7 message (Figure 4) .
The Client channel summary is the same as The Server channel one, only with a different name. Regarding the source of The Client, it is a File Reader, that looks at files in a hard drive with a .hl7 extension.

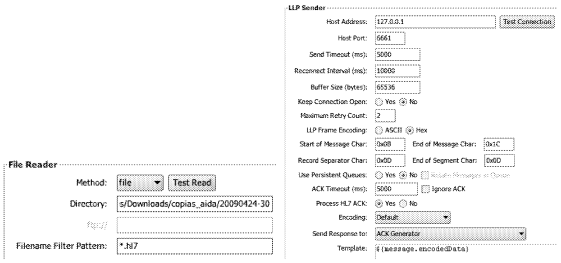


Figure 5: The Client Channel Source.

The Client destination is a LLP Sender that communicates with The Server channel LLP Listener to exchange the HL7 messages and the ACK's (Figure 5). It is here where the administrator defines the message sent timeout, the retry count when the sending process fails, the ACK timeout (in ms) and the possibility of using persistent queues to process all HL7 messages.
The ACK Generator has a simple Channel Reader as a source, that reads The Client answers to The Server channel and saves the ACK messages from such a transaction. As destination it has a File Writer that writes the ACK messages to the hard drive for future reference.
There are several connectors that can be used as source or destination points in every channel. In the Figure 6 are specified all the connectors that can be used.
At last, the administrator has the possibility to interact with some default plugins that Mirth Connect brings with it.The administrator can also add plugins made by himself/herself. There are some examples of them on Figure 7.

Figure 6: The Connectors that can be Used.

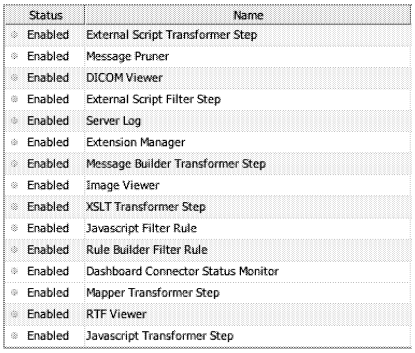


Figure 7: The Plugins that can be Used.

DISCUSSION AND CONCLUSIONS

Regarding testing on localhost, all channels are running on the same machine and all data is written and transferred within the same machine. Taking a benchmark test on the same network, the messages flow much faster in the localhost test, as it was expected, and the average time to send 2000 messages take 46.7 seconds, while the test on the same network, running The Client and The ACK Generator from The Server, the time of transmission of 2000 messages take 81.7 seconds. This time is relatively smaller than a localhost one, as expected. However, this is the test that matters, regarding that the messages have to be handled in a clinic environment.

With respect to the ACK messages, in terms of the localhost test, they take a great amount of time to be processed when compared with the processing time of the HL7 messages in the same test. This is due to the fact that all three channels are running simultaneously on the same machine, which hinders the simultaneous processing of the HL7 messages and the ACK messages. This does not happen in the test when The Server is separated from The Client and the The ACK Generator channels. The splitting of these processes, in terms of the HL7 messages and the ACK messages, is due to the fact that the channels responsible for processing the

HL7 messages and the ACK messages are apart into different machines. The ACK messages are stored on the hard disk for backup with the letter "a", being associated with a timestamp that reports the moment of archiving (e.g. a${SYSTIME}.hl7). The same procedure applies to the HL7 messages, here using the letter "b" (e.g. b${SYSTIME}.hl7. The ${SYSTIME} saves the current time of the machine on the file name.

About sending only one message, the conclusion is that this process is very fast given the volume of messages in the hospital per day (about 5000 messages). In the localhost test the time per message is around 30 ms and in the same network test is 40 ms, which concludes that the performance of this software is very good. Here is an example of the Dashboard Status Panel and its behaviour: (Figure 8):

Status	Name	Received	Sent	Errored
Stopped	Server	1297	1297	0
Started	ACK Generator	584	584	0
Started	Client	586	584	2

Figure 8: The Error report after stopping the Server channel

It was conducted a test to determine if the software Mirth Connect is tolerable to errors. In order to achieve this goal, and after 500 messages had been sent, The Server channel was stopped for one minute and then resumed. It was perceived that after stopping The Server, The Client and The ACK Generator tried to send the message again, and again (it tried ten times more). Two errors were reported by The Client for not having any response from The Server; each attempt with a time-out of 3000 ms (Figure 9):

Connector Info	Event	Info
Source: LLP Listener (HL7V2 -> HL7V2)	Disconnected	
Source: LLP Listener (HL7V2 -> HL7V2)	Busy	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52490
Source: LLP Listener (HL7V2 -> HL7V2)	Disconnected	
Source: LLP Listener (HL7V2 -> HL7V2)	Connected	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52490
Source: LLP Listener (HL7V2 -> HL7V2)	Done	
Source: LLP Listener (HL7V2 -> HL7V2)	Done	
Source: LLP Listener (HL7V2 -> HL7V2)	Disconnected	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52489
Source: LLP Listener (HL7V2 -> HL7V2)	Done	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52489
Destination: File Writer - Destination 1	Busy	Result written to: /Users/fredericoalves/Downloads/hl7/b$...
Source: LLP Listener (HL7V2 -> HL7V2)	Done	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52489
Destination: File Writer - Destination 1	Busy	Result written to: /Users/fredericoalves/Downloads/hl7/b$...
Source: LLP Listener (HL7V2 -> HL7V2)	Busy	
Source: LLP Listener (HL7V2 -> HL7V2)	Busy	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52489
Source: LLP Listener (HL7V2 -> HL7V2)	Connected	Sender: 127.0.0.1:6661 Receiver: 127.0.0.1:52489

Figure 9: The Dashboard Status Panel of Mirth Connect

When resuming The Server channel all the processes pursue normally. On the other hand, and to avoid such kind of errors from any channel, *Mirth Connect* has a system of alerts that can be configured (Benson 2010a).

REFERENCES

Benson T., 2010a. *Principles of Health Interoperability HL7 and SNOMED*, Health Informatics, chap. Why Interoperability is Hard.

Benson T., 2010b. *Principles of Health Interoperability HL7 and SNOMED*, Health Informatics, chap. Standards Development Organizations.

Huang E.W.; Hsiao S.H.; and Liou D.M., 2003. *Design and implementation of a web-based HL7 message generation and validation system. Internation Journal of Medical Informatics.*

Hyun S.; Shapiro J.S.; Melton G.; Schlegel C.; Stetson P.D.; Johnson S.B.; and Bakken S., 2009. *Iterative Evaluation of the Health Level 7—Logical Observation Identifiers Names and Codes Clinical Document Ontology for Representing Clinical Document Names: A Case Report. Journal of the American Medical Informatics Association.*

Kim I.K. and Yun J.H., 2003. *Intelligent Agents and Multi-Agent Systems*, Lecture Notes in Computer Science, chap. Agent-Based Intelligent Clinical Information System for Persistent Lifelong Electronic Medical Record.

Litwin W.; Mark L.; and Roussopoulos N., 1990. *Interoperability of multiple autonomous databases. ACM Computing Surveys.*

Miranda M.; Duarte J.; Abelha A.; Machado J.; and Neves J., 2009. *Interoperability and Healthcare. Proceedings of the ESM 2009.*

Ohe K. and Kaihara S., 1996. *Implementation of HL7 to client-server Hospital Information System (HIS) in the University of Tokyo Hospital. Journal of Medical Systems.*

Orgun B. and Vu J., 2005. *HL7 ontology and mobile agents for interoperability in heterogeneous medical information systems. Computers in Biology and Medicine.*

MODELLING AND SIMULATION OF THE MATERNITY FOR THE UHCO

Khaled Belkadi, Nawal Zahaf
LAMOSI, University of Mohamed Boudiaf, USTO,
BP 1505 Oran M'Naouer, 31000 Oran, Algeria
E-mail: belkadi1999@yahoo.com, zahafn@yahoo.fr

Alain Tanguy
LIMOS, CNRS UMR 6158, University Blaise Pascal,
Les Cézeaux, 63173 Aubière Cedex, France
E-mail: tanguy@isima.fr

KEYWORDS
Hospital, Maternity, ARIS modelling, SIMULA model, Discrete-event simulation.

ABSTRACT

The maternity hospital is an essential service of the main hospital. The aim of this paper is to model and simulate the maternity and related services of University Hospital Centre of Oran (UHCO) in Algeria using ASDI methodology. It is mainly composed of services providing cares and surgery for pregnant women, mothers, babies and persons suffering gynaecological or oncological pathology.

Our aim is to study the utilization rates of the human resources of this service. This will enable us to improve service performances. To achieve this goal, we first use the ARIS tool, graphical algorithms and queuing networks to specify the knowledge model then we simulate models thanks to the SIMULA language to implement action models and with ARENA in a soon future.

INTRODUCTION

The maternity hospital of the University Hospital Centre of Oran (UHCO) is a building of 16 sections or wings on 4 levels. It is mainly composed of admission and examination service, low way delivery, caesarean surgery, neonatal care, gynaecological pathology care, surgery and oncology services.

Hospital systems are complex systems in which problems have to be solved such as the size and number of their critical resources, the enhancement of their efficiency or obviously the understanding of their operations. These problems concern performance evaluation they can be solved using modelling and simulation.

Modelling is a decision aid tool which prevents important financial investments. This paper describes the modelling and the simulation of the main maternity services in the hospital system. The ASDI (Analysis, Specification, Design and Implementation) (Gourgand and Kellert 1991) modelling methodology is adapted and used for this system. It is based on the construction of two model classes: the knowledge model and the action models. The specification of the knowledge model is mainly realized using ARIS tool (Architecture of integrated Information System) (Sheer 2002) and detailed algorithms. The action models are implemented thanks to the SIMULA language. We focus this study on the admission and the delivery services so as to study the utilization rates of personnel of these services.

MATERNITY OF THE UHCO

The maternity of the UHCO in Algeria is classified as maternity of level three; it takes care of normal and risked pregnancies, evacuated patients from other public or private health establishments of the city and other cities, all kinds of deliveries, gynecological interventions, resuscitation of the newborn child and the training of the student doctors. It is organized to respond to different patterns of women care, it is structured in wings, and each of them takes care of a particular category of patients. It is equipped with an obstetrical block, two surgical units, a wing of oncology and chemotherapy, a wing of pediatric care for new born premature or having health problems, two wings for pregnancy with risk, another wing for the gynecology, wings of hospitalization pre-operational, post-operational and post-partum. It also includes a wing of external consultations, a laboratory, an echography room and a training room.

The maternity of the UHCO receives daily patients' tens arriving in different states and many emergency degrees (a bordering average thousand admissions a month) and have a medical and paramedical staff for whom the workload is in continuously increasing. Material resources are already overloaded.

Each day and during 24 hours, patients come to the maternity service of the UHCO for many reasons, delivery, pregnancy problems and gynecology diseases for which the maternity personnel must ensure a fast and immediate care. For that, there is a single point of filtering named admission wing which takes care of patients coming to the maternity, examines them and decides if they must be hospitalized and in which wing they will be affected. The maternity organization is based on thirteen wings for the patients and a wing for premature newborns, each wing has its medical and ancillary medical team. The patient arrivals to the admission wing are random. If the state of a patient is very critical she passes in priority and if it is less urgent (waiting does not worsen her state) she is put in a single queue served by two boxes of consultation. The doctors examine the patient and decide to hospitalize her or let her leave with or without treatment prescription and/or medical assessment. Patient's admission is done without being limited by the capacity of the service because it is not possible to ask a baby to delay his hour of birth or to be unaware of the state of a patient in danger.

The maternity of the UHCO is characterized by the obstetric and gynecological activities under a framework of formation of the student doctors. There are two categories of patients with and without pregnancy and the taking care

of one or other is followed upon its emergency degrees. Several cases are possible. The case of a pregnancy at end which presents the first indicators of delivery is directed to the wing of waiting deliveries and if she is in labour she is affected to the obstetrical block. If she is about a prolonged pregnancy, a delivery by low way or Caesarean section must be caused. In the case of a pregnancy with risks a hospitalization is necessary to preserve patient's and baby's lives. In case of patients without pregnancy, the pathology determines the emergency degree and thus the assignment of the patient to the gynecology wing or oncology or even to operating theatre block if it is an extreme urgency case. In certain cases where the patient requires a follow-up and whose case is not urgent, she is directed through an orientation letter to the external consultations wing to take an appointment. The external consultations are done during the day and during the working days by the student doctors, the assistant doctors and the professors such as each one of them has one day of consultation and its patients.

The Caesarean sections are made without planning, on the other hand the surgical interventions of gynecology are done followed upon a weekly operational program established each Wednesday of the week S for the week S+1 in which the doctor's team taking part in the intervention must be designated. A programmed intervention proceeds in the operating theatre block reserved for the planned acts and during the working hours of the working days.

Setting apart, the external consultations and the planned interventions, all other wings are operational each day and during the 24 hours. For that, a medical and ancillary medical guard is obligatory every day (from 15:00 to 08:30) during all the week end and the public holidays. For the medical personnel, the planning of guard is done each month by designating a team by guard which assures the maternity wings care. Contrary to the ancillary medical personnel which have a mode of guard work (24h and 12h). In the maternity of the UHCO, the formation aspect is very important. To have a good formation, the medical personnel are affected to all the wings during the period of training in medical specialty that lasts five years using a rotational system each six months or more (according to the importance of the wing).

MODELLING METHODOLOGY

The modelling methodology ASDI has been adapted to hospital systems (Gourgand and Combes 1994; Mebrek and Tanguy 2006; Mebrek, et al. 2007). The knowledge model describes the structure and the operating principle of the system in a natural or graphical language; it is built thanks to three subsystems (logical, physical and decisional). An action model is a translation of the knowledge model in a mathematical formalism or in a programming language enabling the evaluation of chosen performance criteria.

The main goal of the modelling methodology is to establish a knowledge model that is as generic as possible and that allows the execution of the action models specific to the systems of the domain. The knowledge model remains an open model which is enhanced by each domain systems study. The management of the knowledge and the execution of the action models imply the help of an open modelling environment in order to include new and more efficient methods and tools. The modelling environment eases information exchange between the project members and helps the conception of action models during the extraction of the information from the knowledge model. It is mainly composed of a modelling methodology, analysis and specification tools, data base, statistics, animation and evaluation tools. It is an attempt to introduce automatism in the modelling process with the formalisation of the knowledge, the analysis of the data to determine the characteristics of the system, the operational research and the simulation for the evaluation. Graphical representations and animation tools help verifying proper operation of the model. The first knowledge model of the hospital logical system has been formalised thanks to the ARIS tool that is appropriate to describe organisations, processes and activities (Green and Roseman 2000), as well as entity-relationship models (Chen 1976).

MODELLING PROCESS

Modelling is a set of techniques that provides the ability to study and understand the structure and the operating principle of a system. We use three rules to build a model that represents the reality: a model must be like the reality, a simplification of the reality and an ideal view of the reality.

The knowledge model is a formalised description of the system that contains the acquired knowledge during the observation phase of the existing system or the specification of the topology and functioning stated by the designers. The action model is a translation of the knowledge model using a mathematical formalism (for example an analytical method which takes advantage of the queuing network theory) or in a programming language (for example a simulation language). It is directly usable and states the performances of the modelled system without using direct measure. Exploitation of the knowledge model and of the action model is called modelling process. This process is generally iterative and consists of four steps which are the elaboration of a system knowledge model, the translation of this knowledge model into an action model, the exploitation of the action model to evaluate the performances of the system, and the interpretation of the results to deduce the modifications to be made on the system. Each step includes a verification and validation phase. A knowledge model has a wide application area.

In order to use ARIS to design a knowledge model (Rob and Brabänder 2008; Güngöz 2004), several modelling hypothesises are to be taken into account:
- Each activity (function in ARIS) is linked to one or more organisational units of the hospital system (admissions, care unit, operating room, the pharmacy, the delivery services and so on);
- Each event possesses its own information document, it is used by several processes and it is referenced in one or more documents of the information system (medical file of the patient, file of the operating room suite, etc.);
- The referenced documents provide the knowledge concerning the key processes.

To match our modelling goals, we chose the ARIS tool-set and we retain two representation types (Sheer, 2000):
- The event-driven process chain (EPC) in order to show that the processes have a well defined structure and to control the logical subsystems flows;
- The organisational structure for the decisional subsystem to detail the relationships in and between the services.

KNOWLEDGE MODEL EPC

The sequence of functions in the sense of an enterprise process is represented in process chain. In these chains, it is possible to indicate the departure and arrival events for each function. The events trigger functions and they are generated by them. Event-driven process chain (EPC) represents the organisational structure of the enterprise, i.e. the representation of the relationships between the data view, objects, the functions and the organisational views. An EPC (figure 1) describes the sequencing of functions. For each function, an initial event and a final event are defined. The events trigger the functions and a function generates events. Function and event are represented by a rounded rectangle and a hexagon.

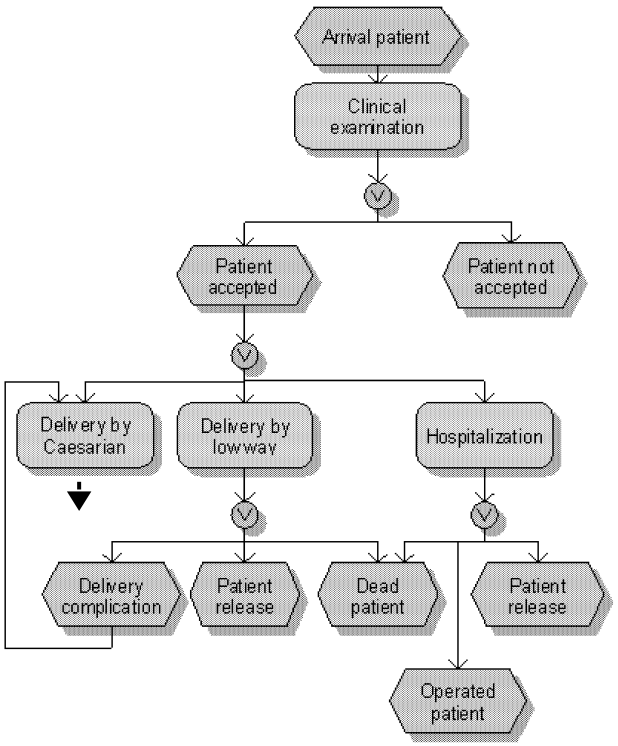


Figure 1: Maternity EPC fragment

As the events define the state or the condition that triggers a function as well as the end state, the start and end nodes of an EPC are always events. An event can trigger several functions simultaneously and a function can provoke several events. To represent the links and the processing loops of an EPC, the system uses a connector (or ruler) which as the shape of a circle. Figure 1 shows the specification of the maternity delivery service by a simplified EPC. Two connectors may be used: an *And* operator and an *Exclusive Or*. The *Exclusive Or* connector allows a patient to select only one case. The EPC are detailed thanks to algorithms and block diagrams.

ACTION MODELS

The Maternity action (or simulation) models of the UHCO are represented by the SIMULA models and later by ARENA models too.

Queuing network model

Figure 2 shows how to use the queuing network model to represent an action model of the maternity. Presently we modelled the main wings of the maternity hospital: mainly admission/examination wing, Low way delivery and surgery blocks for caesarean sections. We collected statistics for guard functioning mode during the weekend (Friday and Saturday in Algeria). So, this study concerns a functioning with few material and human resources.

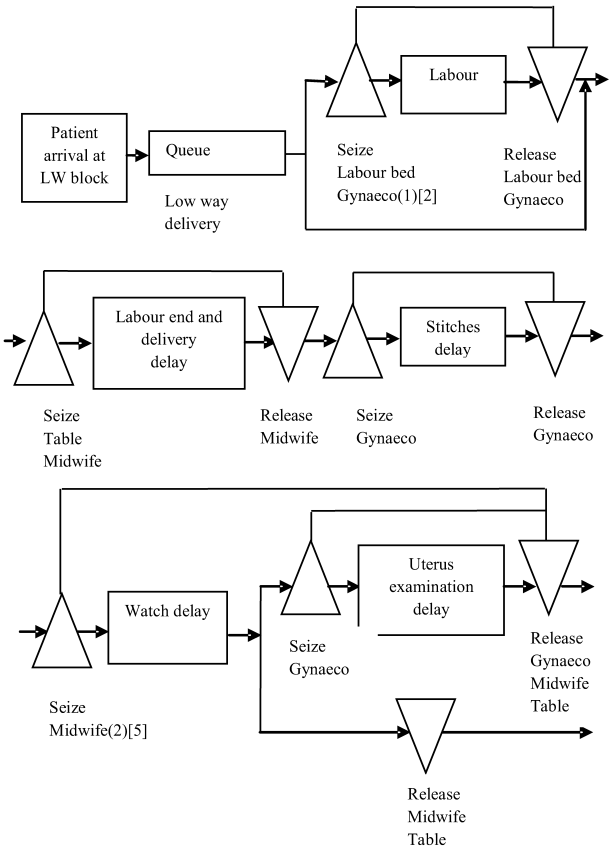


Figure 2: Queuing network model of delivery by low way

The system functioning is the following: a patient arrives, two gynaecologists receive and examine her. The patients are classified in 6 types: no care, simple care, low way delivery, caesarean section, transfer to different wings and last: patients coming from other sites directly to caesarean wing. Figure 1 introduces the admissions process and the dispatching of the 5 first patient types.

Figure 2 introduces the material and human resource management. The delays and service times are model parameters as resource numbers. The labour begins on a labour bed. Sometimes 2 doctors are seized by a patient because one of them is a gynaecologist student. When a

resource is seized (*n*)[*m*] that means *n* resources are needed to treat *m* patients. These numbers are not provided when the value is 1. Some activities have a higher priority than others for instance, treatments have a higher priority than supervision or control of a lot of patients. During the labour, the patient may be routed to the caesarean wing. Patient moves are done by gynaecologists or nurses.

The main activities are described by random delays characterized by their probability distributions. Some statistics do not appear in the scheme of figure 2 and depend on the evaluation tool. The patient arrival is a Poisson's process, its rate is a parameter (λ=0.02 patient per minute) and patient types depend on proportions (0, 4, 11, 3, 11). The direct arrival rate at caesarean block is λ/10.

SIMULA model

The SIMULA language allows us to implement different object oriented simulation models categories (Dahl and Nygaard 1965, 1966) and complex algorithms. It includes coroutines and processes of discrete events simulation. Many classes exist that extend the language possibilities regarding transactions management and statistical computations. The Gpsss class provides base objects such as the service, the storage, the transaction notions as well as the statistical region. Moreover, a simulation report is generated automatically. This class can therefore be used in GPSS programming with all the object-oriented capacity of a simulation language and fast implementation.

We realized a simulation based on pseudo random exponential arrivals during 24 hours for one weekend. A patient arrival list may be used; it is recorded in an Excel file. Obviously this list has a great impact on the utilization rates of material and medical personnel.

The reception and examination activities of admission wing are modelled by uniform probability distributions and resources. The service in the examination box is represented by a *Facility* class of Montréal Gpsss external class. This facility has only one server but provide a complex service due to the use of resources modelled thanks to *Storage*. The resource management is very simple: a patient *Transaction Enter*s or *Leave*s a *Storage* or a statistical *Region*.

The treatment and supervision durations are regarded as uniform distributions which intervals depends on the activity. The uniform distribution as very good properties such as simplicity, compact support and not too bad first approximation of more complex distributions. It is easy to replace it by triangular or specific ones. The table 1 describes some parameters of the models. The durations are given in minutes.

The three regarded wings have the following resources:
Admissions: 2 gynaecologists,
LW delivery: 2 gynaecologists, 5 midwives,
C delivery: 2 gynaecologists, 1 AS (care assistant), 3 Ide (nurses), 3 Iade, 1 Mar (anaesthesiologist).

RESULTS AND INTERPRETATIONS

The obtained results with models written in the SIMULA language and using Montréal Gpsss class are given in figures 3, 4 and 5. They are presented in 3 parts.

Table 1: Duration parameters of the model

Admissions, Low way and Caesarean delivery wings	Duration (min - max)
1 Questions	2-5
2 Examination	5-10
3 Care	2-5
4 Move	5-10
5 Labour begin	30-480
6 Labour end	10-60
7 Delivery	10-30
8 Stitches	10-20
9 Supervision	30-120
10 Uterus review	10-20
11 Preparation	10-20
12 Induction	10-15
13 Operation	30-120
14 Wakeup	5-45
15 Move	5-10

The first one gives a fragment of the simulation trace including the patient type and number, time and event (figure 3). We can see that patients arrive at admission wing (type 1 to 5) or directly at the surgery block (type 6).

Type	Number	Time	Event
5	1	26.669	begin_OtherW
3	1	35.944	begin_LwDel
5	1	50.028	end___OtherW
3	2	61.428	begin_LwDel
3	3	77.110	begin_LwDel
4	1	147.620	begin_CaDel
5	2	177.561	begin_OtherW
5	2	199.860	end___OtherW
2	1	259.669	begin_Care
5	3	265.497	begin_OtherW
2	1	273.264	end___Care
6	1	293.311	begin_ExtCD
5	3	297.494	end___OtherW
3	4	297.805	begin_LwDel

Figure 3: Beginning of the simulation trace

The second part provides the performances of the resources (personnel, tables and beds) and statistical measures (durations of supervision). The resources are modelled by *Storage* class of Gpsss. For each resource figure 4 gives mainly: the number of treated patients, the mean treatment times and the utilization rates concerning the tasks having high priority for medical personnel.

Figure 4 shows that the average time transit for examination and dispatching is short (about 18 minutes) and that the personnel seem to be not very overloaded - excepting admission wing gynaecologists (52% busy).

The third part provides extra resource statistics, they are automatically computed by means of *Region* classes of Gpsss. Figure 5 presents results related to low priority tasks of gynaecologists and midwives: the number of processed patients, average contents and times including waiting

durations of patients. These results are useful to increase treatment quality. A simple computation shows that supervision occupies almost all medical personnel time. The patients wait only few minutes about less than 11 minutes.

```
start time=    0.00
end   time=   1440.00
* facilities *
               avg.    avg.time
      entries contents  transit  status
Admis     37     0.46    17.77    free
* storages *
              avg.    avg.time contents
     entries contents  transit now  max capa util.
AS         8     0.14    24.36   1    1    1  13.53%
IDE        8     0.33    59.02   1    1    3  10.93%
IADE       4     0.31   110.15   0    1    3  10.20%
MAR        4     0.31   111.79   0    1    1  31.05%
TabAdm    37     0.20     7.69   0    1    1  19.77%
GynAdm    37     1.04    20.26   0    2    2  52.06%
BedLWB    11     1.26   164.67   3    4    8  15.72%
TabLWB     8     0.79   142.78   0    3    5  15.87%
GynLWB     5     0.05    14.30   0    1    2   2.48%
MiWLWB     8     0.32    58.36   0    3    5   6.48%
BedCaB     4     0.08    28.73   0    1   10   0.80%
TabCaB     5     0.33    95.84   1    1    2  16.64%
GynCaB     4     0.46    83.06   0    2    2  23.07%
```

Figure 4: Resource performances

```
* regions *
             avg.   avg.time contents nonzero zero
   entries contents transit now  max transit entr.
SupGyLW 11    1.26   164.67  3    4  164.67     0
SupMWLW  8    0.43    77.04  0    2   77.04     0
WtPAdmi 37    0.28    11.07  0    3   21.55    18
WtPLWB  11    0.00     0.00  0    1   11
WtPCaB   5    0.03     7.72  1    1   19.31     3
```

Figure 5: Supervision and patient waiting

These results are obtained for one random list of patients that represents only 24 hours in a year so the study should be completed with the simulation of a lot of lists and confidence intervals computation after in depth statistical study.

We experimented different simulation tools (Belkadi 2007; Belkadi and Tanguy 2009, 2010) (Moussa and Belkadi 2009) such as Witness, Flexsim (Flexsim 2009), ARENA, QNAP2. They have their own abilities: quick development, animation, complex computation, and poor adaptivity. They should be useful for maternity evaluation and the study of its functioning.

CONCLUSION

We presented the maternity hospital of the UHCO and the usefulness of the knowledge model specified thanks to the ARIS tool and algorithms, as well as the transition from this model to the action (simulation) model implemented with the SIMULA language and an extra queuing network.

The SIMULA language and the Gpsss class are very suitable in modelling, simulation and statistics computing. Flexsim and Witness have interesting ability in quick development and animation but the automatically obtained results are not exactly those we need.

A computation of confidence intervals will provide finer performances. We need more specific criterion to take into account more precisely the supervision and teaching activities of medical personnel.

A wider statistical study of maternity service should provide finest parameters. More real patient lists should show the real behaviour of the service that is necessary to tune, the dimension of the system and to optimize its functioning. A PhD student will have to study the maternity in depth, to propose a new organization so as to reduce patient's stay and personnel load. Several themes can complete this work: modelling and simulation of other connected services, using other modelling and simulation tools like ARENA and animation tools, studying the planning of hospital services and the driving of the hospital.

REFERENCES

Belkadi, K. 2007. “Regional military and University Hospital of Oran (RMUHO)”, Internal Report No. 1, Computer Science Department, USTO, Oran.

Belkadi K. and A. Tanguy 2009. “Modelling and Simulation of the Stomatology Service for the RMUHO”. ESM 2009 October, Leicester, England.

Belkadi K. and A. Tanguy 2010. “Modelling and Simulation of the Ophtalmology Service for RMUHO”. WSEAS-ACMOS 2010 May, Sicily, Italy.

Chen, P. 1976. “The entity relationship model–Toward a unified view of data”.*ACM Transaction on data base system*, Vol 1, N°1.

Dahl O. J. and K. Nygaard. 1965. *SIMULA - A Language for Programming and Description of Discrete Event Systems*. Introduction and User's Manual. Norwegian Computing Center, Oslo.

Dahl O. J. and K. Nygaard. 1966. “SIMULA - An Algol-based Simulation Language”. *Communication of the ACM*,No. 9.

Flexsim Products Inc., 2009. “*Flexsim – Process simulation software*”, http://www.flexsim.fr/solution1.html.

Gourgand, M. and C. Combes. 1994. “*A modelling environment for hospital systems”.* Ph.D., Clermont2 University, Clermont-Ferrand, France.

Gourgand, M. and P. Kellert. 1991. “Modelling environment design for manufacturing systems”. *3ème congrès international de Génie Industriel*, Tours, France.

Green, P. and M. Roseman. 2000. “Modelling: an ontological evaluation”. *In Information systems, ATED process*, Vol. 25.

Güngöz. Ö. 2004. “ARIS Architecture of Integrated Information Systems”. Objectives lecture 2004, Aoyama Gakuin University AGU summer term Japan, Url: http://www.bpm-agu.com/downloads/Summary_ARIS.pdf.

Mebrek, F. and A. Tanguy. 2006. “Modelling and discrete events simulation of the imagery pole of a modern hospital”. *6ème Conférence Francophone de Modélisation et Simulation - MOSIM’06,* 3-5 avril 2006, Rabat, Morocco.

Mebrek, F., k.; Belkadi; M. Gourgand; A. Tanguy. 2007. “Modelling and Simulation of the External Sterilization for New Hospital Estaing”. *Colloque sur l'Optimisation et les Systèmes d'Information* (COSI'07) 11-13 juin, Oran, Algeria.

Moussa M. and K. Belkadi. 2009. “Simulation of flows in a service of the imaging RMUHO”, 2nd International Conference SIIE-2009, Hammamet, Tunisia.

Rob D., E. Brabänder 2008. “ARIS Design Platform: Getting Started with BPM”. Springer.

Sheer, 2000. “Aris 5.0 Method”, Edition IDS Scheer AG, Saarebruck.

Sheer. A.W. 2002. “*ARIS-Business Process Modelling*”. Springer.

NUCIA - NURSE CALL SIMULATION IN AGENT ENVIRONMENTS

Tim Vermeulen, Koen Vangheluwe,
Joris Maervoet, Katja Verbeeck
CODeS, KaHo Sint-Lieven (KULeuven)
Gebroeders Desmetstraat 1,
9000 Gent, Belgium
email: name.surname@kahosl.be

Piet Verhoeve, Brecht Stubbe
Televic
Leo Bekaertlaan 1,
8870 Izegem, Belgium
email: P.Verhoeve@televic.com,
B.Stubbe@televic.com

KEYWORDS
Multi-Agent simulation, health care communication systems, emulation

ABSTRACT

The aim of the NuCiA project is to build a simulation and test environment in which future nurse call systems can be developed and tested. The need for a flexible, virtual test system can easily be explained. Hospitals and care institutions are getting bigger while there is a tendency for managing them as a single unit. It is becoming much harder to implement realistic test setups in hardware. The number of simultaneous actions on a variety of devices, the complex user scenario's and the integration of the system with others, makes the problem of testing very challenging. Moreover, a fast evolution of electronics not only imposes shorter times to market, more powerful and intelligent reasoning within the devices itself provokes a big impact on the distributed intelligence and architectural design decisions of the nurse call system.
In the NuCiA project (Nurse Call simulation In Agent environments) we study agent based simulation (ABS) to cope with the current challenges of developing and testing future nurse call systems. One of the challenges ahead will be to apply typical ABS modeling and link it with hardware components and devices to realistically test the operation of virtual end-users in a combined real and virtual test-system.

INTRODUCTION

The NuCiA project, short for Nurse Call simulation In Agent environments, is a joint initiative with Televic Health Care (THC), a Belgian company active in the sector of communication and information systems tailored to E-Health applications. THC 's business consists of developing, designing and manufacturing digital networks for nurse call, intercom, building management, access control, patient entertainment, care registration and assisted living. A major concern for THC is to thoroughly test, analyze and verify their systems before being implemented.

Until now Televic's solutions were mainly developed in a classic way, based on electronic hardware components. Using these components an extra system is build that can be used for testing, analysis and verifications. Specific test scenario's are developed to verify both the global functioning and certain aspects of the system (cfr unit testing).

One of the main problems with the current approach is scalability. Building very big test configurations is physically restricted by the number of devices and the number of people operating them. In this case, fictional load settings can be used, however true realistic testing of hospitals with about 3000 beds is impossible, which may result in unexpected problems at the care institutions' sites. Another problem with the current test approach is the time needed for testing. A first problem is the set-up time. Preparing a new configuration can take up to 14 days. Alternating configurations can easily take one day work. Recent tests with a new processor on a complex hardware component took 3 months.

Besides scalability, other trends influence the testability of the system as well. The fast evolution of electronic devices for instance will not tolerate long development times. More powerful and intelligent devices will make it more complex to make good design decisions. Moreover, the influence of distributed intelligence on the network behavior of nurse call systems is not yet known.

All these trends point out the need for a more flexible virtual test-system. In this paper we propose the use of agent based simulation (ABS) (Bonabeau 2002) to tackle the above mentioned challenges. This virtual system should enable faster, non-real-time simulation. However it must still be possible to couple the test-system with real hardware and mix simulation with real test components in one test-system. The latter aspect is, as far as we know, new for ABS.

This paper is organized as follows. In the next section we shortly describe and motivate the use of ABS in our work. In the following section we discuss in more detail the challenges we have to tackle. Thereafter, the current version of our test system is described together with the results of some performance measures. In a final section we discuss future work and challenges.

AGENT BASED SIMULATION

Agent based simulation or ABS has been proved to be a very convenient bottom-up way of modeling complex systems (Bonabeau 2002). ABS is typically suited for modeling systems driven by the interactions among its entities. Complex emergent behavior at the system level can easily be revealed even when the entities (typically called agents) involved have fairly simple behavior. In contrast to discrete event simulation, where it is common practice to model people as deterministic resources ignoring their performance variation and their pro-active behavior, ABS allows for modeling a heterogeneous population. Each agent might have personal motivations and incentives, and groups and group interactions can be represented. Nurse call system simulation is certainly characterized by the presence of many human users and the interactions between them and the system. This makes ABS surely a good alternative to scale up testing. However, maybe new to ABS, in this project a unique mix of virtual and real testing is aimed at.

CHALLENGES

Large Scale Simulation

Because hospitals can be big and will possibly keep getting bigger in the future, scalability is an important issue. To compensate the bad scalability of Repast Simphony parallelization and distribution are considered. Tests were done with a Repast plug-in using Terracotta (Terracotta 2010, Tatara 2008) that is intended to be used for distributing Repast projects on a network cluster. Adapting the current project to make it possible to use Terracotta is however very challenging.

Generating Test Scenario's

One of the final goals is to make it possible to examine the flexibility of this system in development. Therefore tests should be done with many configurations. One will thus need an easy way to create and change configurations. Tests will be done with normal, standard, configurations, corner-case scenario's and large scale configurations. For the moment the configuration is loaded via a simple XML file, but to be able to use more complex scenario's this will need to be extended. It should for instance be possible to load floor maps of hospitals with locations of corridors, rooms, devices and patients.

Validation Of The Televic Layered Model

Another goal of the test-system is to help in determining the system-architecture. Televic recently introduced a new model for their hard- and software using a layered architecture describing both physical and logical functionalities. Mainly the logical part is new, the layered architecture will improve the flexibility of the system for future Nurse Call applications. The E-health sector is a fast evolving domain. With the right abstractions, this system can correspond to these evolutions. The new test-system can also help with making these correct abstractions and with validating this new model.

Hybridization

As explained above, the nurse call test-system has to combine virtual simulation with testing real components. By adding this hybridization it will be possible to validate the software components by replacing them by real hardware. Another advantage is that it will allow us to reuse the pure hardware tests as before. To realize this, we will need to study the synchronization between the software and the hardware. Also the link with both IP-based components and non-IP-based components should be studied.

Measures And Criteria For Evaluation

An important issue is to evaluate and validate the test-system. This can be done by measuring. A statistical analysis of the test-results will be compared with results of real tests. First we will need to determine what is important to measure. This is also an important question in MAS research in general, how to determine the reliability of self-organizing, autonomous agents. So we will evaluate different criteria according to currently characterized criteria for the evaluation of self organizing systems (Kaddoum et al. 2009).

Visualization

It will be very important that the test-system has a scalable, user-friendly visualization. It became clear that a visualization of everything at the same time will cost too much CPU-power and will slow down the simulation too much. We will need to find a way to allow the user to view a global overview and zoom in to specific parts to see more details. A proper user-interface will also allow Televic to demonstrate their hardware in the hospital configuration of new potential customers.

THE CURRENT TEST-SYSTEM

In a first trial, Repast Simphony (Collier 2002, North and Vos 2005) was used for simulation. Repast is a multi-agent simulation tool, Symphony is a version written in Java. Because poor scalability results the decision was taken to leave Repast Simphony. We build an own multi-agent environment in C# based on the scheduler of Repast.

Loading The Data

First the physical hospital configuration needs to be imported. This is done via an XML configuration file which contains the (initial) location of the corridors, the rooms, the devices, the patients and the nurses.
Next the logical model is loaded. This model has a lay-

ered structure that maps on the physical configuration. The lowest layer groups parts of devices who cooperate to achieve a specific local goal. For instance the red buttons and red lights on devices in a room section are grouped, the red lights will light up if the red button is pushed. This is controlled by a scenario specified for this group. Next groups are combined in bigger groups for more global functionalities. The subgroups, handling the nurse calls of a specific room section, of a complete corridor can be combined with the hand-held of a nurse to notify her if a patient calls.
This layered model should be mapped on the physical devices in an intelligent way. One needs to think about defining a subpart for a complete room or for a section of the room, pertaining one patient. This mapping can also be dynamic. At night nurses might be responsible for more then one corridor.

Testing Configurations

In the current tests only hospitals with a fixed number of corridors and a fixed number of rooms, having each the same number of beds, are generated and tested. In every room is one room-terminal and every bed has the same devices. All beds are occupied by patients and every corridor has the same amount of nurses. The behavior of the patients is defined to call the nurse with a random frequency. Nurses will look for rooms with lighted red lights.

Agent Architecture

Patients, nurses and devices are all modeled as agents and connected to each other through corridors, rooms and room sections. Besides these physical agents, we also have a logical agent model which is hierarchical organized. Agents of the lowest level of the logical model represent specific functionalities of devices , while higher level agents of the logical model group these specific functionalities in a hierarchical way with on top one root combining them all. Agents of the logical model will be able to communicate with their parent and children agents. The logical agent model maps on a logical layered model of communication within the company.

Visualization

The two different architectures are visualized in a different way. For the physical architecture we draw a map with less or more details and the logical architecture is visualized as a tree.

The Physical Architecture
A first realization in the visualization is the view of the physical parts of the hospital.
One can first view an overview of the complete hospital with limited details as in Figure 1. One can view the different corridors with their rooms, but no devices in the room are shown except for the lights outside the room.

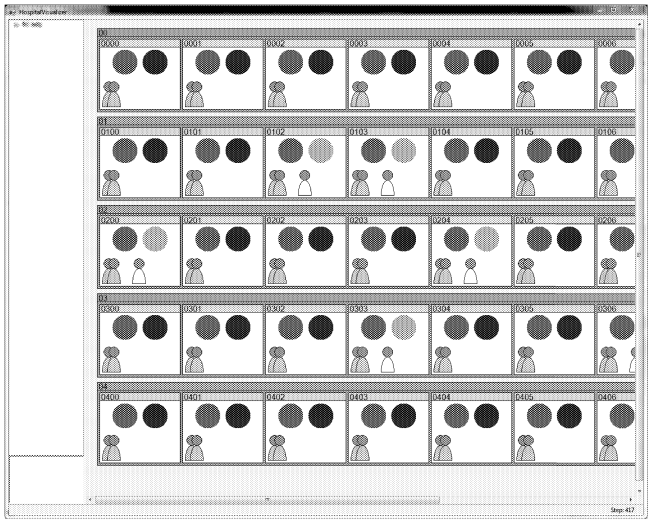


Figure 1: The overview of a complete hospital.

The number of patients and nurses that are present is also visualized. One can select a specific corridor to view more details, the room sections are shown, patients may or may not be present.

The Logical Architecture
A second part of the visualization is the logical structure. On layer 1 there will be one part for the whole hospital, responsible for executing general scenario's. That one part on layer 1 will have many children on layer 2. Parts on layer 2 can for instance be responsible for a specific corridor or specific tasks in a specific corridor. One can select the upper layer part to show it together with his children. Next one can select lower layer parts to see their children, as shown in Figure 2. On the lowest layer, parts are matching specific functionalities of devices.

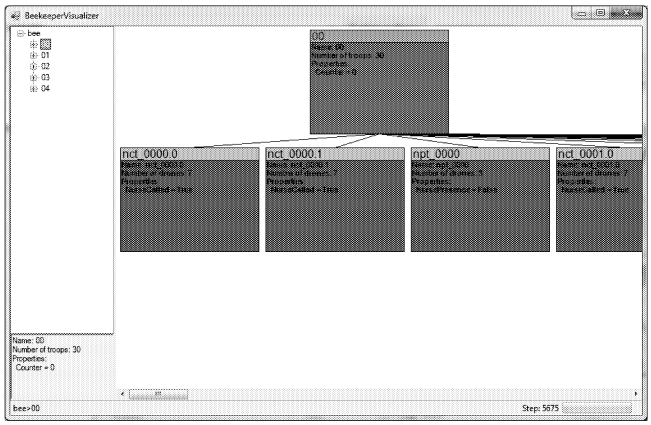


Figure 2: A view of the second layer part with its third layer children.

Complex Scenario's

The functionality of the different logical components of the system is defined by scenario's. Those scenario's consist of binary functions with one ore more premises and one or more consequences. These scenario's can be defined in a generic way. Some of them can be very complex. Scenario's are loaded from an XML input-file and added to the responsible logical components who will verify and execute them.

Performance Measures

The results of some of the performance tests are shown in Table 1 and visualized in Figure 3. With these measures we want to show both the performance gain by writing our own display and the current maximum capacity. The default display of Repast shows all agents all the time which causes a slow visualization.
An other conclusion that we can make is that our current test system is not capable of testing bigger configurations as they will result in slower, far from real-time behavior.

Table 1: Results from measures of the time for 100 steps averaged and recalculated for 1 step.

Configuration						
corridors	1	5	5	10	10	15
nurses per corridor	1	1	1	1	1	1
rooms per corridor	5	5	10	10	20	20
patients per room	1	1	1	1	1	1
total number of agents	30	146	271	541	1041	1561
Mean time for one step for all agents [ms]						
with Repast visualization	42	91	171			
with own visualization	2	8	16	30	55	80
without visualization	1	7	14	27	55	84

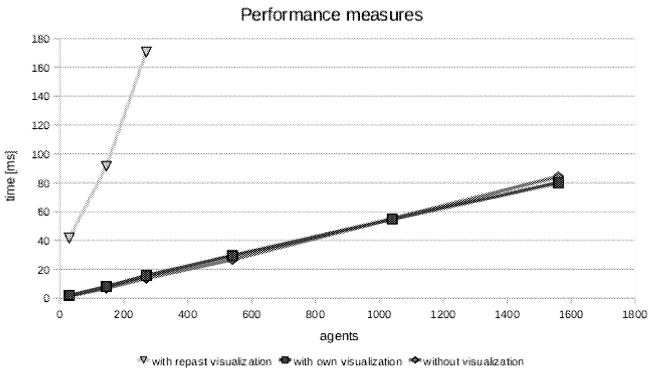


Figure 3: Chart showing the results of the performance measures.

FUTURE WORK

A big challenge left for the Nucia project is to be able to integrate hardware into the simulation, so that the existing hardware test components can be reused. We believe that the simulator we designed so far, is generic enough to make it possible to test hardware as well. One will be able to attach a physical device to the computer and add it to the simulation, replacing a simulated device.
A second part of the future work has to do with testing the flexibility. We will need tools that are able to easily create and alter hospital configurations and system configurations for those hospitals. This will also be important when using the test-system to simulate systems for existing hospitals.
An other core issue that needs further study is the development of scalable visualization. Because this is one of the most time consuming parts of the simulation, further optimization is needed so that the use of calculation power can be kept at minimum.

ACKNOWLEDGMENT

This project was funded by the Flemish agency for Innovation by Science and Technology (IWT090687).

REFERENCES

Bonabeau E., 2002. *Agent-based modeling: methods and techniques for simulating human systems. Proc Natl Acad Sci U S A*, 99 Suppl 3.

Collier N., 2002. *RePast: An Extensible Framework for Agent Simulation.* URL `http://www.econ.iastate.edu/tesfatsi/RepastTutorial.Collier.pdf`.

Kaddoum E.; Gleizes M.P.; George J.P.; and Picard G., 2009. *Characterizing and Evaluating Problem Solving Self-* Systems. Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, Computation World*, 0, 137–145. doi:http://doi.ieeecomputersociety.org/10.1109/ComputationWorld.2009.100.

North M.J. H.T.C.N. and Vos J., 2005. *The Repast Simphony Runtime System.* In *Proceedings of the Agent 2005 Conference on Generative Social Processes, Models, and Mechanisms.* Chicago, USA.

WEB REFERENCES

Tatara E., 2008. *Running Terracotta on a Repast Simphony Project.* URL `http://repast.sourceforge.net/docs/reference/SIM/Running%20Terracotta%20on%20a%20Repast%20Simphony%20Project.html`. Online tutorial.

Terracotta I., 2010. *Terracotta.* http://www.terracotta.org.

TRANSPORT INFRA-STRUCTURE LAYOUT DESIGN

GENERALISATION OF ARTIFICIAL NEURAL NETWORK SUPPORTING PLATFORM TRACK ASSIGNMENT WITHIN RAILWAY STATION SIMULATION

Michael Bažant, Jan Fikejz and Antonín Kavička
Department of Software Technologies
Faculty of Electrical Engineering and Informatics
University of Pardubice
Studentská 95, CZ-532 10 Pardubice
Czech Republic
E-mail: Michael.Bazant@upce.cz, Jan.Fikejz@upce.cz, Antonin.Kavicka@upce.cz

KEYWORDS

Artificial neural network, decision-making support, platform track assignment, railway station simulation.

ABSTRACT

The paper deals with the problem of a decision-making support (realised with the help of an artificial neural network) related to the platform track assignment problem. The mentioned problem is supposed to be solved in simulation models reflecting the railway traffic within passenger stations. The utilisation of two-layered perceptron (as a special kind of an artificial neural network) has been studied during the last period. The standing methodology expects that for each input station direction one trained neural network is available. The presented modified approach pays attention to the construction of just one generalised network, which has a potential to be exploited as a universal decision-making support (related to the platform track assignment) for the trains approaching the station from an arbitrary input station direction.

INTRODUCTION

Assignment of platform track to an arriving train represents a typical decision making task for dispatchers within passenger railway stations. If the inbound trains follow the timetable, the platform tracks are commonly assigned according to a priori created plan. In the case of a delayed arriving train the dispatcher is supposed to make an operative decision (potentially considering a set of substitutive tracks) about a relevant platform track assignment.

The above mentioned assignment problem should be properly solved also within particular simulation models (Adamko and Klima 2008), (Kavička et al. 2007), (Kavička and Bažant 2007), (Kavička et al. 2006). This is important for example, in the case of investigations focused on passenger stations suffering from frequent delays of arriving trains (Chakroborty and Vikram 2008). Assigned tracks ought to correspond to resulting decisions made by experienced station dispatchers in reality. After assigning a platform track to a relevant train many other specialised algorithms (involved within a simulation model) are carried out (e.g. an algorithm focused on a setting train route respecting the rules of an interlocking system, an algorithm calculating the dynamics of the train movement to the assigned platform etc.).

PLATFORM TRACK ALLOCATION

A station dispatcher (managing real railway traffic) partially subjectively evaluates potential platform tracks that are suitable for an assignment to a delayed incoming train. The ultimately assigned track represents the best solution according to the expert knowledge of a particular dispatcher using certain criteria. The same strategy is applicable for a relevant simulation model. The first stage of the original submitted approach is focused on delayed trains from one arrival direction only. Platform track selection is primarily related to construction of a *set of admissible tracks* - denoted as $S_{La,Ld}$, the elements of which can be admissibly assigned to a considered (delayed) inbound train. The platform tracks contained in the set $S_{La,Ld}$ involves those platform tracks, which are exploitable for the trains arriving from the line track L_a and departing through the line track L_d.

Examples of relevant sets of admissible tracks considering an illustrative track layout (fig.1) are as follows: $S_{L1,L1} = \{k_4, k_2, k_1, k_5, k_7\}$, $S_{L1,L2} = \{k_4, k_2, k_1, k_5, k_7\}, \ldots$, $S_{L5,L5} = \{k_7, k_9, k_{11}\}$. The sets can be further reduced according to the specific conditions (e.g. some tracks are removed from the relevant set because of their insufficient length with respect to the considered train etc.).

The next step is associated with the final selection of a particular platform track (from a set of admissible tracks), which represents the most suitable solution (according to specific criteria) for the considered train in time of its real arrival. The mentioned criteria take into account the knowledge of station dispatchers and are formed as follows:

A: Track vacancy degree at the moment of train arrival.
B: Track vacancy period with regard to station sojourn time of an arriving train.

C: Occupation of the neighbouring track at the same platform (owing to the selected track) by a connection train.

D: Further technical and technological preferences of the track in respect to an arriving train.

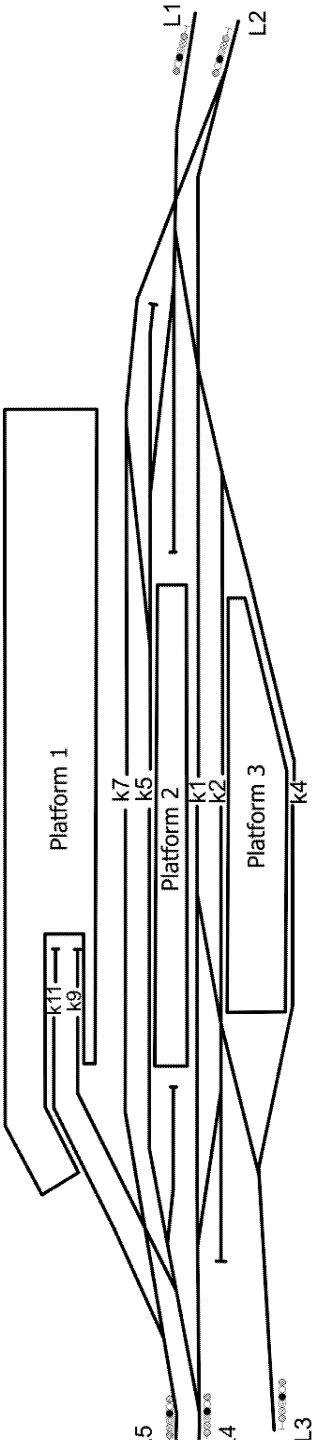


Figure 1: Track layout of an illustrative passenger station

A specified track assignment problem is obviously connected with multiple-criteria decision making focused on selection of variants (Figueira et al. 2004). The finite set of variants corresponds to the above mentioned set of admissible tracks containing the tracks, which stand as candidates for a relevant assignment. If the criteria are at our disposal (*A*, *B*, *C*, *D*) and it is possible to calculate the criterion values (the relevant calculation is described in (Bažant and Kavička 2009)) of investigated decision variants then a criterion matrix can be created. An element of criterion matrix y_{ij} expresses the value of a criterion i (where $i = 1, 2, ..., 4$ reflects criteria *A*, …, *D*) for the relevant variant/track k_j. The mentioned matrix can be formalised as follows.

$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cccc} k_1 & k_2 & \ldots & k_m \end{array} \\ \begin{pmatrix} y_{11} & y_{12} & \ldots & y_{1m} \\ y_{21} & y_{22} & \ldots & y_{2m} \\ y_{31} & y_{32} & \ldots & y_{3m} \\ y_{41} & y_{42} & \ldots & y_{4m} \end{pmatrix} \end{array} \quad (1)$$

When evaluating values of individual criteria, the *maximisation principle* is applied, i.e. the criteria are designed so that a particular variant is the best when the highest relevant criterion values are used. Calculation of the criterion values related to criteria *A* and *B* utilises a static *track occupation plan*, which is constructed for every major passenger railway station. Each running track within the station is linked with data (using the time step of one minute) about its occupation of trains.

Attention is further paid to an artificial neural network applying supervised learning (Nguyen et al. 2003), which represents one of the possible ways of solving the discussed track assignment problem. The mentioned neural network requires a prearrangement of two sets: a set of specific individual traffic situations (training patterns/inputs) and another set of relevant expert solutions. Produced outputs of a trained neural network are then compared with corresponding expected solutions – an expert/supervisor continuously evaluates the quality of the outputs and decides upon the next training steps.

Selection of an appropriate neural network type (e.g. feed-forward network, multi-layered perceptron etc.) represents an essential problem. It is quite difficult to determine the suitable kind of neural network (concerning a given problem) in advance. Thus, experiments with different kinds of networks and their diverse parameterisations were carried out. As a result of the experiments it was claimed that a two-layered perceptron produced the most encouraging results.

The above mentioned methodological approach can be divided into the following steps:

- Gaining knowledge about the platform track assignment problem.
- Specification of the calculation method applied to getting criteria (*A*–*D*) values.
- Computation of criterion matrices (exploiting criteria *A*–*D*) for different traffic situations.
- Separation of available data into disjoint sets (training set and test set).
- Supervised learning of selected neural network using data from the training set.
- Evaluation of the neural network behaviour in respect to input data from the test set.

The standing methodology expects that for each input station direction one trained neural network is available. It means in

fact that a two-layered perceptron has to be trained, tested and then applied to platform track assignments for arriving trains (for the given input direction) within the simulation model reflecting the system of a passenger station. The utilisation of the mentioned approach reached quite encouraging results – for the case study (reflecting Prague main station) the testing stage of the neural network reached a 95 % hit ratio (Bažant and Kavička 2009).

The classical approach based on the mathematical methods of multicriterial analysis is described and evaluated within the contribution (Kavička and Bažant 2009), whereas the artificial neural network achieved better results.

NEURAL NETWORK PARAMETERISATION

A multilayered perceptron network is composed of several layers containing neurons. The neurons from individual adjoining layers are mutually interconnected in such a way that they create a complete bipartite graph (fig. 2), i.e. an output of each neuron (located within a given layer) is distributed into inputs of all neurons situated within the consecutive layer. Inputs of the neural network are distributed into neurons of the first layer in the same manner. The last layer of the perceptron network is called *output layer*, other layers are denoted as *hidden layers*. The number of layers and the numbers of neurons within particular layers represent parameters of the network, the values of which depend on the character of a solved problem.

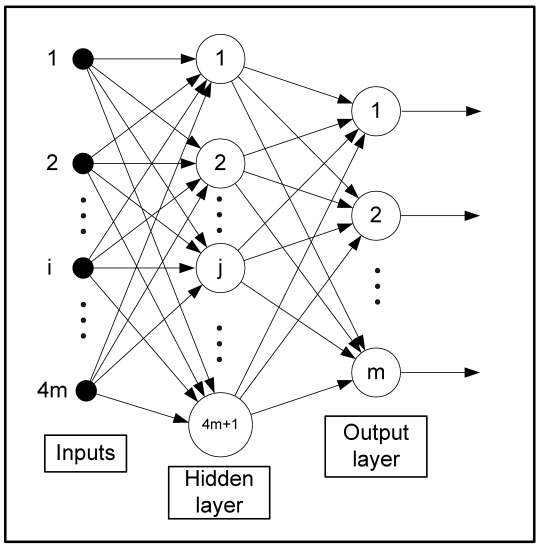


Figure 2: Simplified illustration of two-layered perceptron

Considering the studied problem the number of neural network inputs is derived from:

a) The number of elements involved in a set of admissible tracks (m) related to a given (potentially delayed) arriving train.
b) The number of criteria (four criteria are considered, A–D), according to which corresponding tracks are evaluated.

The number of inputs can be generally calculated as a multiplication of the tracks' number m (from a set of admissible tracks) and the number of applied criteria.

The next parameter reflects the number of neurons located within a hidden layer. If the number of neurons is too low the network is not able to follow all dependencies within the training data. On the other hand, a high number of neurons make the training stage more complicated (requiring application of more training patterns). In addition, the excessive number of training patterns can cause an insufficient ability of generalisation (i.e. the network does not produce satisfactory reactions to unseen input data). It is recommended to set the parameter to a slightly higher number than the number of inputs. Within the frame of a case study the number of neurons within the hidden layer was equal to $4m+1$.

It is essential to determine an appropriate number of neural network outputs in order to qualify the network behaviour especially during a process of learning. The mentioned point corresponds to a specification of a proper number of neurons within the output layer. In the case of a platform track assignment problem, the output layer disposes of m neurons (corresponding to a cardinality of a set of admissible tracks composed of candidating tracks for an admissible assignment).

The parameterisation of individual network layers as well as the methodology of network learning is described in (Bažant and Kavička 2009).

GENERALISATION OF THE NEURAL NETWORK

The reason for having several artificial neural networks (deciding about assignments of platform tracks) within simulation models of passenger station is mainly based on the fact related to the different numbers of inputs and outputs (of those networks) with regard to the different sets of admissible tracks. The number of the given network inputs and outputs directly depends on the number of elements (m) within the corresponding set of admissible tracks (as mentioned in the previous section).

For each set of admissible tracks one trained artificial neural network is available, which supports assignments of platforms tracks to the relevant arriving delayed trains. Let us present an example reflecting the constructions of several sets of admissible tracks (within Prague main station) reflecting different input/output line tracks (fig.3). The symbols Pv_1, Pv_2, Vs_1, Vs_2, Vs_3, Vs_5, Li_1, Li_2, Ho_1 and Ho_2 correspond to the input/output line tracks. For the separate directions reflecting the relevant input/output line tracks the following sets of admissible tracks can be defined: $S_{Pv1,Pv2} = \{ k_9, k_7, k_1, k_2, k_8, k_{14}, k_{20} \}$, $S_{Vs3,Vs5} = \{ k_9, k_7, k_1, k_2, k_8, k_{14}, k_{20}, k_{22}, k_{24}, k_{26} \}$, $S_{Vs1,Vs2} = \{ k_{14}, k_{20}, k_{22}, k_{24}, k_{26}, k_{28}, k_{30}, k_{32} \}$, $S_{Li1,Li2} = S_{Ho1,Ho2} = \{ k_9, k_7, k_1, k_2, k_8, k_{14}, k_{20}, k_{22}, k_{24}, k_{26}, k_{28}, k_{30}, k_{32} \}$.

The utilisation of several artificial neural networks does not seem to be very flexible from the viewpoint of their construction, maintenance and execution within the frame of a simulation model. Thus, to increase the flexibility of a simulation model reflecting the traffic within a passenger railway station we propose to construct generalised artificial neural network, which would be able to make decisions about platform track assignments for any combination of

input and output line track. The proposed and tested solution is based on the generalisation of criterion matrices in such a way that those matrices involves all station platform tracks - table 1 and table 2 demonstrate selected original criterion matrices for the set of admissible tracks $S_{Li1,\ Pv2}$. Table 3 and table 4 illustrate the selected generalised criterion matrices (the columns with zero values correspond to the tracks, which are not admissible for the relevant trains).

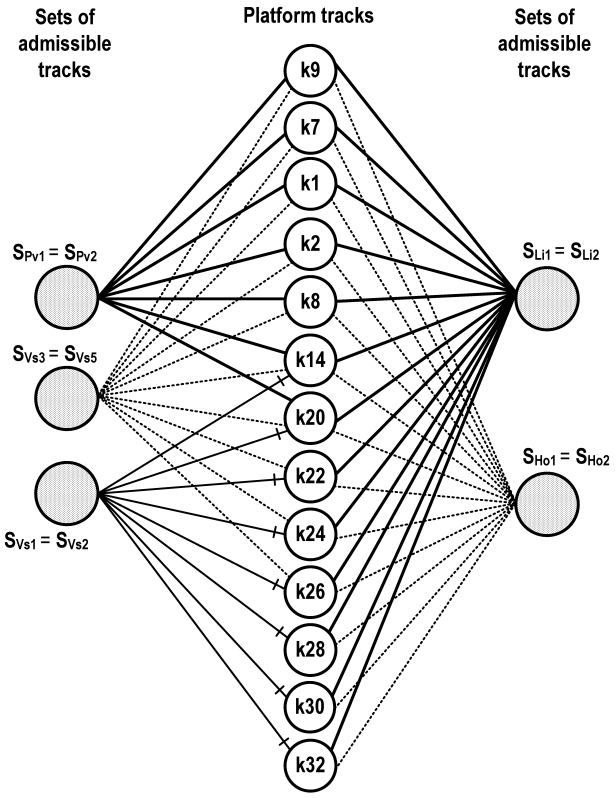


Figure 3: Schematic illustration of sets of admissible tracks

The tables involve one additional row (denoted as *NN*) containing output values produced by the neural network as a reaction to the input values (involved within the rows denoted as *A–D*). The output values express the suitability rates of the corresponding tracks (named within individual columns) in consideration of their potential assignment to the mentioned train. Thus, it is assigned the track with the highest value in *NN*-row (the corresponding track column is highlighted).

EXPERIMENTS

To make experiments with platform track assignments for delayed trains we decided to use simulation model of railway station Prague main station. The investigated trains come from the timetable valid for time period 2009/2010 as well as the station layout/infrastructure.

Platform tracks are divided into to two parts in reality. We made a little change in our model so that we merged two parts of track into one single track. We also did not take into account dead-end tracks that are used only for few trains during a day and these trains do not have conflicts with other trains involved in our model.

Table 1: Criterion matrix considering the train SC 72 delayed 39 minutes

	k9	k7	k1	k2	k8
A	0,27	0,87	0,87	0,60	0,53
B	1,00	1,00	1,00	1,00	1,00
C	0,00	0,00	0,00	0,00	0,00
D	0,90	1,00	0,70	0,75	0,95
NN	- 0,01	0,03	- 0,01	0,10	- 0,02
	k14	**k20**	**k22**	**k24**	**k26**
A	1,00	0,13	1,00	1,00	0,00
B	1,00	1,00	1,00	1,00	1,00
C	0,00	0,00	0,00	0,00	0,00
D	0,65	0,35	0,25	0,15	0,05
NN	**0,90**	0,00	0,02	0,00	- 0,01

Table 2: Criterion matrix considering the train R704 delayed 42 minutes

	k9	k7	k1	k2	k8	k14	k20
A	0,07	0,60	0,33	0,40	0,00	0,67	0,00
B	1,00	1,00	1,00	1,00	1,00	0,13	1,00
C	0,00	0,00	0,00	0,00	1,00	0,00	0,00
D	0,90	1,00	0,70	0,75	0,95	0,65	0,35
NN	0,00	**1,00**	- 0,01	0,01	0,02	0,00	- 0,01

Table 3: Generalised criterion matrix considering the train SC 72 delayed 39 minutes

	k9	k7	k1	k2	k8	k14	k20
A	0,27	0,87	0,87	0,60	0,53	1,00	0,13
B	1,00	1,00	1,00	1,00	1,00	1,00	1,00
C	0,00	0,00	0,00	0,00	0,00	0,00	0,00
D	0,90	1,00	0,70	0,75	0,95	0,65	0,35
NN	- 0,01	0,03	- 0,01	0,10	- 0,02	**0,90**	0,00
	k22	**k24**	**k26**	**k28**	**k30**	**k32**	
A	1,00	1,00	0,00	0,00	0,00	0,00	
B	1,00	1,00	1,00	0,00	0,00	0,00	
C	0,00	0,00	0,00	0,00	0,00	0,00	
D	0,25	0,15	0,05	0,00	0,00	0,00	
NN	0,02	0,00	- 0,01	0,00	0,00	- 0,01	

Table 4: Generalised criterion matrix considering the train R704 delayed 42 minutes

	k9	k7	k1	k2	k8	k14	k20
A	0,07	0,60	0,33	0,40	0,00	0,67	0,00
B	1,00	1,00	1,00	1,00	1,00	0,13	1,00
C	0,00	0,00	0,00	0,00	1,00	0,00	0,00
D	0,90	1,00	0,70	0,75	0,95	0,65	0,35
NN	0,00	**1,00**	- 0,01	0,01	0,02	0,00	- 0,01
	k22	**k24**	**k26**	**k28**	**k30**	**k32**	
A	0,00	0,00	0,00	0,00	0,00	0,00	
B	0,00	0,00	0,00	0,00	0,00	0,00	
C	0,00	0,00	0,00	0,00	0,00	0,00	
D	0,00	0,00	0,00	0,00	0,00	0,00	
NN	0,02	0,00	0,00	- 0,01	0,00	0,00	

To verify ability of artificial neural network generalisation we used the afternoon peak time that is between 5 p.m. and 8 p.m. and we decided to focus on long distance trains incoming from the input track S_{Li1} and outgoing through the output track S_{Pv2}.

For these two directions the following admissible platform track set is defined: $S_{Li1,\ Pv2} = \{k_9, k_7, k_1, k_2, k_8, k_{14}, k_{20}, k_{22}, k_{24}, k_{26}, k_{28}, k_{30}, k_{32}\}$. There are 9 long distance trains for the mentioned afternoon peak time, so we generated criterion matrices for every single train considering delay of the train in interval from 0 to 60 minutes. Using this approach we get 549 situations that were divided into two subsets:

- *training subset* (275 situations),
- *test subset* (273 situations).

Training subset is constructed on the basis of systematic sampling applied to the initial set (sampling period was set to 2 minutes). Test subset was constructed analogically; particular sampling was shifted in comparison with the previous subset (phase-shift was equal to 1 minute)
Neural network training was based on situations included in the training subset and the neural network was able to learn almost all solutions to considered situations (only one situation did not successfully passed the learning stage correctly but the neural network decided to choose the second best solution to the situation). Then we performed testing using test data and obtained the result depicted in Fig. 4. The testing stage reached a 93 % hit ratio.

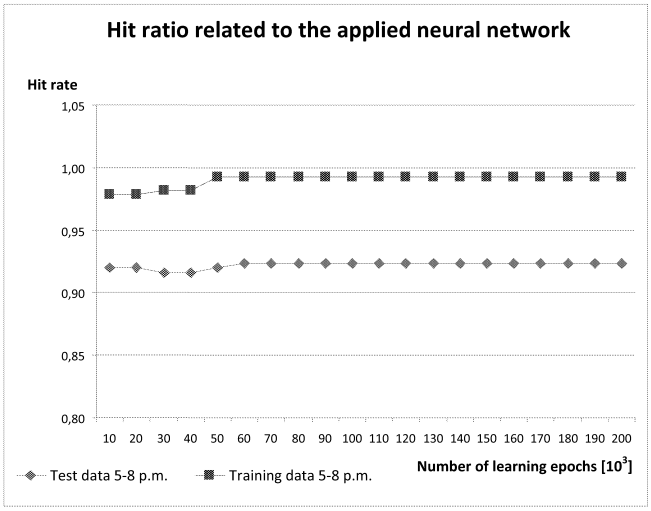


Figure 4: Hit rate of the applied neural network

CONCLUSIONS

The utilisation and testing of a generalised artificial neural network (as a decision-making support related to the platform track assignment problem within passenger railway station) reached quite encouraging results. It means if fact, that it can be used just one generalised network within relevant simulation model, which has a potential to be exploited for the trains approaching the station from an arbitrary input line track and departing through any output line track.

ACKNOWLEDGEMENT

This work has been supported by the National Research Program of the Czech Republic under project "MSM 0021627505 Theory of Transportation Systems".

REFERENCES

Adamko, N., Klima, V. 2008. "Optimisation of Railway Terminal Design and Operations using Villon Generic Simulation Model". *Transport*, No.4, 335-340.

Bažant, M., Kavička, A. 2009. "Artificial neural network as a support of platform track assignment within simulation models reflecting passenger railway stations". *Journal of Rail and Rapid Transit*, Vol.223, No.5, 505-515.

Chakroborty, P., Vikram, D. 2008. "Optimum assignment of trains to platforms under partial schedule compliance". *Transportation Research Part B: Methodological*, No. 2, 169-184.

Figueira, J., Greco, S., and Ehrgott, M. 2004. "Multiple Criteria Decision Analysis: State of the Art Surveys". Springer Verlag, New York (US).

Kavička, A., Bažant, M. 2007. "Simulations as a support for planning infrastructure within Prague Masaryk station". In *Proceedings of 21st European Conference on Modelling and Simulation* (Prague, Czech Rep., Jun. 4-6). European Council for Modelling and Simulation (UK), 363-367.

Kavička, A., Bažant, M. 2009. "*Design of a decision-making support within agent-based simulations reflecting railway traffic*". *In: Proceedings of European Simulation and Modelling Conference 2009 pp. 251-255,* Leicester, United Kingdom 2009. ISBN: 978-90-77381-52-6.

Kavička, A., Klima, V., Adamko, N. 2006. "Analysis and optimisation of railway nodes using simulation techniques". *Computers in Railways X.* WIT Press, Southampton (UK), 663-672.

Kavička, A., Klima,V., Adamko, N. 2007. "Simulations of transportation logistic systems utilizing agent-based architecture". *International Journal of Simulation Modelling*, No.1, 13-24.

Nguyen, H. et al. 2003. "A first course in fuzzy and neural control". CRC Press, Boca Raton, Fl.

SIMULATION-BASED DESIGN FOR INFRASTRUCTURE SYSTEM SIMULATION

Michele Fumarola
Delft University of Technology
email: m.fumarola@tudelft.nl

Yilin Huang
Delft University of Technology
email: y.huang@tudelft.nl

Çagri Tekinay
Delft University of Technology
email: c.tekinay@tudelft.nl

KEYWORDS
Simulation Library, DEVS, Infrastructure, Logistics

ABSTRACT

Simulation models are often used to analyze the behavior and performance of infrastructure systems. The use of simulation models in multi-actor design processes is restricted to the analysis phase after conceptual designs have been completed. To use simulation models throughout the design process, simulation environments need to be adapted to support the interactive and iterative nature of design processes, making the shift from a 'hard' to a 'soft' systems perspective. By using the Discrete Event Systems specification, modular and hierarchical component libraries can be constructed that can be used in participatory design processes. In this paper, we present our experiences on developing and using simulation environments in design processes for container terminals and rail systems. We present the cases, the architecture of the design environments and the use of them in actual design processes. Hereafter, we will discuss the enhancement we are studying to support the multiple perspectives present in multi-actor environments.

INTRODUCTION

The challenges in designing modern infrastructure systems are abundant. Multiple actors are involved that have to keep track of investments, technical feasibility, and, the nowadays increasingly important, societal and environmental impact. Modeling and simulation (M&S) can be used to support these multi-actor environments. However, constructing simulation models requires specialized skills and a lot of time and effort. This does not match well with the interactive and flexible character of a multi-actor design process. The use of M&S in a multi-actor design process has to adhere to certain requirements before it can be successfully employed.
Modern M&S environments stem from a 'hard systems' paradigm that is based on the conception that "there is a desired state, S(1), and a present state S(0), and alternative ways of getting from S(0) to S(1); 'problem solving,' according to this view, consists of defining S(1) and S(0) and selecting the best means of reducing the difference between them" (Checkland 1978). This paradigm suffers from a lack of recognition of different interests in a decision making process. According to Rosenhead and Mingers (2001) this means that in current simulation environments

- problem formulation is in terms of a single objective;
- there are overwhelming data demands;
- people are treated as passive objects;
- there is a single decision-maker with abstract objectives from which concrete actions can be deduced;
- and there is an attempt to abolish future uncertainty, pre-taking future decisions.

'Soft systems' (Checkland 1981) have been presented as a methodology that is closer to reality. It acknowledges multiple actors and mostly diverging opinions, and views the decision making process as a learning process wherein the various actors engage to understand the problem and each other's opinion.
Although the soft systems methodology is not intended to be used with simulation models, this could be a fruitful combination. Aughenbaugh and Paredis (2004) provide a very thorough and to-the-point explanation as to what simulation can bring to design and decision making:

> Without modeling and simulation, design relies on implicit knowledge. Implicit knowledge is unreliable in that designers do not know the assumptions and uncertainty in the knowledge explicitly. When decisions are coupled and require input from several experts, there is no way to make tradeoffs using only implicit knowledge about uncertainties.

Robinson (2001) reports on the use of simulation as a way of understanding the complexity of the problem and to facilitate the discussion among stakeholders. den Hengst et al. (2007) further elaborates on Robinson (2001) by discussing what can be done but what are still the limitations on using simulation models to foster decision making processes. The major shortcomings of modern simulation environments in multi-actor settings are

- the expertise needed to build simulation models;
- the complex code to build the simulation model, verify and validate it;
- the lack of knowledge of the actors concerning M&S results in a hard to accept model;
- and finally, running the experiment takes a long period (multiple hours) of time.

To use simulation models in design processes, it is important to shift from a hard systems perspective to a multimethodological approach as presented by Robinson (2001) and den Hengst et al. (2007). This shift sets specific requirements on the design environment and changes the use of simulation models: whereas now simulation models are purely used as an analysis tool after the conceptual design phase has been completed, it will become a discussion platform throughout the entire design process.

Designing multi-actor design environments

According to Simon (1996), design processes follow a path of first structuring the problem, followed by a formulation of alternative solutions based upon selected criteria and finally by a selection of the best alternative. To allow this, design environments need to provide sufficient flexibility in constructing the alternatives. This calls for a modular framework that supports the construction of alternative designs. den Hengst et al. (2007) reminds us that the complex code can be a burden: hiding this is necessary to use simulation models in design processes. A feasible approach to allow modularity and hide internal code would be to use component-based modeling. Indeed, components provide the major advantage of being self-contained, reusable, replaceable and customizable (Verbraeck 2004).
Hu et al. (2005) discuss the suitability of the Discrete Event Systems Specification (DEVS) (Zeigler and Praehofer 2000) for component based-modeling and simulation. To facilitate the easy development of simulation models using existing components, they suggest the use of variable structure in DEVS, which means dynamically adding and removing components and couplings. Three reasons are given for using variable structure in DEVS: (1) to model systems that exhibit structure and behavior changes, (2) to design and analyze a system under development, and (3) to be able to load only a subset of the system's component when simulating very large models. As the modularity and hierarchical structure of model components provided by the DEVS formalism could benefit model construction in general, recent work has been focused on developing the Event-Scheduling DEVS library (ESDEVS) (Seck and Verbraeck 2009). ESDEVS implements the parallel DEVS formalism on top of the DSOL library (Jacobs 2005). The ESDEVS is based on the event-scheduling worldview wherein executions of the internal transition function are scheduled according to the specified time advance function and unscheduled at the reception of external events. The confluent transition function handles the coincidence between internal and external events. Dynamic structure DEVS is also implemented in the ESDEVS library so that components and coupling relations can be added and removed dynamically during simulation runtime.

A component-based framework hides complex details of simulation models while allowing an easy way of constructing alternative designs. This fits perfectly the idea of having a design environment build for supporting an interactive and iterative multi-actor design process. A multi-actor design environment should support participatory design: an engineering design process is seldom performed by a single person or an actor. Many actors are involved and they all try to achieve their own objectives. A design process should support a certain convergence of interests of all actors.

Outline of the paper

In this introduction, we have motivated the need of requiring a shift in the use of simulation to support design processes in multi-actor environments. In our research, we are particularly interested in supporting the design of logistics systems. In the next section, we are going to present a case study involving the design of automated container terminals. For this case study, we have developed a simulation components library and accompanying tools to use this library in interactive design sessions. After this, we presentd a similar approach for the development of rail transport systems. In both case studies, we have studied how simulation models can be used interactively in the design process, instead of purely as an analysis tool after the conceptual design is concluded. Hereafter, we will discuss what else is needed in a design process to better support decision makers. We will present some concepts that are part of our future work and finally conclude the paper.

AUTOMATED CONTAINER TERMINAL OPERATIONS

Container terminals have become essential in today's supply chain of goods. There is a steady increase in the use of automated equipment to improve the effiency of loading and stacking containers in terminals. Equipment for stacking containers can be automated as well as equipment that is used for transporting the containers from the stacks to the quay cranes. Exploring alternative types of equipment and design options is a time consuming and challenging task. A simulation component library coupled to an AutoCAD design environment can fasten the assessment of design decisions.

Architecture of the design environment

The 3D visualization tool serves as a platform for communicating design decisions to all actors involved in the process. In Fumarola and Versteegt (2011) we have shown that 3D visualization environments enhance communication among actors by having an indisputable and understandable presentation platform. Technically less inclined people tend to prefer 3D visualizations above technical 2D CAD drawings. For this reason, the 3D visualization tool, which in diagram 1 is called 'Virtual Terminal', is an integral part of the design environment. The simulation components library has been developed in DEVS (Zeigler and Praehofer 2000) using DSOL ES-DEVS (Seck and Verbraeck 2009). It has been discussed extensively in Fumarola et al. (2010): it contains a set of components for each type of equipment in a common automated container terminal. The components library contains components for quay cranes, automated guided vehicles, and rail mounted gantries. To control the equipment, is has an implementation of terminal operating system containing algorithms for path finding and scheduling.
Both the visualization and simulation models are instantiated using an XML file that is generated based on CAD drawings. The XML contains the elements extracted from the CAD design that are important for the visualization and simulation model. The visualization tool uses a component library of 3D meshes to use the appropriate 3D mesh for the given equipment. Likewise, the simulation environment queries for the right simulation component using the right name and parameters. Using this structure, one single CAD drawing can be used to fully instantiate the visualization and simulation components. This is reflected in Figure 1.

Use of the design environment

The design environment is made to support the multiple actors present in the design process for automated container terminals. In Hu (2008) and Derksen (2009), whose work were part of the same case study discussed here, an analysis of the actors involved in the decision making process has been carried out. The major identified actors are involved with

- innovation as a way to deploy novel equipment to increase productivity;
- business as to keep costs and revenues in balance;
- engineering as to design a technically feasible terminal;
- and, finally, environment and safety.

The design process of container terminals can be partitioned roughly in five phases: project acquisition phase, global engineering phase, detailed engineering phase, implementation phase, and operation phase (Fumarola and Versteegt 2011). The design environment can support actors in the global engineering phase and the detailed engineering phase. In the global engineering phase, conceptual designs are made that are based on best practices and rules of thumb. Simulation is mostly not used in this phase because of the quick and highly iterative character of this phase. The design environment discussed here is highly suitable for the global engineering phase. It can be used to quickly assess different designs using its easy to use interface. The same models can later be used in the detailed design phase, which serves to study in depth the behavior of the system using extensive scenarios.

RAIL SIMULATION LIBRARY

Modern rail transport planning and design are complex and time-consuming. Many stakeholders such as the transport authorities and operators are involved in decision making, each to their own objectives and responsibilities. To design and develop adequate simulation tools that support these domain-experts in the decision making process is a challenge. Different aspects of the railway system, e.g. the infrastructure, control measures and timetables, shall be combined in a self-contained simulation package to allow for comprehensive experimental analysis by multiple actors, and to provide them with a platform that enhances common understanding to reach well-informed decisions. Due to the long life span of rail infrastructure and services, changes often come up which lead to new issues to study. Thus rail-bounded modeling particularly requires malleable composition and configuration in order to promote model reusability. The design requirements mentioned earlier embrace these needs, and therefore they are used as the guidelines to design our rail simulation library.

Rail Model Hierarchy

Similar to the first case, rail models in the library are defined following the DEVS formalism (Zeigler and Praehofer 2000), which benefits the model structure with modularity and hierarchical composition (or decomposition). The library is built as an extension of the ES-DEVS library (Seck and Verbraeck 2009). Two classes of models are distinguished, the *rail element*s and the *rail component*s, as shown in Fig. 2, which are in line with the concept of atomic and coupled models in DEVS. Rail elements are the irreducible models that can not be decomposed, e.g. a rail vehicle, sensor, signal or a piece of track segment; more see (Huang et al. 2010). Rail components are resultants of composition that may contain rail elements and rail components. The rail elements and components can be composed into more complex components, which in turn can be composed, and so further recursively. As such, users can build complex

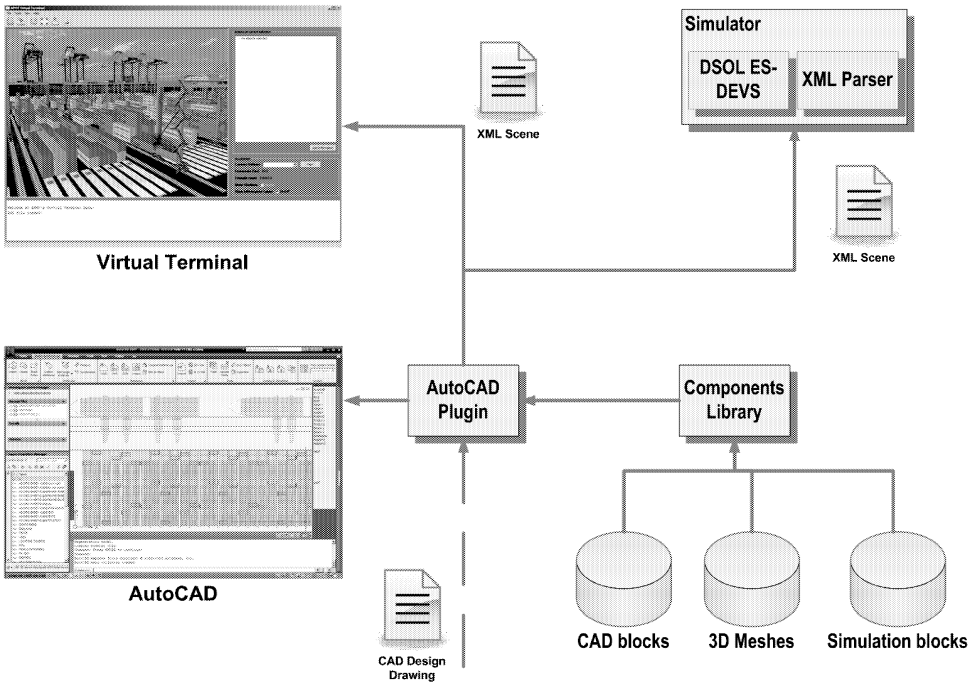


Figure 1: Architecture of the automated container terminal design environment

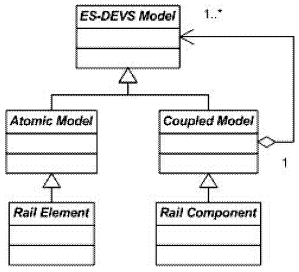


Figure 2: (Simplified) Model Composite Relation

rail network with the components without knowing the complex internal code.

Fig. 3 illustrates a simple example of the rail network model hierarchy. A cloud shape denotes a rail component, a rectangle denotes a rail element, and a one-to-more relation is represented by a triple-line. In general, a top level rail model (component) contains one or more sources and sinks, which respectively generates and removes vehicles in the simulation. (Each vehicle generator in a source can schedule the vehicle generation according to a different timetable.) The rail model composition is flexible. It may contain stops, intersections, block systems, and other model composition. The intersection, e.g., has signals to control the accessibility of the track sections. Switches allow vehicles to move from one track over another. The control unit computes the signalling logic depending on the occupancy of the tracks and switches. Each of those rail elements models one functionality of the rail infrastructure, and the their aggregation forms a higher level component that

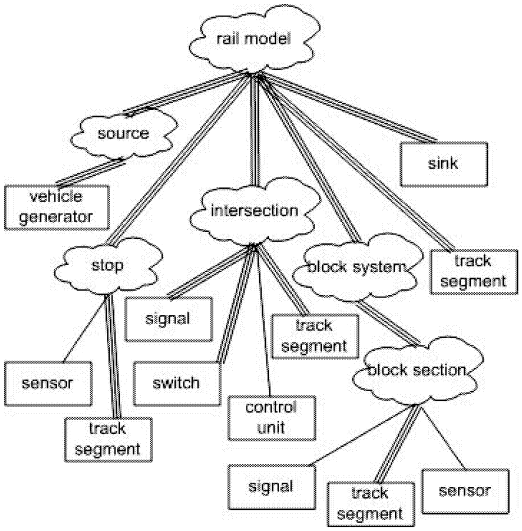


Figure 3: An Example of the Rail Model Hierarchy

performs more complex tasks.

A rail vehicle (model element) is generated in a source. As the vehicle drives in the rail network, it is moved from one rail component to another (dynamic structure DEVS). The rail network model, at its lowest description level, is a directed graph of linked rail infrastructure elements (track segments, sensors, switches and signals). A vehicle model is linked directly with an infrastructure element, each of which is capable of *message propagation*, the principle object-to-object communication mechanism used in the library. Its base concept is the DEVS message passing through paired I/O ports.

Message Propagation

The message propagation can be along the traffic current or in the opposite direction. When a vehicle lacks information about its next infrastructure or the preceding vehicle, it sends a *request-message* forward. The infrastructure element that gets the message propagates the message until the next infrastructure of interest and/or a preceding vehicle is found. The found element (e.g., a track segment that has speed limit change, a signal or a vehicle) replies with a *response-message* which is propagated backward until it reaches the original sender of the request-message. If an element changes state, it also sends an *update-message* backward to inform its potential succeeding vehicle. A message contains information of the sender, the contemplated receiver (if necessary), and the distance between the sender and the receiver. According to this information, a vehicle's movement is computed until its next infrastructure or its preceding vehicle (if any). Once the vehicle reaches the next infrastructure (it will be linked to the next infrastructure) or its preceding vehicle, a new round of request and response message exchange is performed and new movement is computed. Message propagation is a communication mechanism that is transparent to model users. The users only need to setup the rail network. The message sending, forwarding, receiving and the dynamic linking of the vehicles are accomplished by the model elements autonomously.

Rail Model Builder

To simplify model construction and configuration, we have developed the CAD Rail Model Builder (CRMB) as one of the possible approaches to interface with the rail library. Many railway companies and authorities use CAD and GIS applications for infrastructure design. The resulting design files can be used for model generation. However, as these files often do not contain information about the network topology in a hierarchical way, a couple steps of data cleaning, preprocessing and transformation are necessary to make the data useful for DEVS model generation. For example, when the orientation of the track drawing entities is not in accord with the traffic current, the track orientation needs to be corrected first. The positions of switches and crossovers can be detected where more than two track segments connect. Based on the position and orientation of the connecting tracks, the switch type is determined whether it is converging or diverging. In order to identify certain model components, pattern recognition technics are applied. The identified elements, e.g. tracks, switches and crossovers, that belong to one component are grouped and indexed for model generation. Some infrastructure components do not have a determinate pattern of its composite elements. These elements are defined by users in additional data sources (e.g., a configuration file) for model generation.

Enhancements of the library

The rail simulation library is still in development. We plan to separate the rail components into light-rail and heavy-rail specific packages. Both based on the generic rail simulation core. Such restructuring could reduce the library core to a minimum. A graphical modeling interface is also in design to provide a model configuration alternative for users. In a recent paper (Huang and Verbraeck 2009), we proposed a dynamic data-driven approach for rail simulation. The idea is to automate model calibration and validation by comparing model output with rail operation data. In this context, the DEVS based library design would also benefit our development.

FUTURE WORK

In the light of the first and second case studies, we show that DEVS model libraries can provide a feasible framework for modeling infrastructure systems. Supporting component-based modeling, developed DEVS libraries offer modularity to the designer to customize and reuse the model components. Yet, the conducted case studies reveal that several issues need to be addressed to fully support decision making in infrastructure systems.
The design process of infrastructure systems is multi-actor by nature in which every actor has its own interests and perceives the system in his own way. Therefore, the models should support different perspectives from various actors. What is more, each unique perspective can be decomposed into various levels of abstraction, so called resolutions. This will allows the actors to change the resolution of their models to specify the inputs at different levels of precision, analyze the output and reason about the cause-effect relationships. However, the existing designs lack a common consistent framework resulting in a single perspective and statically defined models at different resolutions. A framework with a set of rules, design guidelines and constraints need to be introduced in the existing design environment to allow a problem to be studied inside the whole assumptions space, instead of having a base case and trying to have some intuitions from that. This is important when dealing with the design of infrastructure systems. In a recent paper (Tekinay et al. 2010), we discussed the key issues and preliminary design ideas to provide a multi-resolution and multi-perspective modeling in multi-actor environments.

CONCLUSIONS

The purpose of this paper is to present our experiences when designing and using the DEVS model libraries to support decision making in infrastructure systems. Two

different model libraries including the library for automated container terminals and light-railway systems is given a baseline case to discuss the challenges throughout the design process. DEVS model libraries allow designer to build component-based models which have the advantage of being modular, extensible and reusable. Such capabilities provided by DEVS model libraries are arguably essential for designing infrastructure systems where vast number of components and control mechanisms involved. For the further studies, DEVS model libraries will be a solid backbone for us to deal with the next challenges of providing multiple perspectives at different levels of abstractions to support multiple actors.

REFERENCES

Aughenbaugh J. and Paredis C., 2004. *The Role and Limitations of Modeling and Simulation in Systems Design.* In *ASME 2004 International Mechanical Engineering Congress and Exposition.* American Society Of Mechanical Engineers, 13–22.

Checkland P., 1978. *The origins and nature of hard systems thinking. Journal of applied systems analysis*, 5, 99–110.

Checkland P., 1981. *Systems thinking, systems practice.* John Wiley & Sons.

den Hengst M.; de Vreede G.; and Maghnouji R., 2007. *Using soft OR principles for collaborative simulation: a case study in the Dutch airline industry. Journal of the Operational Research Society*, 58, 669–682.

Derksen T., 2009. *Modelling Modes of Operation: A DSS and a decision making process on the selection of the mode of operation for APM Terminals.* Master's thesis, Delft University of Technology, the Netherlands.

Fumarola M.; Seck M.; and Verbraeck A., 2010. *A DEVS component library for simulation-based design of automated container terminals.* In *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques.* Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering.

Fumarola M. and Versteegt C., 2011. *Supporting automated container terminal design processes with 3D virtual environments.* In H. Yand and S. Yuen (Eds.), *Handbook of Research on Practices and Outcomes in Virtual Worlds and Environment*, IGI Global.

Hu H., 2008. *Choosing the Optimal Mode of Operation for Marine Container Terminals.* Master's thesis, Delft University of Technology, the Netherlands.

Hu X.; Hu X.; Zeigler B.; and Mittal S., 2005. *Variable Structure in DEVS Component-based Modeling and Simulation. Simulation*, 81, no. 2, 91–102.

Huang Y.; Seck M.D.; and Verbraeck A., 2010. *LIBROS-II: Railway Modelling with DEVS.* In B. Johansson; S. Jain; J. Montoya-Torres; J. Hugan; and E. Yücesan (Eds.), *Proceedings of the 2010 Winter Simulation Conference.* IEEE. To be appear.

Huang Y. and Verbraeck A., 2009. *A Dynamic Data-Driven Approach For Rail Transport System Simulation.* In M.D. Rossetti; R.R. Hill; B. Johansson; A. Dunkin; and R.G. Ingalls (Eds.), *Proceedings of the 2009 Winter Simulation Conference.* IEEE, 2553–2562.

Jacobs P., 2005. *The DSOL simulation suite - Enabling multi-formalism simulation in a distributed context.* Ph.D. thesis, Delft University of Technology.

Robinson S., 2001. *Soft with a Hard Centre: Discrete-Event Simulation in Facilitation. The Journal of the Operational Research Society*, 52, 905–915.

Rosenhead J. and Mingers J., 2001. *Rational Analysis for a Problematic World Revisited: Problem Structuring Methods for Complexity, Uncertainty and Conflict.* John Wiley & Sons.

Seck M.D. and Verbraeck A., 2009. *DEVS in DSOL: Adding DEVS operational semantics to a generic Event-Scheduling Simulation Environment.* In *Proceedings of the 2009 Summer Computer Simulation Conference.*

Simon H., 1996. *The Sciences of the Artificial.* The MIT Press, third ed.

Tekinay C.; Seck M.D.; Fumarola M.; and Verbraeck A., 2010. *A Context-Based Multiresolution Multiperspective Modeling Framework.* In B. Johansson; S. Jain; J. Montoya-Torres; J. Hugan; and E. Yücesan (Eds.), *Proceedings of the 2010 Winter Simulation Conference.* IEEE. To be appear.

Verbraeck A., 2004. *Component-based Distributed Simulations. The Way Forward?* In *Proceedings of the 18th Workshop on Parallel and Distributed Computer Simulation.* IEEE Computer Society Press, Los Alamitos, CA, 141–148.

Zeigler B.P. and Praehofer H., 2000. *Theory of Modeling and Simulation.* Academic Press.

The Added Value of Simulation during Liquid Bulk Terminal Design

R. van Duijn, H.P.M. Veeke, T.N. Brans*, G. Lodewijks

*Department of Marine and Transport Technology, Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, Delft 2628 CD, the Netherlands * Royal Haskoning, Industrial Installations, P.O. Box 8520, 3009 AM, Rotterdam*

Keywords: Simulation Integrated Design, Model acceptance, Performance Analysis, Discrete simulation

Abstract

During the concept design phase of a liquid bulk terminal, questions often arise which cannot be easily answered with general design methods. In order to be able to make a better prognosis on terminal requirements and performance, there is a demand for a flexible all-round simulation tool that will enhance, refine and accelerate the concept design process of a liquid bulk terminal. Incorporating simulation into the design process should provide engineers and designers with additional insight into how design considerations will affect terminal performance. This paper shows a method to develop a design support simulation tool which can be used during function design and in the early stages of process design, using the Delft Systems Approach. The tool provides more insight on the effect the various design and operation considerations have on terminal performance. The tool presents instant feedback on terminal performance, based on variable designers' input on terminal design characteristics and logistic flows. The research question is to what extent a simulation tool has added value to the design process of a liquid bulk terminal

1. Introduction

When the decision process is started whether a liquid bulk terminal is needed (or needs to be upgraded) at a certain location, feasibility studies are performed at a strategic level. At this phase, general functionality and performance requirements are determined, based on experience developed throughout the years within engineering departments. At the start of the process design phase, the main objective is to define the required performance and related components needed to fulfil the client's requirements. The whole process would be enhanced if designers could gain an early insight in basic terminal performance capabilities based on demands and possibilities without the need to perform many individual calculations. Furthermore, there are some design aspects which cannot be easily evaluated using conventional methods. The question is if this could change with the help of a simulation tool.

Other liquid bulk studies in literature that are found to be focussing on some part of the liquid bulk logistic chain either have a very broad scope, considering the whole supply chain [Neiro, Pinto, 2004], a pipeline [Relvas, Matos et al., 2006] or on a jetty [van Asperen, Polman et al., 2003] rather than on a terminal. Also, the objective is often to minimize the deviation between planned and scheduled throughputs. Their perspective lies with scheduling activities in existing facilities, while the perspective of this research lies with upgrading existing sites or delivering proposals for new terminals.

A liquid bulk terminal is an industrial facility offering a total package of activities and services to handle, store and control liquid bulk (e.g. oil, gas and chemicals) to and from transportation modes (ships, trucks, train, pipeline) with a balance in handling and services to the transportation modes against minimized costs[Rijsenbrij, 2008]. Liquid bulk terminals can be divided in three major groups: the buffer storage terminal, the independent storage terminal and the trading terminal. Table 1.1 gives a summary of their contrasting characteristics. Because of the wide variety between terminals, terminals can not be considered standard off-the-shelf facilities. Each terminal requires a tailor made engineering solution.

***Table 1.1**, types of liquid bulk terminal*

Type	Use	Tank turnover rate [1/yr]	Fitpurpose for
Buffer storage terminal	Production plants	high	high
Independent storage terminal	Misc. mid-long term (strategic) storage	low	medium
Trading terminal	Short term storage	very high	low

The tank turnover rate shows the annual average number of fully filling a tank, fit for purpose indicates whether the terminal should be designed for a specific application (low flexibility rate), or suitable for a broader scope of products and services (high flexibility rate).

Within the supply chain, a liquid bulk terminal has several functions, all under the precondition of safety and environmental protection. These are:

1) Primary functions:

. • To connect different modalities, in order to maintain the logistic flow in the supply chain;

. • To store products temporarily; The main reason to store is to provide a buffer between modalities, to compensate for scheduling differences in supply and demand, or for strategic reasons.

• To change the product flow size and behaviour. Combining, distributing, and connection to networks.

2) Secondary functions:

. • To perform value added logistics. Some terminals add various petroleum products together to form a blend, e.g. gasoline. In this way, the terminal performs a value-adding transformation on the product flow.

2. Terminal design

In Figure 2.1 and Figure 2.2, the trajectory of a typical terminal design process is shown. The red balloon marked “Simulation” points out the areas where simulation can be a useful support tool.

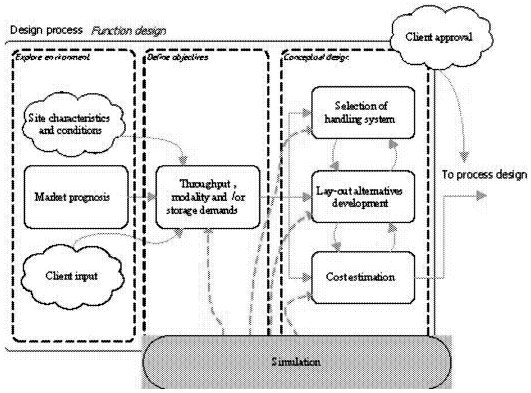


***Figure 2.1**, simulation during function design*

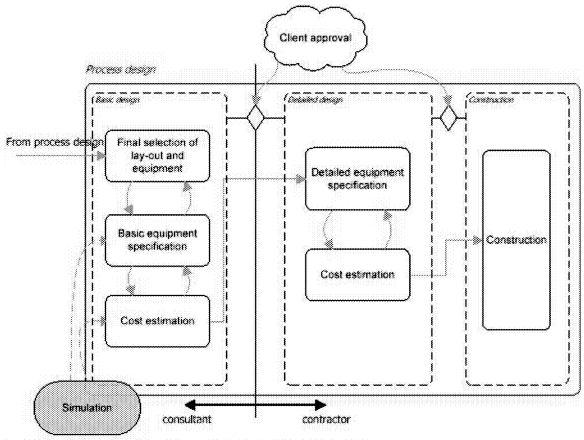


***Figure 2.2**, simulation during process design*

The balloon stops at the early stages of process design. This is where the application of simulation in this research differs from more traditional simulation: traditional simulation is a much more stringent method of delivering a one-off solution to a detailed problem during more detailed design stages. Here, simulation is not to be thought of as a means to instantly increase performance of any design. Instead, it should be regarded as a means to provide more insight in dynamic behaviour which could not be seen otherwise. This increased insight raises more questions regarding the terminal design which otherwise could have been left unnoticed. This leads to a better understanding of what can be expected at a later stage. In both function and process design, a simulation model can be applied in conjunction with the conceptual system model to make flow sizes inside the system apparent and clearly visible (quantifying flows). The total process of integrating simulation into a design process can be recognized as an application of the Simulation Integrated Design (SID) approach [Veeke, 2003].

A conceptual model, using the Process Performance (PROPER) model principles [Veeke, Ottjes, Lodewijks, 2008], is used as a framework on which the simulation will be built (Figure 2.3 and Figure 2.4).

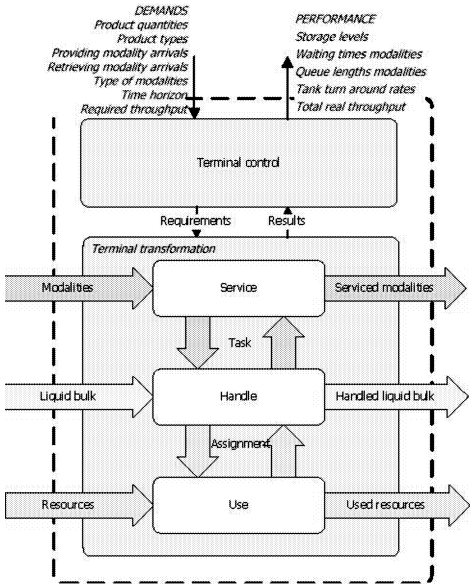


Figure 2.3, *PROPER model of liquid bulk terminal, aggregation layer 1*

In this model, the modalities, products and resources present in the system are regarded as separate but interacting flows. Figure 2.3 shows the terminal activities at the highest aggregation layer. A set demands is translated to requirements by the 'control' function, and similarly, the results of the transformation functions are translated to a performance, perceivable by the environment. Figure 2.4 displays a more detailed aggregation layer. This layer provides sufficient detail to determine the structure of the simulation model. The use of the PROPER model grants access to the use of process description language (PDL). Defining a good PDL eases the implementation of the model in a software environment, and it allows engineers to communicate transparently on the subject, especially if they are not directly involved in developing the simulation model.

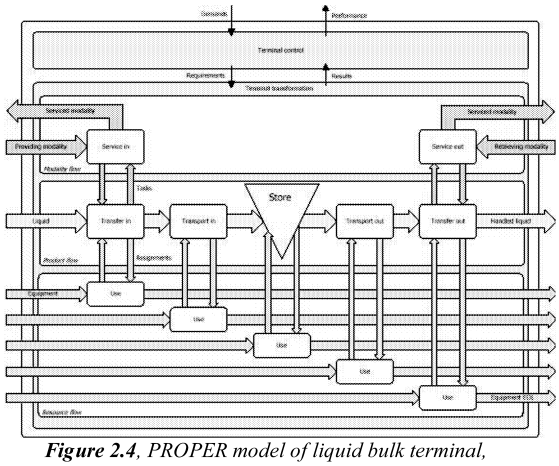


Figure 2.4*, PROPER model of liquid bulk terminal, aggregation layer 2*

3. Simulation model

The simulation model is developed in the programming language Delphi with the discrete-event simulation plug-in TOMAS (Tool for Object-oriented Modelling And Simulation) (www.tomasweb.com). Previously, the function and process design steps have been discussed and compared to the design steps at the company. To keep in line with this distinction, it is chosen to translate these design steps into separate phases of the simulation model, namely a dimensioning and control phase. The first phase will be purely to get an insight in size, dimensions and characteristics of the terminal. Control routines are left out of the scope in this phase. This model corresponds to the function design phase of the design trajectory. An important feature of this model is the absence of process equipment to support the terminal functions. Only the 'providing' and 'retrieving' modalities are specified as physical equipment; the terminal itself can be regarded as a series of functions without the constraints imposed by selecting equipment. The second model will use the output from the first model, e.g. storage levels and other specifications, to determine the equipment needed to satisfy the requirements. This model corresponds to the process design phase of the design trajectory. Because of the imposed restrictions that arise from selected objects from the equipment aspect (e.g. limited storage, limited quays), this phase introduces the need for control: the processes need to be steered into the right direction in order to keep the model running as intended.

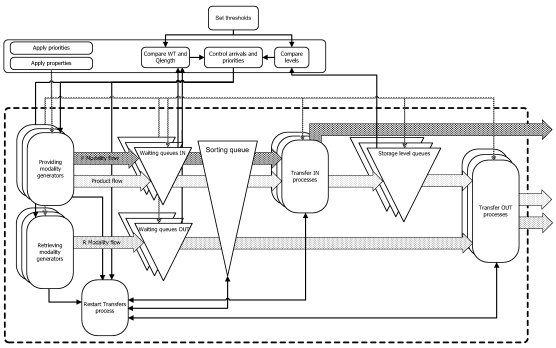


***Figure 3.1**, representation of simulation model contents*

Figure 3.1 shows the simulation elements present in the model. A global PDL is stated below:

- When the designer has provided properties to all elements via the graphical user interface (GUI) and information about the simulation run-time, the simulation is first initialized and then started;
- Providing modality generators;

Generators will provide the system with providing modalities and products. These modalities are created along with carried products and enter their respective waiting queues.

- Retrieving modality generators;

On the output side, retrieving modalities are generated. They are created with a product request. These modalities also enter waiting queues, and will be serviced by suitable Transfer OUT processes.

- Transfer IN processes;

When a transfer IN process is ready and able to service a suitable modality present in a corresponding queue, the modality is moved from the queue to the transfer process. The transfer processes take care of the actual transfer of product from the modality to the storage. After servicing, the product quantity is added to its storage level. The modality leaves the process and is removed from the system.

- Transfer OUT processes;

When a transfer OUT process is ready and able to service a suitable modality present in a corresponding queue, the modality is moved from the queue to the transfer process. When a modality enters an output transfer area, the product request of the modality initiates transfer from the storage to the modality. After servicing, the modality leaves the system.

- Restart Transfers process;

The generators and transfer processes each start a special process at the end of their routine. This process is called Restart Transfers Process.

Restart Transfers determines on the basis of compatibility between modalities, products and available docking areas, the next Transfer process(es) to be resumed.

Next to the main simulation elements, there are also elements that control the arrival processes and the priority of the incoming modalities.

4. Experiments

To examine the added value of the model during the design process, a simulation experiment has been conducted, using an actual project carried out by Royal Haskoning (International Engineering Consultants' firm).

One of Royal Haskoning's tasks within the project was to provide a complete functional design of the new storage area, jetties and quays. The first phase of the design includes a logistic study, including the allocation of products to tanks and to berths. During this phase, the simulation model was able to provide more information about the logistic behaviour, and thus aid in decision-making. The terminal under consideration is a typical trade terminal, as discussed in the introduction: a terminal with a very high tank turn around rate and berths that are utilized to the maximum.

During the dimensioning phase of the simulation, recommended storage sizes were obtained which were comparable to the storage sizes as statically calculated from a turnover requirement.

The dimensioning phase was also used to determine a vapour treatment unit capacity more precisely than was possible using conventional methods.

During the control phase, the simulation model shows how modalities are spread over the available berths as a result; this is more convenient than the conventional occupancy spreading method. The simulations show that in the design the availability of berths is sufficient; the allowable occupancies are not exceeded, and long waiting times were not caused by unavailable berths. Another outcome of the simulation runs was the presence of long waiting times for vessel carrying certain products; this was caused by service delay due to too small storage capacities compared to the modality capacities.

It is assumed that terminal operators in real life are capable of better planning than the control methods of the model are able to simulate at this

time. Still, it would require careful planning of modality arrivals.
Simulation runs with the recommended storage sizes from the dimensioning phase show vessel waiting times that are more evenly spread; the model predicts better performance when the available storage is considerably larger than the largest modality.

5. Conclusions

The added value of simulation during liquid bulk terminal design is:

The close integration of simulation from the beginning of the concept design process provides a better understanding and raises questions about the system to be designed that could otherwise have been left unnoticed.
The insight in peak behaviour, variable development through time and stochastic influences provide added knowledge about the system, and aid in decision-making. Furthermore, the development of a simulation model facilitates designers to think thoroughly about the system's processes, and therefore, it is this development process that generates questions that lead to a better understanding of the system. It is very beneficial if right answers are found to the right questions that have been raised by model-making.

The generic design of the conceptual model, and consequently, the simulation model, adds to the strength of the model: it enables the designer team to start of with the developed model right away, regardless of product types and facilities on site, types of modalities, import or export.
The use of the PROPER model as a conceptual model and consequently, the use of PDL to build the model, enables a swift and precise translation to a programming language. The PDL makes sure that designers, regardless of discipline, can understand the model's processes and provide constructive feedback.

The model in itself gives useful extra information which could not have been seen otherwise, in such an early stage. Experiments show that the simulation model is capable of producing useful additional information to the designers.
To answer specific questions, it is possible to expand the model with additional modules. Adding a new module is easily done due to the object-oriented nature of the model.
With each new more specific question, the model expands and the expansions will remain available for future projects, increasing the diversity and value of the model.

References

Asperen, E. van, Dekker, R., Polman, M., Swaan Arons, H. de, Waltman, L., 2003, "Arrival processes for vessels in a port simulation", Erasmus Research Institute of Management, ERS-2003-067-LIS

Duijn, R. van, 2009, 'The Added Value of Simulation during Liquid Bulk Terminal Design', TU Delft, faculty of 3mE, report nr. 2009.PEL.7382

Neiro, S.Mm.S., Pinto, J.M., 2004, "A general framework for the operational planning of petroleum supply chains", Computers and Chemical Engineering 28, pp. 871-896

Relvas, S., Matos, H.A., Barbosa-Póvoa, A.P.F.D., Fialho, J., Pinheiro, A.S., 2006, Pipeline Scheduling and Inventory Management of a Multiproduct Distribution Oil System", Ind. Eng. Chem. Res. 45, pp. 7841-7855

Rijsenbrij, J.C., 2008, "Introduction to Transport Engineering and Logistics", Delft University of Technology, fac. 3mE, Lecture notes WB3420

TOMAS, Tool for Object oriented Modelling And Simulation, www.tomasweb.com.

Veeke, H.P.M., 2003, 'Simulation Integrated Design for Logistics', DUP science, ISBN 90-407-2417-2

Veeke, H.P.M., Ottjes, J.A., Lodewijks G., 2008, 'The Delft Systems Approach: Analysis and Design of Industrial Systems', Springer-Verlag London Limited, ISBN 978-1-84800-176-3

AGVs in a production environment – A flexible and modular transport system for production

E.E. van Leeuwen
H.P.M. Veeke
R. van der Stappen *
G. Lodewijks

Faculty of Mechanical Engineering, Maritime and Materials Engineering
Delft University of Technology, Mekelweg 2, 2628 CD Delft
* Vector Aandrijftechniek, Industrieweg 175, 3044 AS ROTTERDAM

Keywords: AGV, Production environment

ABSTRACT

In the production environment of the new building of Vector Aandrijftechniek BV an AGV (Automatic Guided Vehicle) system is desired to increase flexibility and efficiency of internal transport of parts and products. The current transportation systems should be replaced. First the current production process is investigated and data is collected of the production times and quantities of last year. The data and investigations lead to three qualified routes between the processes which could be replaced by AGVs. The concepts for these routes are based on the specifications of multiple AGV suppliers and the results from the investigation. Also the dimensions, entrances and exits of the new building, quantities and material flows are used. The concepts differ in the separation or combination of the routes between the processes and the number of handlings which are done by AGVs or other systems. A simulation model is made of the concepts and the production processes between the transports, which could provide the information about the required number and performance of the AGV.

The results of the simulation show large differences in the required quantities of AGVs, reliability and buffer sizes between the concepts. For a production of 70.000 drives per year (including peaks of 100.000 drives) nine AGVs are required on the routes between the storage and the packing station. With a modular design of the AGVs it is possible to use the same chassis on all routes and only the (un)loading device would differ.

For the new building it is recommended to keep the AGV system as simple as possible and to test an AGV in the current production process for experience and to maximize the efficiency.

INTRODUCTION PROJECT AGV CONTROL

1.1 EURODRIVE PROJECT

Vector Aandrijftechniek BV is a company which produces electromechanical drives and is a subsidiary of SEW-Eurodrive. In 2007 a project has started to move to a new building and to become a Model Eurodrive.[1] With this movement the opportunity rises to further optimize the process and therefore a project is initialized for the development of this process. Within this project different topics will be addressed such as an electronic data medium and a painting robot. One of the main subjects of the reorganization considers the internal logistics. Production islands are configured for an optimized assembly process of the drives, but the storage is divided throughout multiple decentral storages in the production environment, which causes a lot of movement to fill up these storages. For the new building a layout is desired with a minimum number of transports and a higher grade of automation for a reduction of costs and to improve the controllability and reliability.

1.2 AGV Project

The incoming goods from the suppliers are sorted and divided by manual transport with a forklift or cart into a centralized storage for the motor parts and a decentralized storage near the assembly islands for the gearbox parts. From these storages the parts will be picked on order and transported by cart to the assembly stations who are specialized for the gearbox. The assembled drives are transported by a conveyor belt to the oil filling and testing station. Further transport to the packing station is done by a rail system through the intervening processes.

[1] This research was executed during 2008 and 2009. In 2009 the complete project for the construction of a Model Eurodrive was stopped as a result of the financial crisis.

The first concept of the new building has a material flow as shown in figure 1. In the middle the centralized storage,

around the centralized storage the decentralized storages with the islands and at the upper left the oil fill and test station. At the top of the layout are the paint, dry and heat cabin and at the upper right the packing station is shown. The production in 2008 was about 57000 drives per year, but the new design will be determined for an amount of an average of 70000 drives per year with a maximum of 100000 drives per year.
The idea for the concept of the new building is to replace the transport systems by AGVs for a more flexible system, which is the same on all the routes. The number of transports between all the processes is around 40 drives per hour.

1.3 Goal of this research

For the new production plant the main objective is to increase the efficiency and the flexibility of the transport processes. One of the research objectives is the replacement of current transport processes with an AGV, Automatic Guided Vehicle, which is a proven transport technology and has a high degree of flexibility. Because Vector has not enough knowledge of AGVs, they set up a project to investigate the possibilities of using an AGV in the new production plant. Therefore the following research question is stated:

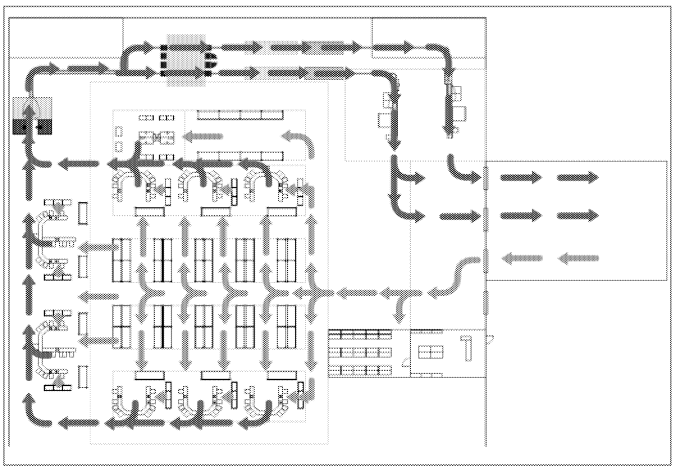


Figure 1 Concept of the new building

What should be the required physical AGV-system for replacing the transport processes in the main production process which will fit in the current layout of the new facility?

To answer this question, a number of sub-questions can be asked:

- *What is the current set up of the production process?*
- *Which transport processes in the main production process could be improved?*
- *What should be the temporary technical specifications of the AGV system?*
    - *Quantities, speed, buffer places, reliability*
- *What is to be expected if the new transport processes are used in a simulation?*
- *What should be the technical specifications of the AGV system?*
- *Which information transfers are necessary?*
- *What will be the physical translation of the technical specifications?*

The main goal of the project can be described as follows:
Research the possibilities for replacement of the internal transportation processes in the main production process with

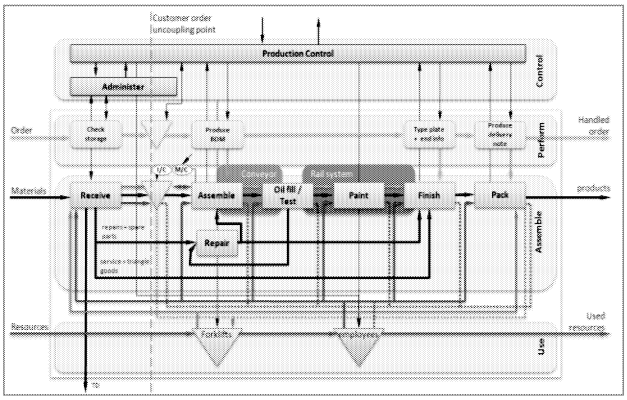


Figure 2 Model of the current production process

AGVs in the current layout of the new production plant. Substantiate the choices for the proposed solutions and elaborate it until it is a concept design of a physical AGV-system. This design must contain the technical and operational requirements of the AGV-system, information transfers between the AGV-system and the environment and a simulation of the physical AGV-system.

The next section contains the investigation of the current processes and the concepts of AGV routes will be introduced and compared in the third section. In the fourth section more details about the simulation model are given and the working principle explained. The results of the simulation experiments will be explained in section and conclusions
and recommendations can be found in the section 6.

INVESTIGATION OF THE CURRENT PROCESS

Figure 2 shows the current production process with the material, order and resources flows according the Delft Systems Approach [Veeke,Ottjes et al., 2008]. Physical materials are received (at cargo supply) and are split in repairs, service goods, triangle goods and materials for the production. Service goods can be end products from SEW or other suppliers. Repair parts will be repaired at the mechanical or electrical workshop, which are afterwards packed and sent (back) to the customer.

The parts between the storage and the assembly islands are order specific (assembly to order). An order is started when an assembly employee receives a bill of materials from the

computer and starts with picking of the materials. From this point the processing time of the order has a minimal average of 60 minutes including a total transport time of approximately 15 minutes. The production planning is based on the delivery date and the presence of the materials in the storage. If an order is made the system checks the stock of materials and if a part is out of stock this will be ordered automatically. If the part is delivered and scanned at the cargo supply, the order will be released to the assembly. All different processes have their own employees, which are divided over the different islands on base of the quantity of orders of the working places, under responsibility of the production manager. Most of the employees are trained for multiple production processes (for example one employee can paint and assemble) which can lead to changes in quantities of employees in the different buffers. The transport between the storage and assembly islands and between the packing station and transport truck at the end of the process is done manually with use of forklifts.
The other transport processes are done by a conveyor belt (orange rectangle) and rail system (blue rectangle).
These three transport processes are qualified for replacement by AGVs and three concepts are explained separately in the next section.

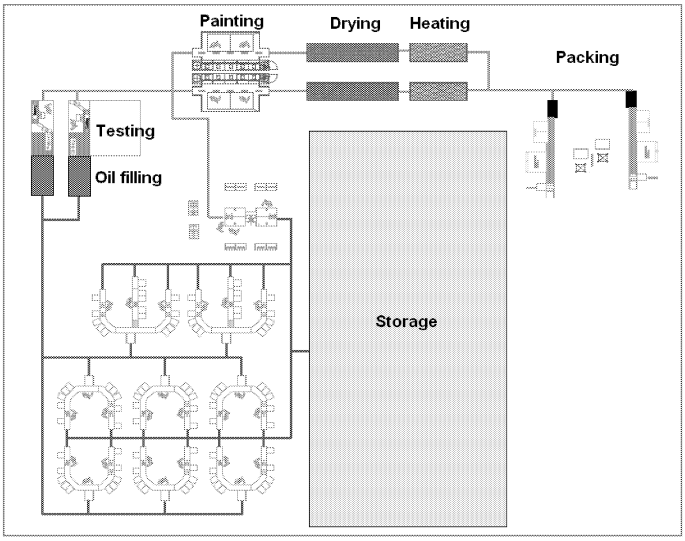


Figure 3 Overview of the routes

CONCEPTS OF THE AGV SYSTEMS

3.1 Blue route

The first route contains the transports between the storage and the assembly islands. The concepts for this route differ in the type of storage:

- Central warehouse; the parts will be picked order specific in a central storage and placed on a tablet. If all the parts are picked the tablet will be transported to the assembly island by AGV;
- Decentral warehouse; all parts will be picked in a decentral storage (supermarket) near or in the assembly island and an AGV can deliver the parts to this supermarket from the sorting island or other supermarkets;
- Central and decentral warehouse; the small and dedicated parts for the gearboxes are in or near the assembly islands and could be replenished by AGV or at least transported to the storage by AGV. The rest of the parts will be picked on order in a central warehouse and transported by AGV directly to the assembly island.

Because of the low purchase and maintenance costs, low complexity, quantity of buffer places and handlings and high flexibility for different sizes and quantities the central warehouse seems a good solution for the route between the warehouse and the assembly islands. See figure 3 for an overview of the routes. In order to simplify the simulation stocking and picking processes are not modelled and central warehousing was taken as a starting point.

3.2 Red route

The red route contains the transport between the assembly islands and the oil fill and test station. The concepts for this route are:

- Separated routes; one AGV drives between the assembly islands and oil filling station and another AGV drives between the oil filling and testing station. A minimum of two AGVs is required for this concept;
- A combination of the routes; one AGV transports the drives from the assembly station to the testing station and the drive is filled with oil on top of the AGV;
- A minimum of routes; the oil filling and testing stations will be combined to one station and the AGV transport the drive only from the assembly island to the oil filling station. Each test station has his own oil filling station.

The three concepts are tested in the simulation model and the results and conclusion can be found in sections 5 and 6.

3.3 Green route

The green route contains the transport between the testing and packing station. The four concepts for usage of AGVs are:

- The AGV drives through all the processes and waits on the process until it is finished; the drive will be loaded at the test station and unloaded at the finish station;
- All transports between the different processes are done by a different AGV, which means a minimal requirement of four AGVs;
- All routes are done by the same AGV, but the AGV do not drive through the painting station and do not wait for the processes; an AGV unloads at each process a drive and loads a finished drive from the same process;
- All routes are done by the same AGV, but the AGV will drive through the paint station where the AGV waits for the paint process.

The four concepts are tested in the simulation model and the results and conclusion can be found in sections 5 and 6.

THE SIMULATION MODEL

The goal of the simulation is to provide information about the different transport concepts and the production processes. The code of the simulation is based on multiple elements, shown in figure 4, and is made by using Delphi in combination with the TOMAS simulation package (Veeke and Ottjes 2002).

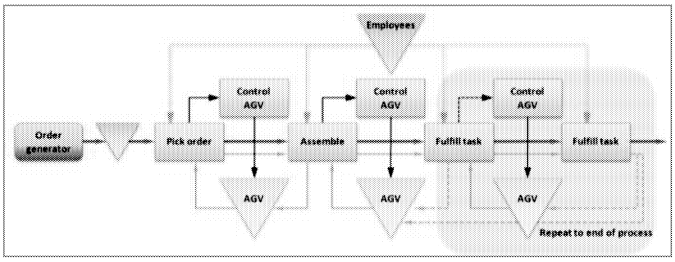


Figure 4 Model of the simulation and elements

It is possible to change the type of concepts and quantities of AGVs, see figure 5.

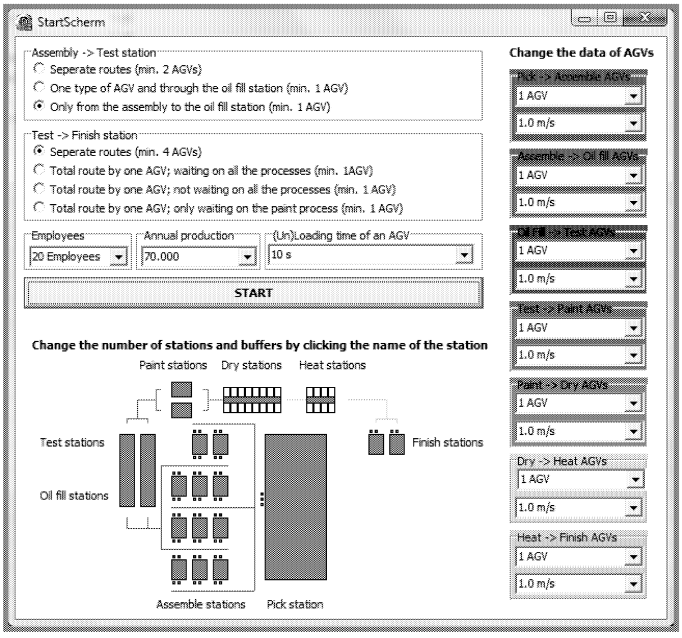


Figure 5 First (input) form of the simulation model

If the user has pushed the start button the actual simulation started. The working principle of the simulation in a nutshell: It all begins with the creation of orders, which normally come from a client system but here they come from a order generator. The order is generated with specifications of the determined number of picks and at which assembly station the drive must be assembled. If all parts are picked the order is placed in a queue of the AGV control and an AGV will transport the order to a buffer of the assembly station. The assembly station waits the determined assembly time and places the order in a queue of the next AGV control. The AGVs on the red and green routes transports the order to the next stations conform the chosen concept. All stations of the different processes have their own process times, which are provided by use of uniform distributions.

Figure 6 shows the form with a graphical interface at the left side and data about the progress, quantities and waiting times of the different processes and AGVs. Different colours are used in the graphical interface to indicate the status of processes and the AGVs.

4.1 Input data

The used data of the simulation is based on the production data of 2008 and the process analyses of section 2. In this paragraph the most important input data is explained.

- Interarrival time of orders

The interarrival time is based on the annual required quantity of drives, which is 70.000, with peaks of 100.000 drives. The interarrival time between the orders is the total annual production time divided by the annual quantity of drives, which represents an average of 100 seconds.

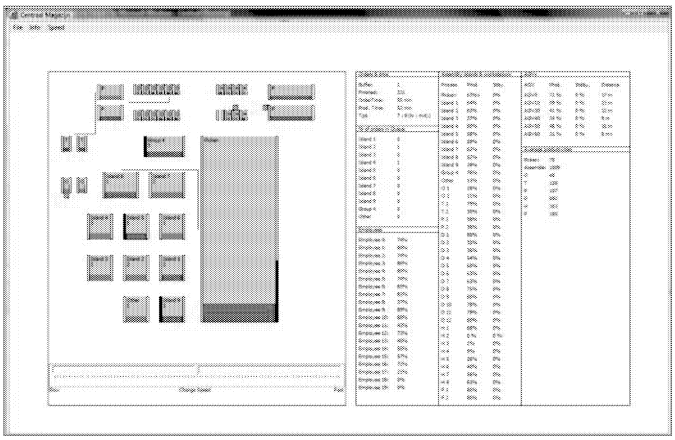


Figure 6 Second form of the simulation model

- Distribution of orders and types of drive

All the assembly islands are specialized for a type of gearbox, which leads to a distribution of drives about the assembly stations. Also the type of drives are different (gearbox and a normal or stock motor or only a gearbox or motor). Table 1 shows the distribution of the drives on the islands. [Vector,2008]

- Process times

The picking time is based on the number of parts of a drive and the average picking time. The assembly time is different for all the types of drive and the different islands, which is caused by the complexity of the gearbox, which is measured , together with the process times of the other workplaces, in the

current production process. If the measured data differs a uniform distribution is used in the simulation model.
The data from the simulation model is verified with the input data to check the accuracy of the simulation and only small changes are observed, which are caused by the multiplication

	Isle 1	Isle 2	Isle 3	Isle 4	Isle 5	Isle 6	Isle 7	Isle 8	Isle 9	Gr 4	Else
Percentage	13%	13%	7%	18%	8%	8%	10%	8%	6%	4%	6%
Cumulative	13%	26%	33%	51%	59%	67%	77%	85%	91%	94%	100%
Gearbox & normal motor	42%	56%	59%	98%	56%	69%	66%	86%	78%	84%	0%
Gearbox & stock motor	50%	42%	30%	0%	42%	27%	30%	6%	16%	2%	0%
Gearbox without motor	8%	2%	11%	2%	2%	4%	4%	7%	6%	13%	0%
Motor	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	100%

Table 1 Distribution of orders and type of drives

of a large amount of distributions. Because the changes are less than 1% they are neglected.

Results

The results and the analyses of the different routes are shown in the next paragraphs. All figures contain a figure of the concept, the required AGVs with different daily production quantities and the average productivities.

5.1 Blue route

The blue route requires two AGVs for the maximum production quantity, which must transport two tablets from the warehouse to the assembly islands. For independent loading and unloading of the tablets roller decks with a push or pull device can be used.

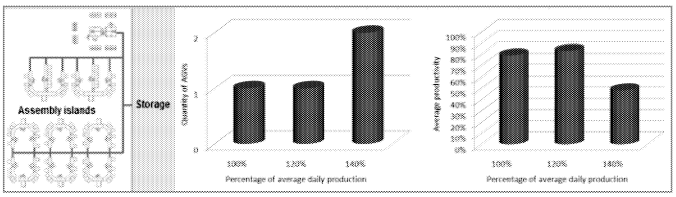


Figure 7 Results of the blue route

5.2 Red route

The results of the best solution of the route between the assembly islands and the oil filling and testing station are shown in figure 8. The concept is chosen with the minimal number of routes and the combination of the oil filling and testing station, because of the low quantity of AGVs and handlings, low complexity, a high controllability of the oil filling process and the possibility to maintain one of the two AGVs if the production of drives is lower than the 140% of the average daily production quantity.

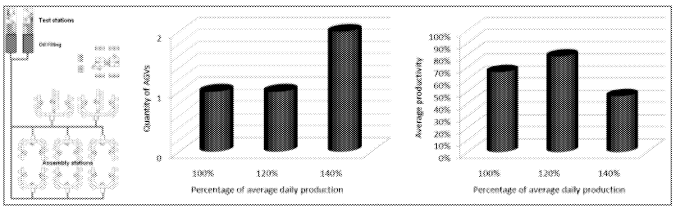


Figure 8 Results of the red route

The two AGVs have to transport only one tablet from the assembly station to the oil filling station, which is the same tablet as used on the blue route. At the test stations the empty tablets can be buffered for the return transport to the picking station and can be transported to the picking station during the breaks or in the evening.

At the assembly station the motor and gearbox are placed together on the tablet, which increases the maximum weight of the tablet to 250kg. For loading and unloading one roller deck with push and pull device can be used. It must be possible for the AGV to load and unload independently at all (un)load locations at the islands and storage.

4.3 Green route

The results of the best solution of the route between the testing and the packing station are shown in figure 9. The route with the total replacement of all the current transport systems is preferred for the total concept, because the AGVs are the same for all different transport processes and with the possibility to expand the system. The replacement of all the current transport systems by AGVs has a single uniform and transparent transportation system as result.

Five AGVs are required for the maximum productivity of drives per day. Because the drive is transported through the painting, drying and heating processes, it is required that the drive is hanging at the AGV instead of resting on a tablet. This because the wet paint and the required space around the drive in the painting cabin. By using a mechanical coupling in the hanging mechanism of the drive it is possible to load and unload the drive automatically at buffer stations in the drying and heating cabins.

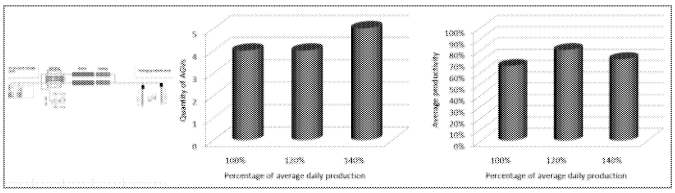


Figure 9 Results of the green route

The chassis of the AGV (with the drives, control and energy storage) can be the same as for the other routes, which saves costs in design and maintenance. Only the structure on top of the AGV is specific on all routes.

5.4 The total concept

The combination of routes is shown in figure 3 and the overview of the required quantities and productivities of the buffers is shown in figure 10. Impressions of the used AGVs are shown in figure 11.

A modular concept is used for the AGV system. The chassis of the AGV (with the drives, control and energy storage) are

the same for all AGVs and only the (un)loading device on top of the AGV differs.

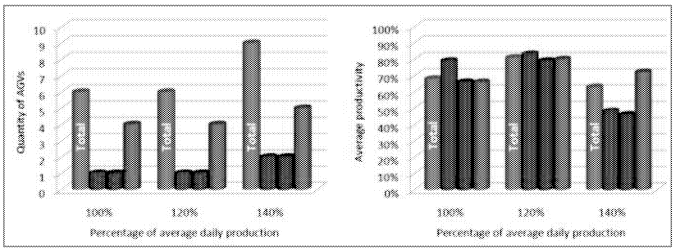


Figure 10 Overview of the quantities and productivities

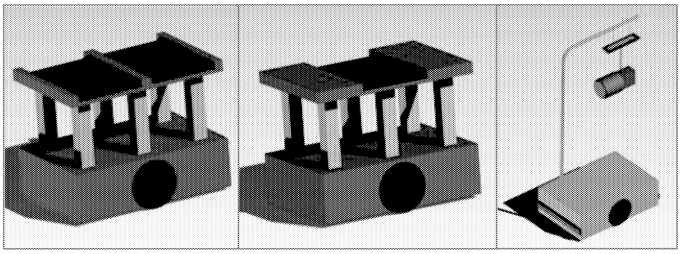


Figure 11 Impressions of the used AGVs

Buffer stations and locations

The buffers stations of the AGVs are near the loading locations of the transports to reduce the drive time to a minimum. During the waiting time in a buffer station it is possible to charge the energy storage of the AGV. For the processes buffer locations are also required, which can avoid the peaks and differences in process times and are located at the beginning or end of the manual processes. The quantity of buffer stations depends on the difference in output or input of the stations.

Information transfers

Because of the use of different routes and a relatively small quantity of AGVs a simple AGV control without a large quantity of information transfers is required. The first information transfer in the system is the detection of a tablet in a buffer/loading place, which indicates a new transport. If a tablet is detected the AGV control receives a "call" for an

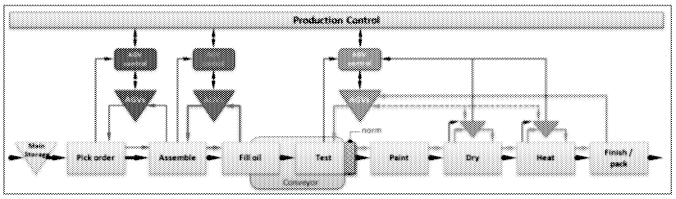


Figure 12 Information transfers

AGV. The AGV control passes this order directly to an idle AGV in the buffer station, else the AGV control will store the order in a queue until an AGV reports itself available for a new transport. Figure 12 shows the different information transfers.

The information at the end station is enough for the blue and red route, from picking to oil filling station, but more information is required for the route between the drying and heating cabins. If the drive is unloaded at a buffer station in the dry or heat cabin, the AGV control must determine if the AGV can load a new transport from the dry or heat stations, in case a drive is ready for the process, or drives back to the test station, because in the near future no drive is ready for that process. The information about the status of the drive in the dry and heat cabin is stored in the AGV control.

Costs

The expected costs of a transportation system with AGVs is compared with the current transport system. The costs are based on offers from different AGV and rail system suppliers [FROG, 2009] [Rocla, 2009] and in company knowledge about employee and conveyor belt costs. The costs per year for all transport methods are based on the purchase and maintenance costs. Table 2 shows the costs of the two systems for the three different routes. The current system contains manual transport (M), a conveyor belt (C) and a rail system (R).

	AGV system	**Current system**
Blue route	€ 26.000,-	€ 50.000,- (M)
Red route	€ 26.000,-	€ 23.000,- (C)
Green route	€65.000,-	€38.000,- (R)
Total costs	**€117.000,-**	**€ 111.000,-**

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

Nine AGVs divided over three routes are required for replacement of the present transport systems from the storage to the packing station within the main production process of Vector Aandrijftechniek. The AGVs replace the current manual transports with carts between the storage and assembly islands, the conveyor belts between the assembly islands and oil stations and the rail system between the test and packing station. Because the AGVs do not have fixed routes in the production layout less floor surface is required and all islands are easy accessible. It is also easy to change the positions of the (un)load places, because the route of the AGV is virtual.

The maximum total weight of all transports is the same for all three routes, but the quantity of tablets and the way of loading and unloading is different, which requires a modular AGV; the chassis with the drives, control and energy storage is the same for all AGVs and the (un)load device for all AGVs is different. The advantages of the modular structure include reduced engineering and maintenance costs and in case of problems it is possible to exchange the (expensive) chassis between the routes. The use of automated loading and unloading devices makes the AGV completely independent of

the other systems and makes it possible to use the current islands.
Due to the relatively high expected maintenance costs in comparison to the present transport systems (rail and conveyor belts), the AGV system is not the cheapest transport system, but because of the flexibility, reliability (slight overcapacity and individuality of the AGVs) and less required floor surface the AGV system is a suitable solution for the transport systems in the Model Eurodrive. Almost all other Eurodrives have the same main production processes, with only some differences in product range and production quantities, which makes it is possible to replace the current transport systems with the same AGV system.

6.2 Recommendations

The AGV system must be very reliable, therefore it is recommended to make all components and especially the AGV control as simple as possible. As the number of required AGVs is low and the transport routes are not complex, this is feasible.

Because the implementation of an AGV system is complex and the experience of employees with the usage of AGVs is minimum, it is recommended to test the AGV system in a simple transport process in the current production process. It is advised to start with a simple route, with only two or three (un)loading points and only one job at the time. If this implementation is successful, the number of jobs and (un)loading points can be enlarged.

The required quantities of AGVs for the new building are based on 70.000 drives per year, which is 13.000 drives more than the current production quantity. To realize the transports with a peak in the production process (100.000 drives per year) nine AGVs are required, but with average values three of them are idle. As it is easy to expand the system with more AGVs it is recommended to start with only six AGVs, which decreases the purchase costs and makes two of the routes easy to control, due to the lower number of AGVs on the routes.

ACKNOWLEDGEMENTS

This work is supported in part by Vector Aandrijftechniek BV.

REFERENCES

FROG, 2009, Free ranging on grid, http://www.frog.nl/Technologie/Frogtechnologie

Rocla, 2009, Automatic Guided Vehicles, http://www.rocla.fi/productlist.asp?Section=477

Veeke, H.P.M., Ottjes, J.A. and Lodewijks, G., 2008. The Delft System Approach: analysis and design of industrial systems, Springer.

Veeke, H.P.M. and Ottjes, J.A., 2002 TOMAS: Tool for Object-Oriented Modelling and Simulation. In: Proceedings of the Business and Industry Simulation Symposium. Washington D.C.

Veld, J. in 't, 2002. Analyse van organisatieproblemen, een toepassing van denken in systemen en processen, Stenfert Croese.

INTERMODAL TRANSPORT SIMULATION

FURTHER DEVELOPMENT OF INTERMODAL TRANSPORT IN BELGIUM: THE PORT OF ZEEBRUGGE

Ethem Pekin
Cathy Macharis
Ellen Van Hoeck
Tom van Lier
Vrije Universiteit Brussel
Department MOSI-Transport & Logistics
Pleinlaan 2
B-1050, Brussels, Belgium
E-mail: Ethem.Pekin@vub.ac.be

KEYWORDS
intermodal policies, intermodal terminals, GIS model.

ABSTRACT

This paper describes a geographic information system-based location analysis model, developed to analyse the characteristics of the Belgian intermodal terminal landscape. The model, which originally takes the port of Antwerp into account, is extended to include the port of Zeebrugge. Based on the transportation costs, the model compares intermodal transport with unimodal road transport. After visualising the current intermodal terminal landscape, the model handles various scenarios. Highlighting the market areas of intermodal terminals, the model is used as a policy support tool to come to an integrated vision on the future development of intermodal transport in Belgium.

INTRODUCTION

Due to the globalisation of the economy and the associated growth of international trade, the increase in freight flows became higher than GDP growth. From the sixties, freight is containerised and transported more and more by containers. As a result, overseas transport substantially became cheaper and more efficient. In parallel to the developments at the maritime side, which includes the building of a growing number of container terminals at sea ports and the significant enlargement of container vessels; logistics have also adapted itself towards this containerisation of freight, namely through intermodal transport. Taking a quick look at the major sea ports, one can observe that a significant increase in the amount of containers that were handled especially during the last decade. The sea ports in Belgium including the port of Antwerp and the port of Zeebrugge, for example, faced an annual increase of approximately 10 percent. Of course these containers need to be transported on to the hinterland and this is where intermodal transport plays an ever increasing role.

Intermodal transport is defined as the combination of at least two modes of transport in a single transport chain, without a change of the loading unit for the goods, with most of the route travelled by rail, inland waterways or an ocean-going vessel, and with the shortest possible final journey by road (ECMT, 2003). The movement from one mode of transport to another usually takes place at an intermodal terminal. In Figure 1, a typical maritime-based intermodal chain is shown. Containers are being handled in the port, are further transported by barge or rail wagons towards an inland terminal, from which the containers are transported by road to the end destination.

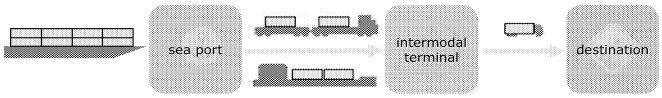


Figure 1: Intermodal transport chain

Providing various socio-economic benefits, transport also generates adverse effects and the growth in transport use is likely to make these impacts much more dense. Since theeighties, road transport increased its market share, following technological and economic developments. Today, road transport is widely included in transport systems and constituting a more liberalised market, it is highly competitive with its flexibility. On the other hand, the problem of congestion in the road network decreases its reliability and increases the transportation costs. In addition, road transport generates more negative externalities such as climate change, air pollution and accidents than rail or barge transport in most cases. In this setting, intermodal transport receives an important attention. Therefore intermodal transport is promoted through policies being addressed at all political levels. Policy measures may include amongst others: provision of intermodal infrastructure, subsidies for new services, research and development activities etc.

This paper aims to evaluate the impact of a major infrastructure project, which constitutes a potential for further growth of intermodal transport in Belgium. To this end, a geographic information system (GIS)-based intermodal transport model, called the LAMBIT (Location Analysis Model for Belgian Intermodal Terminals) is developed. The paper focuses on the port of Zeebrugge and the Seine-Scheldt-West (SSW) project, the large-scale European transport project enhancing sustainable development, improving accessibility of regions and raising economic efficiency. The paper analyses the opportunities and impact of the SSW project on the intermodal terminal landscape. Current policy measures directed towards

intermodal transport such as subsidy schemes are also incorporated into the scenario-based analyses.

In section 2, intermodal transport in Belgium is introduced. Section 3 explains the policies towards the port of Zeebrugge, whereas section 4 presents a scenario-based analyses for the port of Zeebrugge. Finally section 5 draws conclusions for this paper.

INTERMODAL TRANSPORT IN BELGIUM

Belgium has an extensive transport network to distribute containers arriving at the ports of Antwerp and Zeebrugge. During the last decade, new volumes handled at the sea ports have lead to further possibilities for intermodal transport in Belgium. From a social point of view, intermodal transport took a growing political interest especially for its environmental performance. Intermodal transport is in most cases more environmental friendly than road transport (Kreutzberger et al., 2006). As a result, the number of intermodal terminals increased considerably during this period. In 2009, there are 19 intermodal terminals in Belgium of which there are 13 barge (inland waterway) and new projects are coming. The rail/road terminal landscape has not evolved as rapidly, but new services were nevertheless set up. In 2004, the railway company B-Cargo decided to set up the Narcon (National Rail Container Network) concept in order to offer intermodal transport by establishing rail/road services between the port of Antwerp and the port of Zeebrugge and various inland terminals. For international routes, specialised direct trains are scheduled. Figure 2 indicates a concentration of terminals near the Flemish waterways. At the moment, new terminals are planned for Wallonia. It has to be noted that most of the barge terminals are offering daily services to the ports of Antwerp and Rotterdam but some of them have a small scale such as terminals in Herent and Mol.

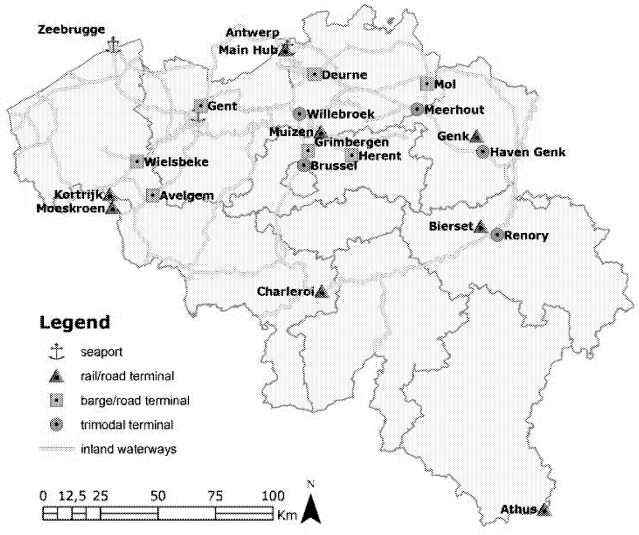


Figure 2: Intermodal terminal landscape in Belgium

The modal split of the port of Antwerp indicates a move towards the inland waterways and rail for container transport. In 1996, 70 percent of the containers were transported by road, 6 percent by rail and 24 percent by barge. In 2009 this was 10 percent by rail, 35 percent by barge, leaving 55 percent for the road sector. Even with this evolution in modal split, in absolutely terms, there is still an increasing amount of containers that are transported by road every year. Compared to 1996, 2,15 million TEU (twenty-foot equivalent) is transported by road in 2007. This underlines the importance of projects like the Seine-Scheldt-West, which can provide further opportunities for intermodal transport. This project will be explained in the next section.

POLICIES TOWARDS THE PORT OF ZEEBRUGGE

There is a relationship between containerisation and intermodalism. Parallel to the growth in freight transport, there has been a growing use of containers that enabled the operation of intermodal transport systems. Intermodalism can be built upon two phases. First phase is the growth of container transport at the maritime side. Freight is containerised and transported more and more by containers. Figure 3 shows that the containerised traffic has almost tripled from the 1990s to the early 2000s. Focusing on the important northern European ports, an average annual growth rate of 9 percent is seen. The Belgian ports of Zeebrugge and Antwerp have sustained an average of respectively 11 and 9 percent growth between 1980 and 2009. The second phase of intermodalism can be found at the landside, where containerisation is extended to the hinterland of the seaports (Hayuth, 1987).

The port of Zeebrugge handled in 2009 a total volume of 2.3 million TEUs. In spite of the economic and financial crises which set in the last quarter of 2008, container handling has increased by 9.4 percent compared to 2007. In tonnage the total container traffic also increased by 4.3 percent to 21.1 million tons. It has to be noted that the container transport accounts for half of the handled cargo of the port, where Roro transport also takes an important share. One of the key success factor for the port of Zeebrugge lies in the growth registered in the deepsea segment. Focusing on the maritime accessibility, the port enjoys the competitive advantage of being able to handle the largest container vessels (8,500-14,000 TEU).

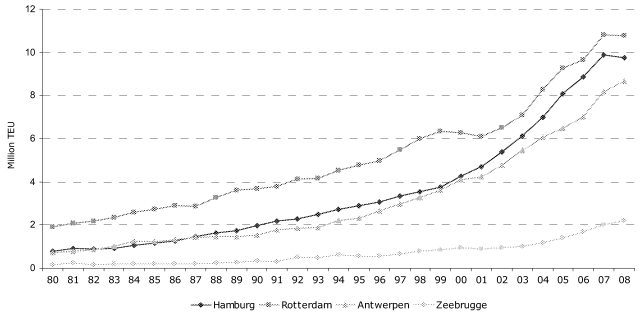


Figure 3: European container traffic growth

The port of Zeebrugge also evolved into a strong container hub thanks to various container feeder services. In this context, special attention was paid to rail and waterway connections towards the hinterland of the port. In Figure 4

the modal split of containers for the inland traffic of the port of Zeebrugge is presented. Estuary shipping has been established in 2008 to ship containers from Zeebrugge via the sea to the Scheldt estuary. During this period, considerable cargo volumes from and to the Antwerp region and to the Albert Canal have been realised. Special inland ships were developed with loading capacities of 250 and 350 TEU respectively to reach the Scheldt estuary by sea. The ships with the smaller capacity connect the port with the inland waterway terminal in Meerhout through the Albert Canal. It has to be noted that it takes 15 hours to cover the distance between Zeebrugge and Meerhout. The larger capacity ships on the other hand establishe an intensive connection to the Antwerp region. However, substantial cargo volumes are transported by rail especially with Germany, France and Italy. As explained in previous section, national rail transport between the sea ports of Antwerp and Zeebrugge is also established. Road transport accounts higher share in the modal split due to the truck combinations that are used in roro transport and the port of Zeebrugges' leading position in the world in the handling of new cars. Considering the growth rate of container transhipments for the port of Zeebrugge, Figure 4 can conclude that estuary shipping and inland navigation should further be promoted for achieving a sustainable modal split. In this respect, the port of Zeebrugge aims to expand the usage of inland navigation by upgrading the existing diversion canal of the river the Leie within the Seine-Scheldt project.

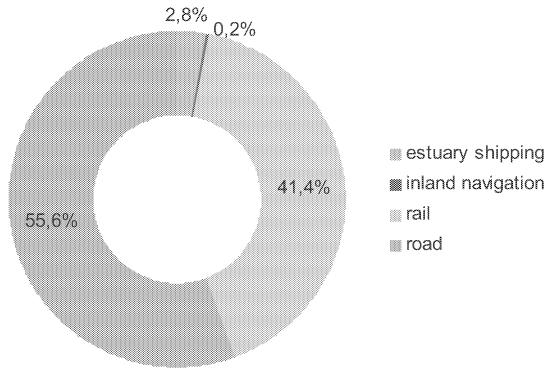


Figure 4: Modal split of container for inland traffic of the port of Zeebrugge

The entire Seine-Scheldt project consists of different sections in Belgium and Northern France which, once linked together, will link Paris and Ile-de-France to the north-western European waterway network. This will give a lot of advantages which are listed below.

- This connection will provide a solution for logistic players by linking large economic centres. The existing and planned multimodal platforms near the waterways bundle and distribute goods, attract industrial and logistics activities and create added value for the regional economy and population by penetrating into the heart of urban areas.
- The link Seine-Scheldt will relieve the north-south axis because it will carry large volumes of bulk and high value goods as well as exceptional and outsize convoys.
- The project is an example of sustainable development as inland navigation is the most environmentally friendly mode of transport. According to Inland Navigation Europe the Seine-Scheldt connection will be more eco-efficient than other ways of transport.

In Belgium the Seine-Scheldt-West (SSW) connection is seen as the logical completion of the international Seine-Scheldt project. To realise the SSW connection, a present alignment is used via the Port of Zeebrugge through the diversion canal of the Leie. The aim of the diversion canal of the Leie is to give the opportunity to the coasting ports to reach the inland vessels within the Trans-European inland waterway network. In Figure 5 a visualisation of the projects is given. The red line represents the Seine-Scheldt project which opens up the Trans-European inland waterways network. The green line is part of the Seine-Scheldt project, called Seine-Scheldt-West which includes maritime traffic to the port of Zeebrugge.

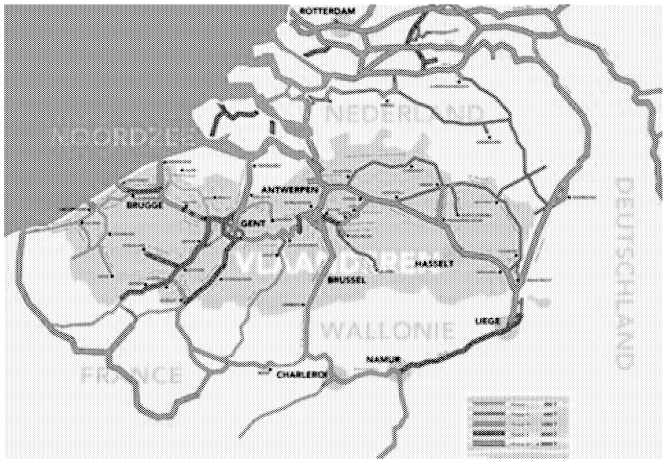


Figure 5: SS and SSW project

To realise the SSW connection, the diversion canal of the Leie needs to be adapted for inland ships with capacities of 4500 tonnes. Through the project, one million truck movements over the road are avoided which has a positive outcome on both environmental (e.g. climate change and air pollution) and non-environmental transport externalities (e.g. accidents and congestion).

The advantages of the SSW are partly on the economic importance of the Flemish ports in terms of employment but it also attracts foreign investments in sectors as industry and logistics (Waterwegen en Zeekanaal, 2008).

The focus of this paper lies on the comparison of intermodal and unimodal road transport specific for the port of Zeebrugge. Therefore the Location Analysis Model for Belgian Intermodal Terminals (LAMBIT) is used. This model is extended by including the Port of Zeebrugge in the specific Seine-Scheldt project. In the next section the LAMBIT model is explained.

ANALYSING PORT OF ZEEBRUGGE WITH THE LAMBIT MODEL

In this section the methodology of the LAMBIT model is explained, followed by the results of the scenarios for the port of Zeebrugge.

Methodology

LAMBIT is a geographic information system (GIS)-based location analysis model which makes it possible to conduct ex-ante and ex-post analysis of policy measures in favour of intermodal transport. The LAMBIT model explores the relative attractiveness of each transportation mode through a price (cost) minimisation model. The model develops several scenarios, namely policy measures that are applicable for intermodal transport:

- Location of new terminals
- Price scenarios
- Subsidies
- Internalisation of external costs

Construction of the model

LAMBIT is built on three main inputs: transportation networks, transport prices, container flows from the municipalities to and from the sea port.

1. Transportation networks

LAMBIT is composed of the different network layers (for each transport mode) and the location of the intermodal terminals and the port of Antwerp (as nodes in the network) (Macharis, 2000 and 2004). In this paper, the port of Zeebrugge is added to the model. A GIS network was set up by including four different layers: the road network, the rail network, the inland waterways network and the final haulage network. The geographic locations of the intermodal terminals and the municipality centres are defined and connected to the different network layers.

Figure 6 depicts the different network layers and nodes including three intermodal transport modes. The networks for Belgium were built by merging the following digital databases:

- Road layers and municipalities are obtained from the MultiNet database of Tele Atlas
- Rail and inland waterways layers are extracted from the ESRI (Environmental Systems Research Institute) dataset for Europe

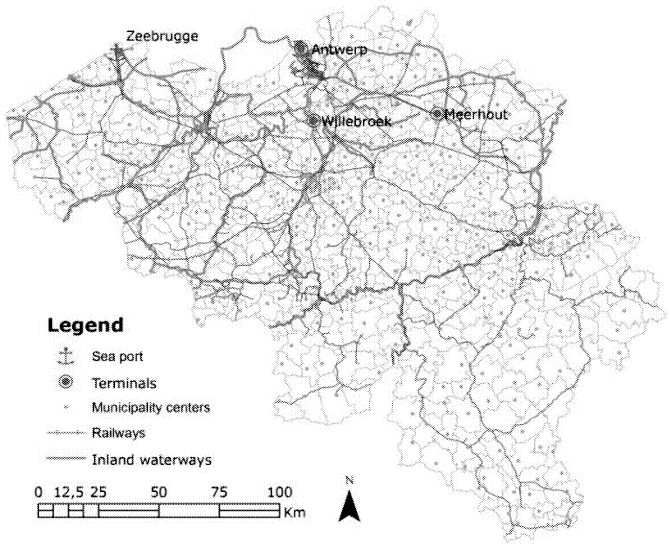


Figure 6: Network layers and nodes

2. Transport prices

The LAMBIT methodology is based on two concepts: the intermodal cost structure and the break-even (critical) distance. Considering the total transport prices and the distance travelled, unimodal road transport is cheaper in the short distances but once the break-even distance is achieved, intermodal transport offers a competitive alternative.

The transport prices are calculated based on the real market price structures for each transport mode and they are associated with the network layers. The variable costs are uploaded to the network layers and the fixed costs are attached to the nodes, which also indicate the origin and destination for each path.

The total price of intermodal transport is composed of the transhipment cost in the port of Antwerp to a barge or a wagon, the cost of the intermodal main haul (barge or rail), the transhipment cost in the inland terminal to a truck and the cost of final haulage by truck. The following formula explains the calculation of intermodal transport:

The total intermodal transport cost is obtained by adding all of these fixed and variable costs based on the existing market prices.

3. Containers from the Belgian municipalities

The final input for the LAMBIT analysis is the container flows from the sea ports. In this paper, the statistics of road transport from the Directorate-general Statistics and Economic Information of Belgium were used.

Functioning of the model

Using a shortest path algorithm in ArcInfo, various comparisons are conducted in order to find the shortest path and the attached transport price from the port of Zeebrugge to each Belgian municipality via intermodal terminals and via road only. For each destination, the total transport price for unimodal road, estuary shipping/road, inland waterways/road and rail/road transport are compared and the

cheapest option is selected. The output consists of the market areas of each inland terminal are highlighted in the map of the model. These visualisations help us to see how large the market area of each intermodal terminal is. As a further step, the container flows data can be used to show the amount of containers that are currently transported by road to the municipalities within the market area, which gives an indication of the existing potential volume that can still be shifted. This is particularly useful when a new service or location of a new terminal needs to be analysed.

The port of Zeebrugge case

The port of Zeebrugge is currently connected with the Flemish hinterland by three intermodal routes: terminals in port of Antwerp, the trimodal terminal in Meerhout and the barge terminal in Willebroek. For the moment, only rail and estuary shipping is possible between the port and those destinations. Concerning rail transport, the port of Zeebrugge is connected with the port of Antwerp through the Narcon network, which is subsidised by the federal government. Narcon network and rail subsidies have already been analysed with the LAMBIT model (see Macharis and Pekin (2008)). In this paper, the main focus will be on inland navigation, which would become possible if the SSW would be realised. In this section first a current situation will be presented. Then a future scenario will be developed to show the situation when the SSW connection is implemented. The analysis is concluded with a subsidy scenario.

Current situation

In Figure 7 the current situation for the market area of intermodal terminals is shown. In this scenario, current market prices (without any subsidies) for rail transport and estuary shipping are used to show the market area of the terminals. The municipalities are highlighted, when intermodal transport has a more attractive transport price compared to unimodal road transport. Only the terminals in Willebroek and Meerhout take market area. For the port of Antwerp, road transport remains a cheaper option compared to estuary shipping as a result of extra handlings and post-haulage. Rail transport has higher market prices when subsidies for rail transport are not considered. This explains why there is no market area for the rail terminals in Antwerp and Meerhout. However in reality containers are transported by rail to/from the port of Zeebrugge such as the rail shuttles to the Main Hub in Antwerp.

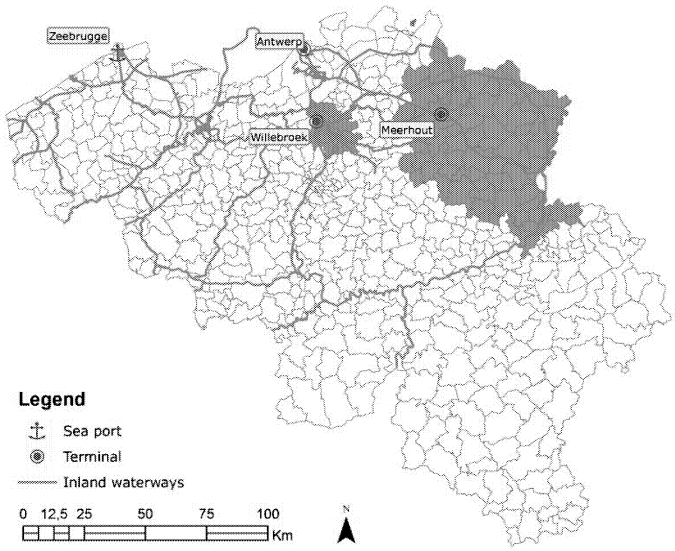


Figure 7: Current situation

Estuary shipping from the port of Zeebrugge has initiated in 2008 towards Willebroek, Antwerp and Meerhout and will run for 10 years until 2018. Estuary shipping presents higher depreciation costs (damages to the estuary barges), which will limit their life cycle. Additionally the handlings of estuary barges are difficult. These characteristics of estuary shipping have a negative impact on its competitiveness. This can even be critical when the subsidy enters its fourth year, when minimum container rates will be required from the shipping lines.

Implementation of the SSW connection

In the long term, connection of the port of Zeebrugge to the Flemish inland waterway network, namely through the SSW-project, can provide benefits for inland navigation. In this section, a hypothetical future scenario is developed with the SSW connection. Once this connection is established, inland waterway barges departing from the port of Zeebrugge can access the Belgian inland waterways. Therefore main barge terminals in Belgium are also added to the LAMBIT model.

A container flow analysis is performed to verify hinterland potential of the port of Zeebrugge. Figure 8 shows the container flows from the port of Zeebrugge, based on the national statistics. The container flows density indicate that the terminals in Meerhout, Willebroek and Renory are located in the network, where there are higher freight volumes from the port of Zeebrugge. The figure also shows to large flows to the port of Antwerp, due to the bills of lading which is often Antwerp and not Zeebrugge.

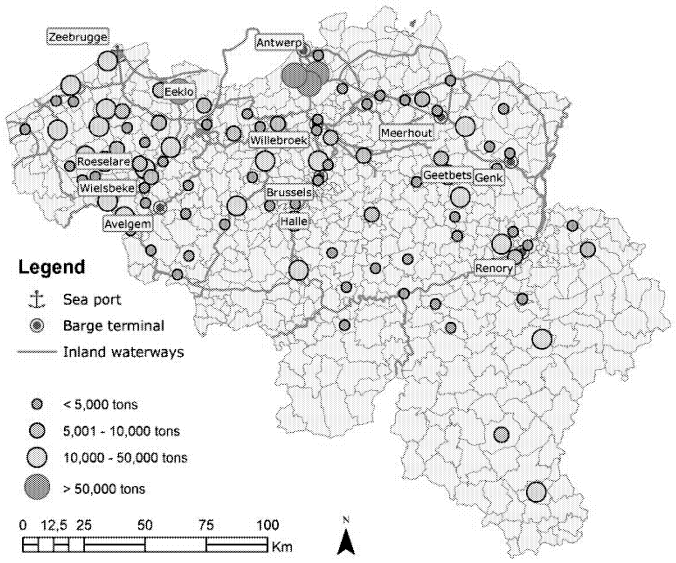


Figure 8: Container flow analysis from the port of Zeebrugge

The implementation of the SSW connection is shown in Figure 9. Considering the current market prices for barge transport, inland waterway transport is competitive in 182 municipalities. The terminals which are located far from the port of Zeebrugge (Renory, Brussels, Meerhout, etc.) benefit more from the lower variable costs of intermodal transport compared to unimodal road transport and so they have larger market areas. This is explained by the intermodal cost structure. The longer the distance travelled, the greater the extent to which the lower variable costs of intermodal transport can compensate for the extra transhipment costs at the terminals.

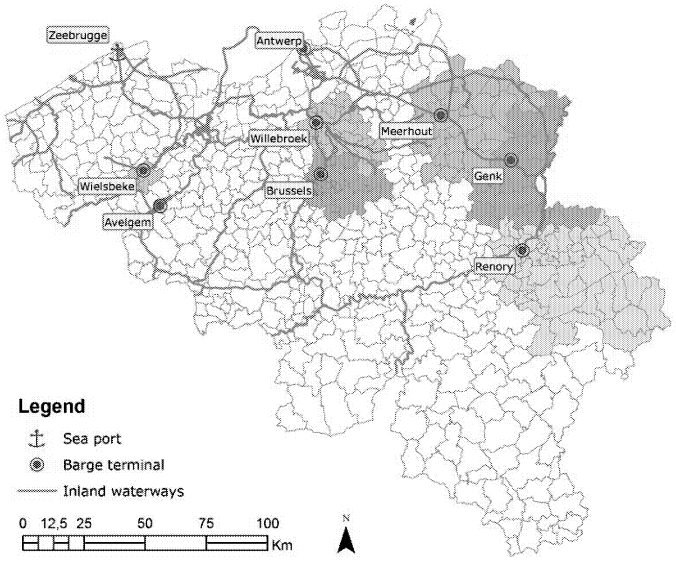


Figure 9: Barge connection - SSW project

As illustrated in Figure 9, the SSW project can lead to a modal shift from road transport to barge transport. The market areas for intermodal terminals increase considerably when the port of Zeebrugge is connected to the inland waterway network. The number of municipalities provides an indication of the potential of barge transport. However, this analysis should be completed with an idea on the amount of containers that are transported from the port of Zeebrugge to all these municipalities. Taking the national statistical data into account, the addition of the canal results in an over three-fourths potential increase in intermodal transport meaning a modal shift of 98,509 tons. This tonnage is a potential modal shift from road transport. In order to calculate total potential for barge transport, the flows that go now via estuary shipping and even rail transport should also be considered. Overall the barge terminals sum up to 227,515 tons. Table 1 presents the changes in the market potentials for the intermodal terminals with respect to the current situation.

Table 1: Market area of intermodal terminals when the SSW connection is implemented

Intermodal Terminals	Estuary shipping		Barge transport	
	The number of municipalities	Volume in ton	The number of municipalities	Volume in ton
Meerhout	74	117,135	31	90,463
Wielsbeke			4	18,719
Willebroek	15	11,871	28	20,145
Avelgem			1	0
Genk			30	9,102
Renory			52	66,251
Brussels			36	22,835
Antwerp	0	0	0	0
Total barge terminals	89	129,006	182	227,515
Total road transport	500	1,753,864	407	1,655,355
Total	589	1,882,870	589	1,882,870

Subsidy scenario

An option that can be used to stimulate intermodal transport is to decrease the costs linked to the infrastructural development and to transport operations at the terminals. As mentioned, the inland navigation from the port of Zeebrugge could be subsidised. This could be done by two subsidies: estuary subsidy and barge subsidy. The estuary shipping aid scheme (N 53/06) aims:

- to achieve the modal shift from road to inland waterway navigation;
- to compensate the external costs that road transport does not incur;
- to generate sufficiently large amounts of traffic of goods after the expiry of the start-up period so that regular container service lines between the Flemish coastal ports and the hinterland can be operated without any state intervention necessary.

The subsidy is composed of investment (building of estuary ships) and exploitation (estuary services). For the first three years a maximum amount of € 4.4 per TEU is provided. Subsidies are also granted for each container handled by an intermodal barge terminal. This kind of measure for inland waterway transport is applied in Belgium in Walloon, Brussels and Flemish regions but at different tariffs for each. In May 2007, the EC authorised a Flemish measure to grant a subsidy of € 17.5 per container transhipped at a Flemish inland container terminal from or to an inland waterway vessel (N 682/06). Similar initiatives are also developed in Wallonia and Brussels. According to the government decision of December 2004, the Walloon government foresees a subsidy of € 12 for the containers that are

transhipped to a Walloon inland container terminal from or to an inland waterway vessel (OPVN, 2006). In 2008, this subsidy is extended for the period 2008-2013. An identical subsidy scheme is also valid for the Brussels region for the period 2007-2009 (N 720/06).

From the perspective of this paper, these subsidies are included in the model. In Figure 10 the results of this scenario is shown. Compared to Figure 9, the subsidy scenario shows that intermodal transport grows substantially thanks to the inland waterway subsidies. The subsidy enables the terminal in Antwerp to compete with unimodal road transport. An interesting finding from this scenario is that it visualises the impact of the regional differences in the degree of subsidies for inland navigation. This can be illustrated with the terminals in Willebroek and Brussels. The terminal in Willebroek will enjoy the subsidy scheme of the Flemish government (€ 17.5), which is higher than the subsidy scheme for the terminal in Brussels (€ 12). As a result the terminal in Willebroek takes market area from the terminal in Brussels. It has to be noted that the subsidy scheme for barge transport depends on the type of container. The Flemish subsidy considers intermodal loading units (containers). The subsidy scheme for Brussels on the other hand makes a distinction for the size of containers hence subsidy of € 12 is applicable for twenty-foot containers. The subsidy gradually increases to € 18 for thirty-foot containers and € 24 for forty-foot containers. Overall, the subsidy scenario decreases the market area of unimodal road transport by 108 municipalities. Taking the volumes in to account, the subsidy schemes can lead to further potential increase of 319,434 tons for intermodal transport. In Table 2 the market areas of terminals are summarised.

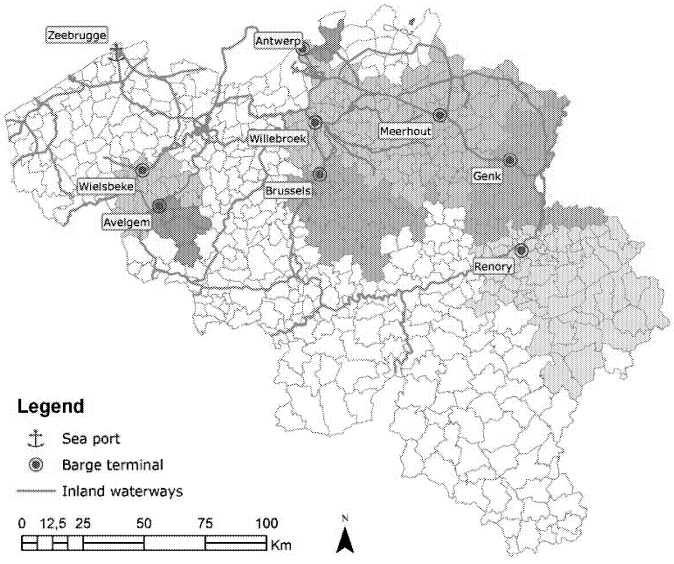


Figure 10: Subsidy scenario

Table 2: Market area of intermodal terminals when the subsidy schemes are introduced

Intermodal Terminals	Barge transport	
	The number of municipalities	Volume in ton
Meerhout	38	102,377
Wielsbeke	21	62,120
Willebroek	63	282,280
Avelgem	9	4,103
Genk	36	9,102
Renory	58	66,251
Brussels	62	18,875
Antwerp	3	1,840
Total barge terminals	209	546,949
Total road transport	299	1,335,921
Total	589	1,882,870

CONCLUSIONS

From the perspective of society, intermodal transport can help in reducing both the environmental (such as climate change and air pollution) and non-environmental problems (such as congestion and accidents) caused by our transportation system. By shifting freight and especially containers to barge, rail or estuary shipping and only use road transport for the pre and final haulage, congestion, exhaust emissions and accidents can be reduced. From the perspective of transport users, intermodal transport can be a cost-efficient option compared to unimodal road transport as the cost per unit decreases in barge and rail transport when economies of scale are experienced. These lower unit costs on the long haul should compensate for the extra handling costs at the inland terminal. This is possible if the distance between origin and destination is long enough (the so called critical distance).

In this paper, the intermodal transport network in Belgium was investigated. Focusing on the port of Zeebrugge, the main goal of the paper was to evaluate the potentials for intermodal transport from this sea port. To this end, the SSW project was analysed with the LAMBIT model.

LAMBIT is a GIS-based location analysis model which makes it possible to conduct ex-ante and ex-post analysis of policy measures in favour of intermodal transport. The LAMBIT model explores the relative attractiveness of each transportation mode through a price (cost) minimisation model. The model develops several scenarios and visualises the market area of intermodal terminals.

The scenarios described in this paper show that the current hinterland connections for the port of Zeebrugge to the terminals in Antwerp, Meerhout and Willebroek are exposed to certain limitations such as higher rail subsidies or difficult handling for estuary ships. In other words, without any rail subsidies shuttles between the port Antwerp and the port of Zeebrugge were not possible. Furthermore the estuary shipping constituted certain operational constraints and costs. Therefore the model took the SSW-connection into account to evaluate the potential for barge transport. The future scenario showed positive developments for intermodal transport. When the subsidies on inland navigation are considered as well, the potential benefits of the SSW project are even more emphasised.

REFERENCES

European Commission (2007) Subsidy case N682/06 Vlaamse regeling voor steun aan inter-modaal vervoer via de waterwegen.

European Commission (2007) Subsidy case N720/06 Subsidieverlening door de Haven van Brussel aan lijndiensten voor containervervoer.

European Commission (2006) Subsidy case N53/06 Pilot project of the Region of Flanders for the granting of aid to estuarine navigation and inland waterway navigation for the transport of containers from and to the Flemish coastal ports.

Hayuth, Y. (1987) Intermodality: concept and practice; structural changes in the ocean freight transport industry, Lloyd's of London Press Ltd, Colchester.

Kreutzberger, E., Macharis, C. and Woxenius, J. (2006) Intermodal versus unimodal road freight transport - a review of comparisons of the external costs. In: Jourquin, B., Rietveld P. and L. Westin (Eds.), Towards better Performing Transport Systems, Taylor and Francis. London, pp. 17-42.

Macharis, C. (2000) Strategische modellering voor intermodale terminals. Socio-economische evaluatie van de locatie van binnenvaart/weg terminals in Vlaanderen (Strategic modelling for intermodal terminals. Socio-economic evaluation of location of barge/road terminals in Flanders), PhD Thesis, Vrije Universiteit Brussel, Brussel.

Macharis, C. (2004) A Methodology to Evaluate Potential Locations for Intermodal Barge Terminals: A Policy Decision Support Tool. In: M. Beuthe, V. Himanen, A. Reggiani and L. Zamparini (Eds.), Transport Developments and Innovations in an Evolving World, Springer, Berlin, pp. 211-234.

Macharis, C., and Pekin, E. (2008) Assessing policy measures for the stimulation of intermodal transport: a GIS-based policy analysis, Journal of Transport Geography, 17 (6) pp. 500-508.

Office de Promotion des Voies Navigables (OPVN) (2006) Etude du potential de transport fluvial de conteneurs le long de la dorsale Wallonne. OPVN. Liège, 30 pp.

van Klink, H.A., van den Berg, G.C. (1997) Gateways and intermodalism, Journal of Transport Geography, 6 (1), pp. 1-9.

Waterwegen en Zeekanaal NV. (2008) Seine-Schelde West : Resultaten van de haalbaarheidsstudie, brochure D/2008/3241/255

Waterwegen en Zeekanaal NV. (2010) http://www.wenz.be/Projecten/Seine_Schelde_West Accessed on 04/02/2010

USING AUCTION MECHANISMS FOR COORDINATING CONTAINER FLOWS IN INTERMODAL FREIGHT TRANSPORT NETWORKS

Edith Schindlbacher and Manfred Gronalt
Department of Production and Logistics
University of Natural Resources and Life Sciences Vienna
Feistmantelstraße 4, 1180 Vienna,
Austria
E-mail: edith.schindlbacher@boku.ac.at

KEYWORDS
Event-oriented multi-agent based simulation, intermodal transportation, decision support, network flow analysis, disruption risks.

ABSTRACT

The concept of a multi-agent simulation tool, depicting the nodes and links in an intermodal freight transport network is presented. The tool is intended to analyze network robustness and behavior, given a framework of several risk scenarios. For this purpose, auction mechanisms to coordinate load unit flows in case of rerouting demand are deployed. The risk scenarios include the consequences of operational failures, infrastructure or equipment breakdown, as well as the impact of natural hazards. The paper is organized as follows: The first section gives a short introduction to the topic. The second section provides insight to related work. In the following two sections the concept of the simulation model and proposed implementation is presented. Finally, an outline of the planned model application is given.

INTRODUCTION

Intermodal freight transport is characterized by the subsequent and alternating use of different transport modes in order to move intermodal load units from the consignor to the consignee. For the transport process, a large number of activities, actors and resources are needed, which to a certain extent implies complexity in technological, operational and organizational concerns. Inherent to complex systems is a particular degree of susceptibility to disturbance and failure. In fact, all potential incidents, e.g. ranging from loading problems at the shipper, a fault event during the transport on one of the modes or a disturbance emerging at a transshipment facility (container terminal or port) as well as problems at the last mile-transport ending at the addressees location, result in a time delay and/or a diminished capacity of either a network link or a network node.
Given such a situation, and always depending on the severity of the negative impact, a demand for rerouting of a particular amount of load units can occur. The decision on either how many items to send to any other network node, or which other network link to use, or both, depends on the willingness of the other network participants to cooperate and overtake workload. For both, link and node operators, this willingness depends on their operational strategy and on potential capacity restrictions.
Every single actor in the system has different goals to pursue. For transport link operators these goals may be to maintain maximum throughput while at the same time ensuring maximum reliability. Terminal/port operators might try to gain maximum revenue by optimizing the number of transshipments, while keeping the storage utilization at a reasonable level. Furthermore, the behavior of an actor can be contingent on the role it has to play, e.g. for a terminal operator the goals may differ if it is a gateway terminal or a local hub.
In order to plan and coordinate how to accomplish the rerouting task, a multi agent approach is proposed, depicting the network actors in charge. Thus, the communication and coordination process takes place between agents displaying the single container terminals as network nodes, and an agent, who is responsible for their corresponding network links, representing rail, road and waterway connections. The mechanisms for reallocating the individual work load are related to auction theory.

RELATED WORK

A survey of Davidsson et al. (2005) shows, that agent technology has been applied to numerous problem areas within the field of transport logistics. The authors further classified the reviewed papers according to the usage of the system, the types of agents and the applied type of coordination. The system-usage signifies, if the agents serve as an automation system, where required acts are performed by a self-acting mechanism, or a decision support system. The agent type refers either to their behavior as reactive agents, which in the most basic case are defined by a collection of independent action rules for different situations, or in contrast to deliberative agent architectures. The decision making process of the latter can be described as a sense-model-deliberate-act cycle, based upon an explicitly represented model of the world. A third type, called hybrid agent architectures, intends to integrate both,

properties for conducting routine tasks as well as more advanced and long term assignments. The dimensions of coordination are control, which can be centralized or decentralized, the structure, denoting the set of agents, their roles and ways of communication, and agents' attitude, which can be cooperative or competitive behavior. The results of the reviewed papers show that approaches settled in intermodal transport context are mainly designed as decision support systems.

In the field of supply chain management Schneeweiss (2003) contributes by identifying classes of distributed decision making (DDM) problems. The analysis structures several types of DDM systems and corresponding contributions of different scientific disciplines. Multi-agent systems, among many others, are stated to have potential for future work especially in the area of problems forming out of information asymmetries.

Based on the facts that an intermodal transport chain depends on a large number of actors and that the integration between them can be critical, Henesey and Törnquist (2002) underline the need of sharing important information which may have an influence on the daily planning of the actors. In their work they focus on the importance of collaboration between the different intermodal transportation parties. The authors propose that since the rail-port interface affects the overall performance of the total transport chain, a multi agent system simulation approach containing rail traffic, a container terminal and their interface, allows the observation of the behaviors, plans, and co-ordinations between the agents, as a basis for a decision support system.

Rizzoli et al. (2002) and Gambardella et al. (2002) introduce an agent based system combined with a discrete-event simulation software for planning the flow of ITUs using combined rail/road transport (among and) within inland intermodal terminals. The model is built in three modules: a road network planning and simulation module, a terminal simulation module and a rail corridor simulation module. The modules are designed to work parallel and interchange information. A timetable containing departure and arrival times accounts for the travel times in the rail corridor between two terminals. Additionally, a schedule for the truck arrivals is provided for each terminal. The model simulates the basic terminal internal processes (loading and unloading of ITUs, also considering storage and buffer areas), gathering performance indicators of the terminal equipment. A rail corridor is considered as an abstract representation of a path in a rail network. The train travels are simulated through the expiration of a set time.

Parola and Sciomachen (2005) analyze the impact of possible future growth of the container flows at sea on the land infrastructure in the northwestern Italian port system by means of a strategic discrete event simulation model. They give particular attention to the modal split re-equilibrium and highlight the degree of saturation of the railway lines as well as the level of congestion of the truck gates at the particular terminals in a port. The system consists of two main components, which are the ports of Genoa and La Spezia. These components are composed by different maritime container terminals which are considered as black boxes in the study, as the authors focused their attention on the logistics network.

The study considers in more detail variables in the so called macro environment (representing logistic activities and interconnection phases), using a modular structure, e.g. the rail traffic share. Its behavior results from the simulation on the basis of single micro-systems (modules), such as the berth management and the marshalling yard. The explicitly represented modules are related to the input and output flows of containers in the system, which are the ship berthing, the truck gate and the rail yard.

A simulation approach on strategic level is introduced by Caris et al. (2007). The authors propose a discrete event simulation methodology to analyze policy measures with the intent of stimulating intermodal barge transport. The intermodal transport network is divided into three components. First, the inland waterway network, comprising terminals, waterways and container flows. Entities in this system are barges carrying containers through this network. Second, the port area (in the case study the port area of Antwerp), and third, a lock planning module completes the system. The focus of the work is on inland waterway connections, rail links in the network are not taken into account. The model intends to give support in operational planning of the considered network, e.g. to analyze the potential for a barge hub or analyze different barge services.

Grundspenkis (2008) gives an insight into trends in applications of the intelligent agent paradigm in the domain of logistics. The author points out the special characteristics of different types of agent systems, which are first single agents constituting multi-agent systems, second, holonic systems, and third, multi-multi-agent systems. Further, two generic meta-types of agents, management agents intended to achieve their goals with regard to their environment and given action space, and service agents solving well defined tasks autonomously, are introduced. Moreover, a prototype of a simulation tool for a freight delivery supply chain, based on holonic agents using auctions to make deals between clients and logistic companies, is used to investigate the influence of different auction protocols on the price for the transport in a specified supply chain.

Conitzer (2010) describes different settings for decision making based on the preferences of multiple agents. Given settings without payments, one approach to decide among several alternatives is either to vote over them or to rank them. In this field, research on combinatorial voting, which is decision making on a number of interrelated issues, is at the very beginning. Methods to task and resource allocation without payments seek to get a "fair" result of allocation. Introducing payments approves to quantify the preferences and to transfer utility from one to another

agent. For this purpose, exchange and auction mechanisms are applied. This approach provides the benefit, "that the resource ends up with the agent who values it the most (or the task ends up with the agent who minds doing it the least)", and thus achieving an efficient allocation.
A multi-agent approach for simulating cooperative recovery behaviors in an inner-company context is proposed by Cauvin et al. (2009). A general framework for disruption management is presented, which aims at supporting decision making in distributed industrial systems facing disruptions.
Though considerable research has been done in the field of supply chain uncertainty and risk, little direct attention has been paid to transportation as supply chain activity (Sanchez-Rodrigues et al. 2008). Presenting a supply chain uncertainty model focusing on transport operation, the authors consider transport uncertainty with regard to variations of several factors as transit times, schedules, volume or also mode. It is shown very well, how transportation uncertainty holds a (considerable) share in supply chain risks.
Folga et al. (2009) present a methodology to analyze effects on waterway commodity flows and transportation costs due to infrastructure disruptions. The methodology resolves the mode and route of transportation, based on general cost of the transportation modes, incorporating infrastructure interdependencies and cascading impacts. The rerouting options include a modal shift to rail or road.

CONCEPTUAL MODEL

The overall aim of the presented model is to analyze the flow of load units through and within the considered network, with focus on changes of these flows, emanating from a framework of disruption risks which is developed on the basis of former work (see Gronalt et al. 2008, Schindlbacher et al. 2009).
Conceptually, the simulation model consists of 4 different levels. In the first level the *physical network* is defined by rebuilding the physical components (intermodal container terminals, roads, railway links, waterway links) of the analyzed network. The second level implements the *service network*, which specifies the operational parameters of the physical elements, e.g. the terminals opening hours or the offered train connections and schedules between them. The criteria defining the *flow of the load units* in the network are specified in the third modeling step, and finally, the fourth level implements the disruption scenarios to be analyzed.

Physical network

A network is constituted through nodes and links. In this case the network nodes are either container terminals or so called "junction nodes" which have no other functionality than connecting two or more rail network links. The terminal nodes represent the single container terminals performing lifting and storage functionality according to their individual characteristics. Each terminal node has a corresponding catchment area, where load units enter or leave the network by truck. This area is specified by the average transport distance and average transport time to the terminal. Further, each terminal node has assigned at least one residual area which aggregately depicts other connected container terminals which are external to the investigated network and therefore not explicitly modeled. They are connected through railway links, and also function as sources and sinks of load units for the evaluated network.
The links of the network are the connections between the terminal nodes and the sources and sinks of load units by road and rail. By now, waterways are not considered in the model. Topologically, network links are defined through the mode they represent, their direction and distance.

Service network

The service network provides the network topology with parameters defining the transport service the network "offers". There are several roles a terminal can fulfill in the network, marked by different characteristics like transshipment and storage services, capacities and priority rules, or also the opening hours for truck and train acceptance and hours of internal operation.
Furthermore, the transportation potential within the network is determined, respectively limited, by the capacity and utilization specifications of road and rail links. Here the travel time, which is partly dependent on the utilization and status of every single link, is one important factor influencing the load unit throughput.

Load unit flows

As already mentioned, the locations of the network inflow are given through the catchment areas of the terminal nodes and the residual areas. The itinerary of a load unit through the network is a result of the service network and according decisions made in the terminal nodes. On the one hand, these decisions depend on terminal internal parameters, like the share of direct moves, the assigned storage times, transshipment distribution between the modes, or the lifting capacities (which are e.g. further reluctant on services moves). On the other hand, the train schedule and truck arrivals determine the transshipment requests to be served. The combination of these factors determines the proceeding of every load unit in the terminal and the network.
The load units keep records about the network path that has already been used. This serves two reasons. First, to ensure that they can't be sent back (with exception in case of a disruption), and second, to collect data for network performance analysis. When a load unit is sent from a terminal to the catchment area or residual area, it

leaves the network and the totally covered distance and time in the network is calculated.

3.4 Risk concept

There are two types of sources of disturbance in the intermodal transport chain, i.e. the failure is of either network component (node or link) internal or external origin. Internal problems at a node or link can be caused by irregularities in the operational processes or the malfunctioning of equipment or infrastructure. Negative impact from outside the system, thus from external sources, can be triggered by natural hazards, influence from the surrounding infrastructure, but also by adverse human behavior.
Independent of the source of origin, failures in one network component potentially influence the other network nodes or links in a negative way. The consequence of an incident always is a delay and/or a diminished capacity of either a network link or a network node.

IMPLEMENTATION OF THE MODEL CONCEPT

Model components and agent structure

The simulation model contains several components:

- terminal process modules
- terminal agents
- link agent
- catchment area agents
- residual area agents
- environment agent
- administrator agent.

The representation of the terminal nodes is realized through two different elements. First, for each node a module processing the transshipment and storage functionality of the specific container terminal evaluates, if the planned tasks can be completed. Based on this outcome, the corresponding terminal agent can decide whether coordination with the other network components becomes necessary, or not. Steps to be taken include providing information to the others about e.g. delays in service or problems in accepting load units to transship, or to request the rerouting of load units. The announcement of information addresses to all network agents except the environment agent and the administrator. Rerouting requests only affect to the terminal nodes, thus the terminal agents.
One link agent is responsible for the management of all network links. This means that it routes the trains and trucks through the network and checks for feasibility in case of rerouting demand. Additionally, it has several strategies to react on faced disruptions.
The agents of every catchment area and the different residual areas can induce rerouting requests when the regular network inflow is blocked due to a disruption in the network path to the destined terminal, or in case an acceptance problem is announced. Both types of “areas” can’t be involved in a rerouting process since their only function is to provide for network inflow and outflow of the load units, hence they do not need transshipment functionality.
The risk concept is on the one hand implemented by the network components themselves (internal origin), and on the other hand by the environment agent which induces the disturbances with external origin.
The first job of the administrator is to structure the simulation run. Secondly, it is applied to avoid problems due to concurrency issues by controlling and leading the agents’ communication.

Auction mechanisms to coordinate rerouting demand

As already mentioned, the actors to be in charge of deciding in the rerouting process, are the agents representing the container terminals and one agent, who is responsible for all network links. After an incident affected the network to a severity that a rerouting becomes necessary, several options regarding the question how this will take place are provided. A task to bid and decide on is always a batch of load units, containing enough items to form a train.
The rerouting process (actors involved, initiating actor and the actor who is in charge of the decision) can vary for the different disturbance scenarios, respectively their impact on the network. This is for example, that certain network actors will be excluded from the rerouting process a priori, as if the disturbance occurs at one network link, nodes only interconnected by this one are out of the game per se. Further, if the incidence strikes a network link, the link-agent has to inform the terminal-agents about the consequences (changing initiator).
The decision parameters to reflect preferences and utilities of the network actors which can be applied to decide on a rerouting possibility are capacities or time windows of nodes and links, or also the costs for using a certain network element. Which parameter, or combinations of them, are chosen, depends on the expectations to the desired result. One can ask for the best feasible solution for the initiating agent, i.e. one specific terminal, or in contrast, for the whole network, initiating a process that can be compared to a reverse auction. Reverse auctions do not intend to allocate a resource, but try to assign a task. So, a bid on a task indicates how much the bidder wants to be rewarded for carrying it out. Obviously, in this case the lowest bid wins.
In the case of decisions regarding the individual interests of a terminal, the only decision criterion may be the time window, and in the case of a network oriented solution the total costs for using other links and nodes. But also, a multicriteria auction may be used, where an agent decides on both, feasible time window and cost factor. All variants have in common that a single-bid approach will be used. The rerouting process

follows the structure of a (reverse) sealed-bid auction. The characteristics of this format are that all bidders place one offer at the same time, representing a tuple of a time window and costs of accepting, while not having information about the bids of their competitors. The auctioneer then decides for the best choice.
For both, the rerouting request and the accordant offers, different sources of information are available. Naturally, every agent has its own information basis containing its beliefs and goals (Bagherzadeh and Arun-Kumar 2006). Here, to give an example, all directly connected nodes will be listed. But also a general information store can be introduced to share knowledge, e.g. all existing network participants and their connections parameters.
Considering the above mentioned system characteristics, a versatile communication framework has to be developed in order to enable the use of auction mechanisms to solve the problem of rerouting demand in intermodal freight transport networks. This tool is intended to help to draw conclusions on how to implement suitable strategies to handle disturbance events and provide for a robust network planning.

FUTURE RESEARCH OUTLINE

Future research plans involve the application of the model to the network of Austria's main inland container terminals, including the other network components according to the description of the physical layer:

- main rail connection between the terminal nodes
- one (aggregated) main road connection per catchment area and corresponding terminal node
- presumably 3-4 residual area nodes for the main foreign import/export destinations.

The simulation experiments intend to analyze potential capacity bottlenecks for different settings of the service network combined with variations in transport demand. We assume that the results of additionally implemented disruption scenarios will provide insight into the structure of reliable partners for coping with adverse situations, as well as show the ability of the network to recover from shocks.
Regarding the decision rules for rerouting processes, different threshold values will be used for defining the initiation of a request, as well as the acceptance criteria used by the respondents. Further, variations in the degree of cooperativeness of the single network components will provide different levels of detail in information to the others, and may result in changes in the network behavior.

REFERENCES

Bagherzadeh, J. and S. Arun-Kumar. 2006. "Flexible communication of agents based on FIPA-ACL." *Electronic Notes in Theoretical Computer Science* 159, 23-39.

Caris, A.; G.K. Janssens; and C. Macharis. 2007. "A simulation approach to the analysis of intermodal freight transport networks." In *Proceedings of the European Simulation and Modelling Conference* (St. Julians, Malta, Oct 22-24), EUROSIS, 283-286.

Cauvin, A.C.A.; A.F.A. Ferrarini; and E.T.E. Tranvouez. 2009. "Disruption management in distributed enterprises: A multi-agent modelling and simulation of cooperative recovery behaviours." *International Journal of Production Economics* 122, No. 1, 429-439.

Conitzer, V. 2010. "Making decisions based on the preferences of multiple agents." *Communications of the ACM* 53, No. 3, 84-94.

Davidsson, P.; L. Henesey; L. Ramstedt; J. Törnquist; and F. Wernstedt. 2005. "An analysis of agent-based approaches to transport logistics." *Transportation Research Part C* 13, 255-271.

Folga, S.; T. Allison; Y. Seda-Sanabria; E. Matheu; T. Milam; R. Ryan; and J. Peerenboom. 2009. "A systems-level methodology for the analysis of inland waterway infrastructure disruptions." *Journal of Transportation Security* 2, No. 4, 121-136.

Gambardella, L.M.; A.E. Rizzoli; and P. Funk. 2002. "Agent-based planning and simulation of combined rail/road transport. *SIMULATION* 78, No. 5, 293-303.

Gronalt, M.; H. Häuslmayer; W. Jammernegg; E. Schindlbacher; and M. Weishäupl. 2008. „A risk assessment approach for inland container terminals." In *Proceedings of the 11th International Workshop on Harbor, Maritime & Multimodal Logistics Modeling & Simulation* (Campora S. Giovanni, IT, Sep 17-19). 251-256.

Grundspenkis, J. 2008. "Intelligent Agents in Logistics: Some trends and solutions." In, *Proceedings of the 11th International Workshop on Harbor, Maritime & Multimodal Logistics Modeling & Simulation* (Campora S. Giovanni, IT, Sep 17-19). 174-179.

Henesey, L. and J. Törnquist. 2002. "Enemy at the gates: Introduction of multi-agents in a terminal information community." In *Maritime Engineering & Ports III.* C.A. Brebbia and G. Sciutto (Eds.). London: WIT Press, 23-32.

Parola, F. and A. Sciomachen. 2005. "Intermodal container flows in a port system network: Analysis of possible growths via simulation models." *International Journal of Production Economics* 97, 75-88.

Rizzoli, A.E.; L.M. Gambardella; and P. Funk. 2002. "A simulation tool for combined rail/road transport in intermodal terminals." *Mathematics and Computers in Simulation* 59, 57–71.

Sanchez-Rodrigues, V.; D. Stantchev; A. Potter; M. Naim; and A. Whiteing. 2008. "Establishing a transport operation focused uncertainty model for the supply chain." *International Journal of Physical Distribution & Logistics Management* 38, No. 5, 388-411.

Schindlbacher, E.; M. Gronalt; and H. Häuslmayer. 2009. "Multi-Agent Simulation for Analysing Inland Container Terminal Networks." In *Proceedings of the 12th International Workshop on Harbor, Maritime & Multimodal Logistics Modeling & Simulation* (Puerto de la Cruz, ES, Sep 23-25). 140-145.

Schneeweiss, C. 2003. "Distributed decision making in supply chain management." *International Journal of Production Economics* 84, No. 1, 71-83.

THE IMPACT OF FUEL PRICE INCREASES ON INTERMODAL TRANSPORT COMPARED TO THE INTERNALISATION OF EXTERNAL COSTS

Ellen Van Hoeck
Cathy Macharis
Ethem Pekin
Tom van Lier
Vrije Universiteit Brussel
Department MOSI-Transport & Logistics
Pleinlaan 2
B-1050, Brussels, Belgium
E-mail: Ellen.Van.Hoeck@vub.ac.be

KEYWORDS
Intermodal transport, fuel price, external costs, Location Analysis, GIS network model

ABSTRACT

This paper aims to determine whether an increase in fuel prices is sufficient enough to raise the market area of intermodal transport to the same degree that would be accomplished by stimulating intermodal transport through policy instruments, more specifically an internalisation of external costs. In a first step, several fuel price scenarios are therefore analysed in order to verify the impact of different fuel price evolutions on the market area of unimodal road transport compared to intermodal transport in a Belgian context. This analysis is performed using the LAMBIT-model (Location Analysis for Belgian Intermodal Terminals), which is a GIS-based model (Macharis and Pekin, 2008). With this model the different fuel price increases can be analysed and a visualisation of the impact on the market area of the different transport modes is enabled. In a second step, the LAMBIT model is used to analyse the effect of an internalisation of external costs in order to compare this policy measure with the effect of a fuel price increase. Results of this comparative analysis show that internalisation seems more effective than expected fuel price increases.

INTRODUCTION

The increasing importance of door-to-door and just-in-time services has led to a strong sustained growth of road transport (European Commission, 2006). The modal share of rail transport is very small for the EU-27. Also in Belgium this is the case, since 78% of goods is transported via road, while rail transport and inland waterways account for respectively only 10% and 12% (Universiteit Antwerpen *et al.*, 2008). To stimulate a more sustainable transport, the European Commission has developed policy measures to shift the balance between transport modes with special focus on promoting intermodal transport.

Various combinations of policy instruments can be formulated along the intermodal transport chain. E.g. Member States can formulate *subsidy schemes* in order to promote the growth of intermodal transport and this is one of the most common policy measures in this area (Macharis et al., 2008). Another option which leads to a more efficient use of transport infrastructure is the *internalisation of external costs* (Macharis, Pekin, van Lier, 2010). Ricci and Black (2005) measured the social costs of intermodal transport, thus focussing on both the private and external costs, and conclude that full internalisation of external costs will benefit intermodal transport.

The aim of this paper is to determine whether an increase in fuel prices would be sufficient enough to raise the market area of intermodal transport to the same degree that would be accomplished by stimulating intermodal transport through internalisation of external costs of transport. This paper focuses on containerised transport, with focus on the transportation costs, which are mainly determined by fuel prices, the efficiency of the transportation mode and taxes. Other major factors influencing the attractiveness of containerised transportation such as time in transit and reliability of transit time remain outside of the analysis.

In order to verify the effect of an increase in fuel price on the market area of intermodal transport several price scenarios will be analysed by using the LAMBIT-model (Location Analysis for Belgian Intermodal Terminals). The LAMBIT model can also be used to evaluate the impact of different policy measures on the market area of intermodal transport so the impact of an internalisation of externals costs can also be performed using this model.

In section 2 the impact of fuel prices on the price of intermodal and road transport is given. Section 3 shows the methodology and the results of the LAMBIT-model. Finally conclusions are drawn and future research perspectives are presented in section 4.

IMPACT OF FUEL PRICE ON THE PRICE OF BOTH INTERMODAL AND ROAD TRANSPORT

Freight transport consists of different activities that are linked to both internal and external costs. Internal costs are those incurred by a transport operator and contain various components such as personnel, fixed assets, energy, stock return, time, organisation costs and insurance, taxes and charges (Ricci and Black, 2005). In our analysis, focus will be on the energy costs part of internal costs. In the second and third subsections, the break-even distance and internalisation of external costs are introduced.

Energy Cost Framework

The energy costs are connected to the retail fuel price. For Belgium, the main direct factors influencing the energy costs are the international Brent price and national petroleum taxes (excise duties and VAT).

Brent Price

In their Annual Energy Outlook 2009, the EIA (Energy Information Administration, 2009) developed three different future scenarios for the Brent price, namely a low price case, a business as usual case and a high price case. The forecast for 2030 expressed in $/barrel was converted to the diesel price in Belgium, taking into account that the crude oil price effects the retail fuel price for only 40% (Belgian Petroleum Federation (BPF)). This $/barrel assumptions were converted in euro using an exchange rate between dollar and euro set at 1,1 in accordance with the forecasts of EIA and BPF. Table 1 shows the effect of increases in the retail fuel price according to three scenarios forecasted by the EIA (2009).

Table 1: Effect of an Increase in Retail Fuel Price according to Three Scenarios

	Scenario 1 low price case	Scenario 2 business as usual case	Scenario 3 high price case
Crude oil price ($/barrel)	50$	130$	200$
Increase vs current price	↑ 30%	↑ 160%	↑ 300%
Increase in diesel price	↑ 10%	↑ 50%	↑ 90%

National Petroleum Taxes

On January 6th of 2010 nearly 35% of the fuel price excl. VAT and 53% of the fuel price incl. VAT consisted of excise duties and energy contributions. These excise duties are set by the federal government and are unaffected by a change in the Brent price. Furthermore the retail fuel price is subject to a VAT of 21%. A change in Brent price consequently affects the retail fuel price for only 40% (BPF).

The diesel tax levels for road transport are lower than for road transport in France and in the Netherlands. For Belgium the diesel tax for road transport amounts to 0,362€ (BPF). In Belgium inland shipping is exempted of fuel duties (NEA, 2008) and also for rail transport no diesel tax is charged (Infrabel).

Price of Electricity

According to the European Commission (2008) 84% of the Belgian railway network (including freight and passenger transport) was electrified in 2005. According to B-Cargo 30% of all rail freight transport uses diesel traction (both national and international rail transport).

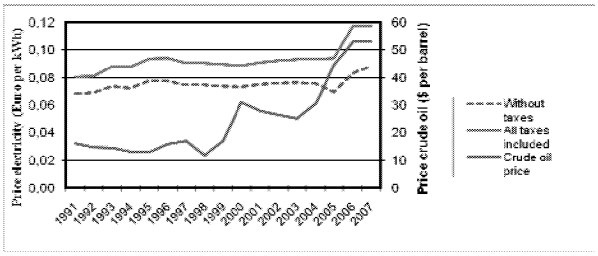


Figure 1: Price of Electricity compared to Price of Crude Oil

Figure 1, where the price of crude oil is compared with the price of electricity in Belgium, clearly shows that the price of electricity is less volatile then the price of crude oil. The oil price increase in 1999-2000 had no impact on the price of electricity and during the crude oil price increase in 2005 the price of electricity rose as well but to a much smaller degree. An outspoken increase of the Brent price will therefore only have a small effect on the cost of power source for rail transport. The price of electricity is also determined by other factors for example the liberalisation of the energy market, the internalisation of the external costs of electricity, etc. (Ecorys, 2006). No electricity tax for rail transport is applied in Belgium (Infrabel).

Break-even Analysis

In transport theory, intermodal transport offers an alternative to unimodal road transport from a certain distance, called the 'break-even distance' (Vrenken et al., 2008). This is due to the fact that the scale advantage of intermodal transport only starts counting when the costs of transhipment and terminal handling costs have been offset. So intermodal transport only

becomes a competing mode once the 'break-even distance' is reached.

A fuel price increase affects the variable cost of both unimodal and intermodal transport. Figure 2 visualises the impact on the break-even analysis. Intermodal transport becomes more competitive but this is tempered to some degree as the cost of initial and end road haulage also rises. In section 3 the effect on the break-even distance will be calculated with the LAMBIT-model.

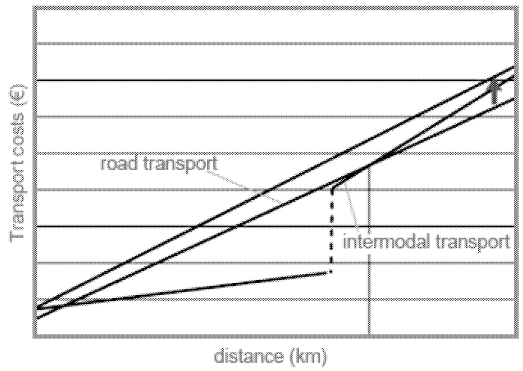


Figure 2: Break-even Analysis between Road and Intermodal Transport

External Cost Framework

In order to be able to compare the effect of a fuel price increase on the market share of intermodal transport with the effect of a policy measure on the modal share of intermodal and unimodal transport, the internalization of external costs was chosen as the policy measure to be analysed. In the following section the concept of externalities, external costs and internalization of these costs is briefly presented.

Externalities are changes in welfare caused by economic activities without these changes being reflected in market prices (Weinreich et al., 1998). In the field of transport these externalities arise when transport consumers/producers impose additional costs on society without having to bear these costs themselves. External costs are externalities expressed in monetary terms.
In economic literature, the most important external costs of transport are (Infras/IWW, 2004):

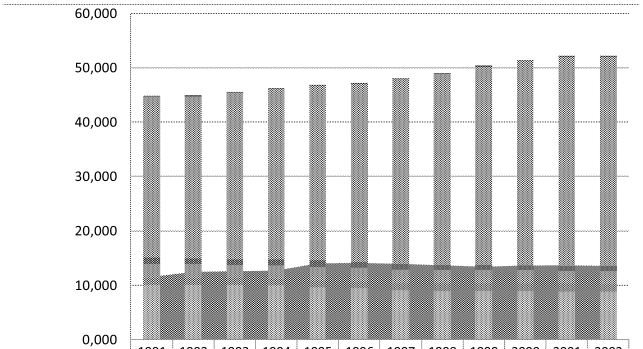
- Accidents;
- Noise;
- Air pollution;
- Climate change;
- Congestion;

Calculation of the relevant external costs in this specific case is based on best practices in the field of external cost assessment currently available in scientific literature. (Maibach et al., 2008) Although there is growing consensus on the main methodological issues, there remain many uncertainties when performing an external cost assessment in practice. Marginal external costs of transport activities depend strongly on parameters such as fuel type, location (urban, interurban, rural), driving conditions (peak, off-peak, night) -and vehicle characteristics (EURO standards) (Int Panis and Mayeres, 2006). As a result, the external cost of one truck-kilometer in urban areas during peak traffic can be up to five times higher than the cost of an off-peak inter-urban kilometer of the same vehicle (Maibach et al., 2008).

Since, in the context considered here, calculation of the societal cost of transport to and from the intermodal terminals on existing transport infrastructure (roads, rail and canals) is required, focus will be on short run marginal external costs. Short run marginal external costs of transport are related to an additional vehicle entering the (existing) system, and consider only variable costs (i.e. costs depending on traffic volume), neglecting fixed costs to run the system or additional costs for possible network improvements in the longer run. If long term policy measures would however include building new transport infrastructure to connect intermodal terminals, long term marginal external costs need to be added to account for the external costs caused by new roads, rails and/or canals. External costs on the intermodal terminals itself (e.g. the external cost caused during transhipment) are not yet included in the model.

Since intermodal transport incorporates modes such as barge, rail and short sea shipping that have lower external costs in most of the trajectories (see for an overview of studies Kreutzberger et al., 2006), various European transport policies aim to initiate a shift of freight from unimodal road transport to more sustainable intermodal transport. In 2007, the European Commission (EC) announced a European freight transport action plan, including the introduction of the concept of "green transport corridors". Green transport corridors include shortsea shipping, rail, inland waterways and road transport combinations to enable environmentally friendly transport solutions for the European industry. Also a partial internalisation of external costs of road freight transport through the Directive on the charging of road transport for infrastructure use (Eurovignette) is currently under revision. These measures are planned to come into effect before 2011.

External costs of road transport depend highly on the location, time and vehicle type. As a result, most external cost categories show significant country-related differences. Therefore use was made of the external cost figures for Belgian road transport from De Ceuster (2004). This study calculates values on a Belgian level (more specifically the Flemish part of Belgium) and takes into account the five most important short term marginal external cost categories: air pollution, climate change, accidents, noise and congestion. In addition, the short term marginal external cost of damage to the road, caused by additional trucks on the road, is also taken into account (MEC road in Figure 3). The study also takes into account the effects of taxation of road transport (including excises and VAT on fuel, traffic taxes, taxes on insurance premiums and on maintenance of vehicles, Eurovignette, vehicle purchase taxes and registration taxes) in order to determine which part of the external costs is already internalized. This is very useful since it allows taking only the part of non-internalised external costs into account. Figure 3 shows values for the marginal external costs and taxes for a heavy duty diesel truck for Flanders over the period 1991-2002. In 2002 total marginal external costs amounted to 0,52€/km.



	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002
TAXES	11,54	12,45	12,57	12,68	14,01	14,08	13,93	13,66	13,38	13,62	13,64	13,52
MEC ROAD	0,14	0,14	0,14	0,14	0,14	0,14	0,14	0,14	0,14	0,14	0,14	0,14
MEC CONGESTION	29,57	29,84	30,59	31,37	32,16	32,90	34,08	35,11	36,45	37,61	38,63	38,55
MEC ACCIDENTS	1,28	1,11	1,05	1,09	1,18	0,95	0,97	0,89	0,98	0,92	0,78	0,84
MEC NOISE	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45
MEC CLIMATE	1,30	1,30	1,32	1,32	1,32	1,33	1,32	1,33	1,33	1,33	1,34	1,34
MEC AIR	10,07	10,06	10,02	9,88	9,61	9,45	9,13	9,04	9,01	8,95	8,90	8,86

Figure 3: Marginal External Costs versus Taxes - heavy truck diesel (in €/100vkm)

Note the high and increasing proportion of congestion costs over the years, accounting for 74% of total short term marginal external costs in 2002, whereas the other external cost categories remained stable or gradually decreased. Such high marginal external congestion costs are confirmed by other authors (Hertveldt et al., 2009) As can also be seen from Figure 3 (the shaded area), the existing taxation system on heavy diesel trucks compensated for 26% of short term marginal external costs in 2002, leaving 74% of external costs non-internalized.

ANALYSING FUEL PRICE INCREASES VERSUS INTERNALISATION OF EXTERNAL COST WITH THE LAMBIT MODEL

Methodology

LAMBIT is a geographic information system (GIS)-based location analysis model which makes it possible to do ex-ante and ex-post analysis of policy measures in favour of intermodal transport. Starting from a reference scenario which includes all the existing intermodal terminals and current market prices and that serves as a benchmark, different policy measures can be analysed such as the location of new terminals, the application of particular price scenarios, the granting of subsidies and the internalisation of external costs.

Construction of the Model

Three main inputs form the core of LAMBIT: transportation networks, transport prices and container flows from the municipalities to and from the port of Antwerp.

1. Transportation Networks

LAMBIT is a GIS-based model, including four different layers: the road network, the rail network, the inland waterways network and the final haulage network. The geographic locations of the intermodal terminals, the Port of Antwerp and the municipality centres are defined and connected to the different network layers (Macharis, 2000 and 2004).

Figure 4 depicts the different network layers and nodes including nine inland terminals and four Narcon (National Rail Container Network) rail terminals.

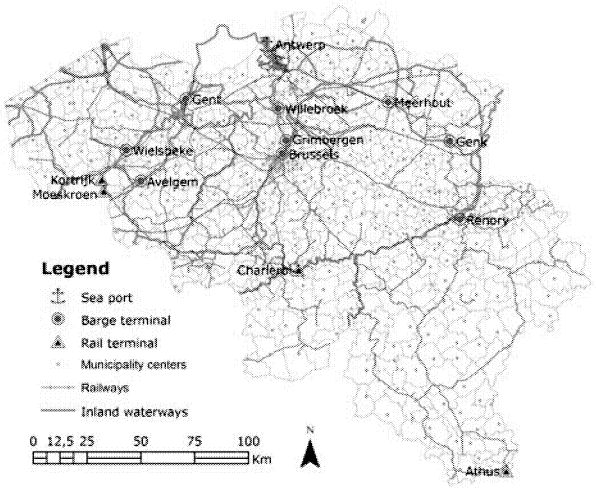


Figure 4: Network Layers and Nodes

2. Transport Prices

The LAMBIT methodology is based on two concepts: the intermodal cost structure and the break-even distance. This implies that unimodal road transport is cheaper in the short distances but intermodal transport becomes a competitive alternative once the break-even distance is achieved.

The transport prices are calculated based on the existing market price structures for each transport mode. The variable costs are uploaded to the network layers and the fixed costs are attached to the nodes. These nodes also indicate the origin and destination for each path.

The calculation of the price of intermodal transport is explained by following formula:

$$IT = PH + TH + MH$$

In which:

- IT: price of intermodal transport
- PH: price of pre/post haulage by road transport
- TH: price of terminal handling in intermodal terminals
- MH: price of main haulage by barge or rail transport

3. Containers from the Belgian Municipalities

As a final input the container flows from the port of Antwerp are added to the LAMBIT model by using the statistics of road transport from the Directorate-general Statistics and Economic Information of Belgium.

Functionalities of the Model

Various comparisons are conducted in order to find the shortest path and the attached transport price from the port of Antwerp to each Belgian municipality both via intermodal terminals and via road only, using a shortest path algorithm in ArcInfo. For each destination, the cheapest option is selected based on a comparison of the total transport price for unimodal road, inland waterways/road and rail/road transport. The market areas of each inland terminal are highlighted in the map of the model. Additionally, the container flows data can be used to show the amount of containers that are currently transported by road to the municipalities within the market area, giving an indication of the existing potential volume that can still be shifted. This is particularly useful when the potential of a new terminal location needs to be analysed.

Fuel Price Scenarios

Three fuel price scenarios based on the forecasts of EIA are analysed with the LAMBIT model. These fuel price increases have an impact on the variable costs of transport modes.

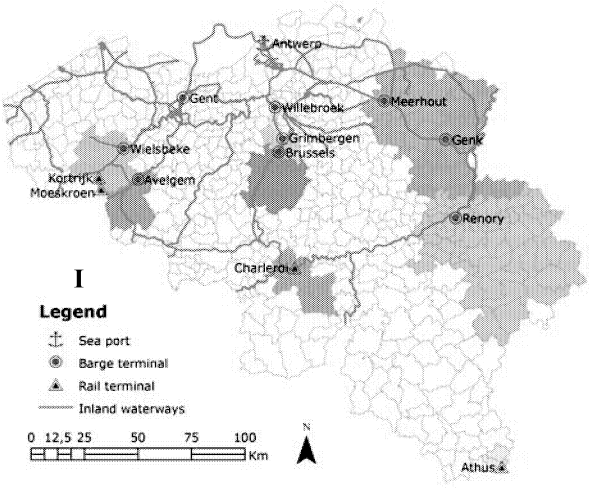
The reference scenario (map I) in figure 5 presents the existing intermodal inland terminals with their market areas. When intermodal transport has a more attractive transport price compared to unimodal road transport based on the current market prices, the municipalities are highlighted, with the green areas representing the market area of intermodal waterway terminals and the red/orange areas that of the intermodal rail terminals. Nine barge/road (inland waterways) terminals and four rail/road terminals are included in this reference scenario.

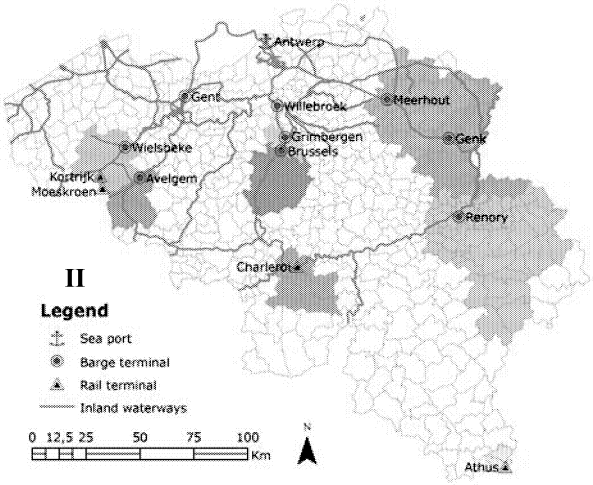
The low price scenario (map II) in figure 5 demonstrates the impact in the terminal landscape of a gradual retail fuel price increase of 10% due to an increase of the Brent price of 30%. If the fuel price increases to such a small degree, the impact on intermodal transport will be mixed because the price of pre/post haulage increases while the additional price effect on the long haul remains relatively limited. This 10% increase in fuel prices therefore results in a minor decrease in the market areas of barge terminals, with the terminals in Genk and Renory losing respectively 3 and 5 municipalities to unimodal road transport, while in contrast, the terminals in Brussels and Meerhout increase their market areas by 1 municipality each. Minor growth in the market areas of the rail terminals in Athus and Charleroi is observed. Not surprisingly intermodal rail transport cannot compete with unimodal road transport on a short distance.

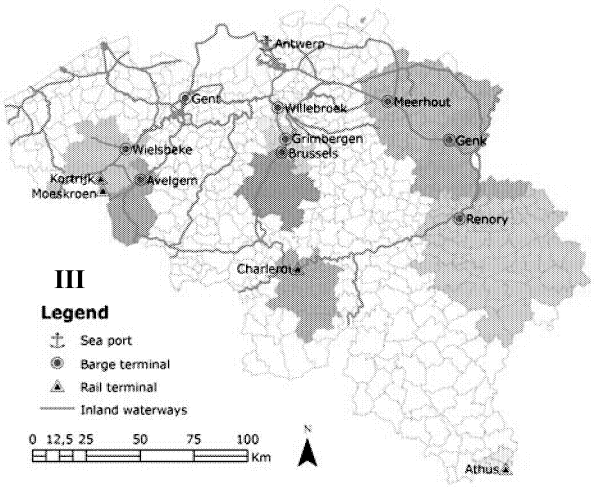
When the fuel price is increased by 50% (map III), unimodal road transport however begins to lose market

area more clearly, with almost all of the barge terminals increasing their market areas. Only the terminal in Ghent is still not able to compete with the prices of unimodal transport. The intermodal terminals in Wielsbeke and Meerhout however increase their market areas by more than 5 municipalities, and the market areas of the rail terminals in Charleroi and Athus show a significant growth. Theoretically a modal shift will occur when the retail fuel price increases with 50%.

Finally, the high price scenario (map IV) represents a 90% increase of fuel price which also not surprisingly leads to larger market areas for the barge terminals, with unimodal road transport losing 35 municipalities to the barge terminals. The expansion of the market area for inland waterways transport is most outspoken for the terminals in Wielsbeke and Brussels, where the terminals increase their market areas by 10 and 7 municipalities respectively. Also the market areas of the rail terminals are altered: in addition to the growth in the market areas of Charleroi and Athus, the rail terminals in Kortijk and Moeskroen are also able to attract some market area.







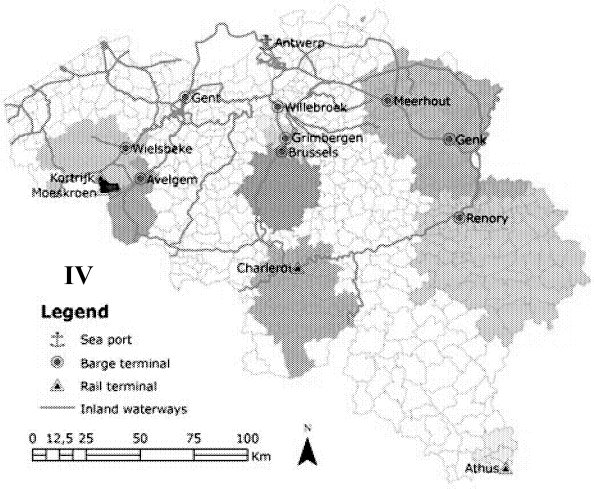


Figure 5: Fuel Price Scenarios

Internalisation of External Costs

In a next step, the effect of a fuel price increase was compared with the policy measure of internalising the external costs, by adapting the cost functions of the LAMBIT model. Current road taxes were subtracted from the total external costs for road since these taxes can be viewed as a partial internalisation. No taxes were applied to inland waterways and rail transport. Table 2 shows the average values for marginal external costs for each transport mode used in LAMBIT, expressed in € per TEU, including all costs related to the usage of transport infrastructure such as accident costs, noise, air pollution, climate change and congestion (§ 2.3).

Table 2: The marginal external costs in €/TEU

Year	Road	Rail	Barge
2002	0.39	0.07	0.06

Figure 6 shows the impact of an internalisation of external costs for all modes of transport, based on current market prices. No subsidies for rail and inland waterways transport are taken into account. The analysis indicates an increase in the market areas of intermodal terminals but the impact differs for rail and barge terminals because of the difference in the external costs for each transport mode.

The major change occurs in barge terminals: the terminals in Genk, Willebroek and Grimbergen face a moderate growth in their market areas, while the terminals in Brussels, Renory and Wielsbeke experience a considerable growth. The terminal in Gent has the potential of acquiring up to 10 municipalities by offering cheaper transport prices compared to the unimodal road transport. Considering the rail terminals, the terminals in Athus and Charleroi would increase their market areas, reaching now municipalities which were formerly in the market area of the unimodal road transport. Nevertheless, an internalisation of external costs does not lead to any change in market area for the rail terminals in Wielsbeke and Avelgem.

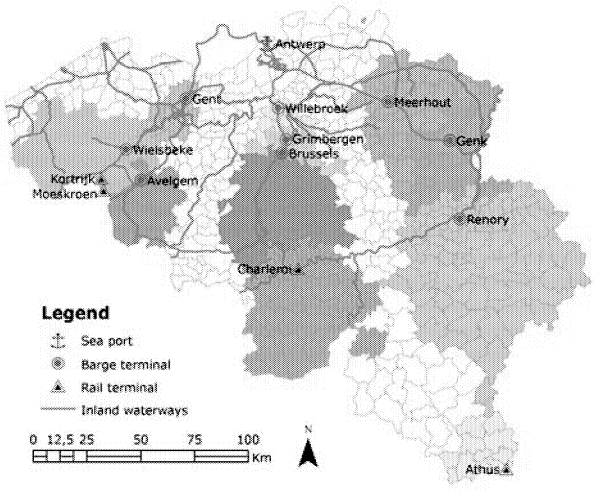


Figure 6: Internalisation of External Costs

Overall, it can be concluded that a fuel price increase is beneficial from a sustainability viewpoint, but to a more limited effect than a full internalisation of external costs since the impact of an internalisation is larger than the effect of an increase in fuel price. Even if the fuel price is almost doubled, the impact of a full internalisation of external costs remains larger. A strong stimulation of intermodal transport would thus require the application of (additional) policy instruments. The political acceptability of a full internalisation of external costs is however the subject of strong debate.

CONCLUSIONS AND FUTURE STEPS

The LAMBIT model makes it possible to make ex-ante and ex-post analysis of policy measures to stimulate the intermodal transport market. Based on the current market prices for each transport mode, this geographic information system (GIS)-based location analysis model compares intermodal transport with unimodal road transport. Based on forecasts of the Energy Information Administration future oil price scenarios are set up and converted to diesel prices in order to evaluate fuel price increases on the market area of intermodal transport.
Analysis of various fuel price scenarios by means of the LAMBIT model shows an increase in the market areas of intermodal terminals. Both intermodal barge and rail terminals increase their market areas if the fuel price increases, but the result differs depending on the size of the increase. If the fuel price increases with a small degree, the impact on intermodal transport is mixed because the price of pre/post haulage increases while the additional price effect on the long haul is relatively limited. When the fuel price increases however more significantly, the break-even distance becomes significantly smaller due to the stronger price advantage for intermodal transport on the long haul.

Also the internalisation of external costs is analysed with the LAMBIT-model. Comparative analysis shows that even when the fuel price is doubled, it cannot compete with the effect of a full internalisation of external costs on the market area of intermodal transport. It is therefore the responsibility of the policy makers to additionally stimulate the use of intermodal transport via policy measures making intermodal transport more attractive as it has lower societal costs. Although the European Commission is a strong advocate for internalisation, the political viability of a full internalisation of external costs is however less clear.

Further research should focus on refinements in analysing intermodal versus unimodal transport by including other transport decision variables such as service, transport time, generalised costs, etc. in the model and to expand the LAMBIT framework towards markets in Benelux or Europe. Modal choice does not only depend on price as single factor (although it often remains the critical factor), but depends also on other factors like transport time, reliability, flexibility, etc.

REFERENCES

De Ceuster, G., 2004. *Internalisering van externe kosten van wegverkeer in Vlaanderen.* Study performed for Vlaamse Milieumaatschappij, MIRA, Transport & Mobility Leuven, Leuven, Belgium.

Ecorys, 2006. "Analysis of the impact of oil prices on the socio-economic situation in the transport sector", Economic study for the Europian Commission (DG TREN), Rotterdam, The Netherlands, 27 april 2006, 123 blz.

Energy Information Administration (EIA), 2009a, "Annual Energy Outlook 2009: Early release overview", report, Office of Integrated Analysis and Forecasting, U.S. Department of Energy, Washington DC januari 2009, <http://www.eia.doe.gov/oiaf/aeo/pdf/overview.pdf>.

European Commission, 2006. *Keep Europe Moving. Mid-term review of the European Commission's 2001 transport White Paper*. Luxembourg: Publications Office of the European Union.

European Commission, 2008. *EU Energy and Transport in figures: statistical pocketbook 2007/*2008, Luxembourg: Office for Official Publications of the European Communities, 212 pp.

Hertveldt, B., Hoornaert, B. and Mayeres, I., 2009. *Langetermijnvooruitzichten voor transport in België: referentiescenario*, Federaal Planbureau, Planning Paper 107, Brussels, Belgium

INFRAS/IWW, 2004. *External Costs of Transport: update study*. Karlsruhe/Zürich/Paris: the International Union of Railways (UIC)

Int Panis, L. and Mayeres, I. (2006) Externe kosten van personenvervoer. In: M. Despontin and C. Macharis (eds.), *Mobiliteit en (groot)stedenbeleid. 27ste Vlaams Wetenschappelijk Economisch congres*, 19 and 20 October 2006, Brussels, 417-446.

Kreutzberger, E.; Macharis, C., Woxenius, J.., 2006. Intermodal versus unimodal road freight transport - a review of comparisons of the external costs. In Jourquin, B., Rietveld, P., and Westin, L. (Eds.), *Towards better Performing Transport Systems*, Taylor and Francis. London, pp. 17-42.

Macharis, C., 2000. Strategische modellering voor intermodale terminals. *Socio-economische evaluatie van de locatie van binnenvaart/weg terminals in Vlaanderen,* PhD Thesis, Vrije Universiteit Brussel, Brussel.

Macharis, C., 2004. *The optimal location of an intermodal bargeterminal*. In : Beuthe, M.; Himanen, V. ; A. Reggiani and L. Zamparini (Eds.), Transport Developments and Innovations in an Evolving World, Springer-Verlag, pp. 211-234.

Macharis, C., Verbeke, A., 2004. "Intermodaal binnenvaartvervoer: Economische en strategische aspecten van het intermodaal binnenvaartvervoer in Vlaanderen", Antwerpen–Apeldoorn, Garant, 176 blz.

Macharis, C., and Pekin, E., 2008. Assessing policy measures for the stimulation of intermodal transport: a GIS-based policy analysis. *Journal of Transport Geography*. Accepted

Macharis, C., Pekin, E., Caris, A., Jourquin, B., 2008. *A decision support system for intermodal transport policy*, Brussel, VUBPRESS, 151 pp.

Macharis, C., Pekin, E. And Van Lier, T., 2010, "A decision analysis framework for intermodal transport: evaluating different policy measures to stimulate the market" in Givoni, M. & D. Banister (Eds) Integrated Transport: From Policy to Practice, Routledge, Oxfordshire, UK, p.223-240.

Maibach, M., C. Schreyer, D. Sutter (INFRAS), H. P. Van Essen, B. H. Boon, R. Smokers, A. Schroten (CE Delft), C. Doll (Fraunhofer Gesellschaft-ISI), B. Pawlowska, M. Bak (University of Gdansk), 2008. *Handbook on Estimation of External Cost in the Transport Sector. Internalisation Measures and Policies for All external Cost of Transport (IMPACT), Version 1.1.*, European Commission DG TREN, Delft, CE, The Netherlands.

NEA, 2008. Final report for the 'study on administrative and regulatory barriers in the field of inland waterway transport » - part B country reports, Zoetermeer, The Netherlands.

Ricci, A., Black, I., 2005. The social costs of intermodal freight transport. *Transport Economics*. Vol. 14. p.245-285.

Universiteit Antwerpen, VRIND 2008, Studiedienst van de Vlaamse Regering op basis van diverse bronnen, EC DG TREN, NMBS (België en gewesten), FOD MV, Algemene Directie Statistiek (België), PBV (Vlaams Gewest) in: Steunpunt Goederenstromen, 2008. Indicatorenboek 2008 – Duurzaam goederenvervoer Vlaanderen.

Vrenken, H., Macharis, C., Pekin, E., Peeters, A., Van Lier, T., Vaghhi, C., 2008. "Benefits and Costs of Intermodal Transport", TRALOTRA Workshop, Vrije Universiteit Brussel, 3 October 2008,

Weinreich, S., K. Rennings, C. Geβner, B. Schlomann and T. Engel, 1998. *External Costs of Road, Rail and Air Transport – a Bottom-Up Approach*. Paper presented at the 8th WCTR, Antwerp, Belgium.

A SIMULATION METHODOLOGY FOR THE ANALYSIS OF BUNDLING NETWORKS IN INTERMODAL BARGE TRANSPORT

An Caris
Gerrit K. Janssens
Transportation Research Institute
Hasselt University - campus Diepenbeek
Wetenschapspark 5 - bus 6, 3590 Diepenbeek, Belgium
e-mail: {an.caris,gerrit.janssens}@uhasselt.be

Cathy Macharis
Department MOSI - Transport and Logistics
Vrije Universiteit Brussel - Managementschool Solvay
Pleinlaan 2
1050 Brussel, Belgium
e-mail: Cathy.Macharis@vub.ac.be

KEYWORDS
Discrete event simulation, Inland navigation, Hinterland, Bundling

ABSTRACT

This paper presents the modelling methodology of a discrete event simulation model for intermodal barge transport. An intermodal freight transport network is modelled with the objective to understand the system and analyse various network configurations. Intermodal transport networks exhibit an increased complexity due to the inclusion of multiple transport modes, multiple decision makers and multiple types of loading units. Because of this increased complexity and the required level of detail, discrete event simulation is the appropriate tool of analysis. The conceptual and computerized modelling process is described and an application involving the simulation of alternative bundling networks is presented.

INTRODUCTION

In this paper a discrete event simulation model is presented to support decisions in intermodal barge transport. The objective of the simulation model is to assess the impact of policy measures on performance measures such as turnaround time of vessels, waiting time of barges in the port area and handling time of inland barges at sea terminals. According to Law (2007), simulation is a technique to imitate the operations of a real-world facility or process. The facility or process of interest is called a *system* and a set of assumptions about how it works is made in order to study it scientifically. An intermodal freight transport network is modelled with the objective to understand the system and analyse various network configurations. Intermodal transport networks exhibit an increased complexity due to the inclusion of multiple transport modes, multiple decision makers and multiple types of loading units (Caris et al. (2008)). The complexity of the intermodal transport system makes it impossible to describe all interactions by a mathematical model. Because of this increased complexity and the required level of detail, discrete event simulation is the appropriate tool of analysis. The intermodal hinterland network of the port of Antwerp serves as the real-world application in this study. This paper continues the work of Caris et al. (2009) by applying the proposed modelling methodology to the analysis of bundling networks. In the following section the simulation model is presented. Next, an application is demonstrated by simulating two alternative consolidation networks for intermodal barge transport. Finally, conclusions and directions for future research are given.

SIMBA MODEL

The discrete event Simulation model for InterModal BArge transport (SIMBA) described in this section is incorporated in a Decision Support System for Intermodal Transport Policy making (DSSITP), presented in Macharis et al. (2008). The DSSITP assessment framework uses three different models that are capable of assessing policies intended to enhance the growth of intermodal inland waterway and rail transport. The impact of policy measures is measured on all related transport modes and at multiple aggregation levels. Three core models, LAMBIT, SIMBA and NODUS make up the decision support system for intermodal transport policy making. For a detailed description of the LAMBIT and NODUS models, the reader is referred to the respective chapters in Macharis et al. (2008). The objective of the SIMBA model is to simulate possible policy measures for intermodal barge transport, but it can also be applied to analyse planning decisions of private stakeholders. Consequences and implications for the network performance measures can be estimated before implementation of a policy measure. The first subsection gives an overview of the current network configuration. Next, the conceptual model of the hinterland waterway network is presented. In the last subsection various aspects in the computerized modelling process are discussed.

Intermodal transport network

Figure 1 represents the port area of Antwerp. Three clusters of sea terminals can be identified. Until recently the main center of activity was situated on the right river bank. Sea terminals on the right river bank are either situated behind the locks (cluster 1) or in front of the locks at the river Scheldt (cluster 2). The two clusters are separated by three lock systems, indicated in figure 1 by three white blocks. Barges have to pass one of the three available lock systems to sail between cluster 1 and cluster 2. With the construction of a new dock (Deurganckdok) in the port of Antwerp, a third cluster of sea terminals emerged on the left river bank. Inland barges spend time in the port area, calling at multiple sea terminals and passing through the time-consuming locks.

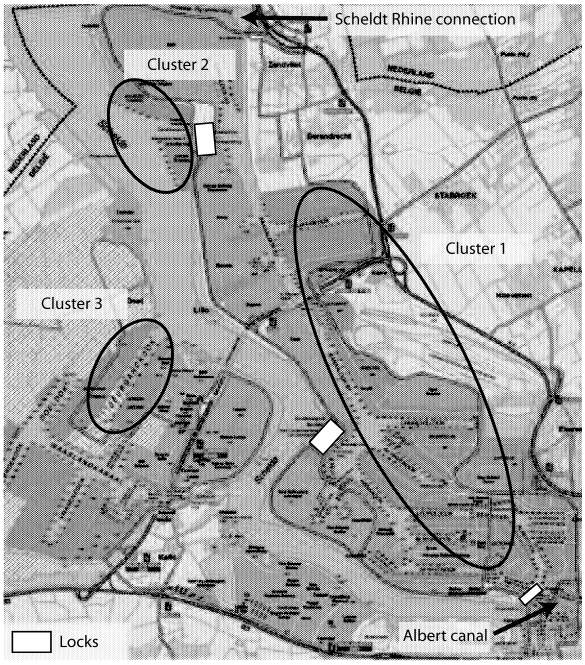


Figure 1: Port area of Antwerp (Adapted from Port of Antwerp)

In the analysis of alternative bundling networks, clusters are defined as all sea terminals at the same side of the three lock systems. Cluster 2 and cluster 3 are both situated on the left of the locks, directly accessible from the sea side through the river Scheldt. Therefore, these clusters are jointly referred to as 'left river bank'. Cluster 1 will be referred to as 'right river bank'. Inland vessels have to pass through a lock in the port area to sail from cluster 1 behind the locks on the right river bank to the sea terminals at the river Scheldt or on the left river bank in the Deurganckdok. Inland barges coming from the Albert canal have direct access to sea terminals on the right river bank in cluster 1 without having to pass through a lock system. Barges may also sail through the Scheldt-Rhine connection to Rotterdam and Amsterdam.

Shuttle services transport containers from inland terminals to sea terminals in the port area and carry containers from sea terminals to inland destinations in a round trip. A structural overview of the current network figuration, as assumed in the further analysis, is presented in figure 2. All inland terminals along each waterway axis that are included in the simulation model are mentioned. Three regions of origin can be identified in the Belgian hinterland network of the port of Antwerp: the basin of the Upper Scheldt and the river Leie, the Brussels - Scheldt Sea Canal and the Albert Canal.

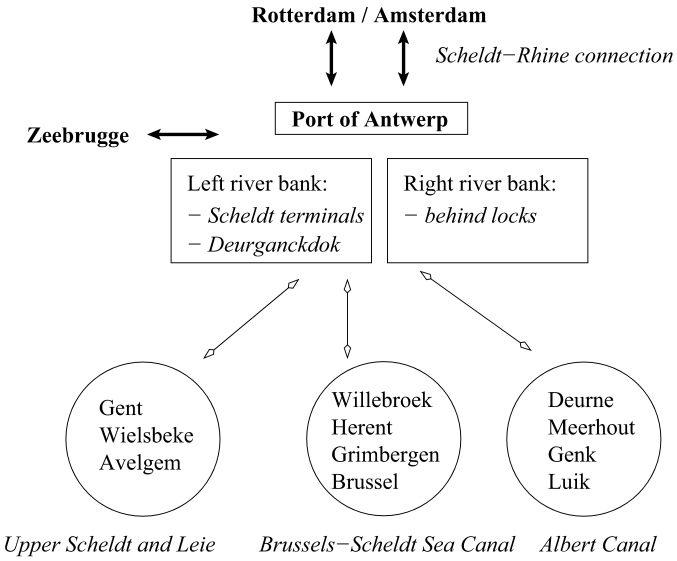


Figure 2: Current network configuration

Conceptual modelling

Three interrelated components can be identified in the intermodal hinterland network, as depicted in figure 3. The first component in the intermodal freight transport network is the inland waterway network. The inland waterway network is made up of terminals, waterway connections and container flows. Barges originate from the different inland terminals and carry containers in round trips to the various ports. Barges are of multiple sizes and carry a variable number of containers, based on real data input from inland container terminals. All barges enter the port area and visit one or multiple sea terminals. This may result in a low number of containers loaded or unloaded during a terminal call. A second component is the port area of Antwerp. Barges may visit sea terminals at the left river bank and right river bank in the same round trip. When sailing

from one cluster of sea terminals to the other, barges have to pass through one of the lock systems in the port area. Other port destinations are the port of Rotterdam or Amsterdam via the Scheldt-Rhine connection or the port of Zeebrugge via the Scheldt estuary. On the right and left river bank, barges queue for handling at the sea terminals. Service capacity at sea terminals is limited by the quay length for handling vessels. Maritime as well as inland vessels moor for loading or unloading containers at sea terminals. However, priority is given to seaborne vessels. Inland barges moor as soon as enough quay length is available. The handling time at the sea terminal depends on the number of containers that need to be unloaded from or loaded into the inland vessel. In the inland waterway network as well as in the port area multiple locks are present. Therefore, the lock planning constitutes a third major component.

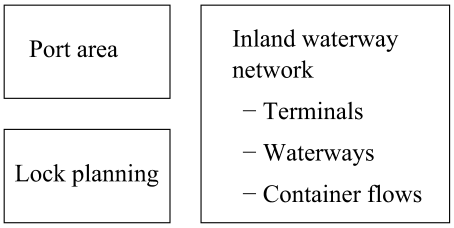


Figure 3: Components

Computerized modelling in Arena

This section describes how the conceptual model is translated into a computerized model in the simulation software Arena. The first subsection presents the general simulation approach. Next, assumptions underlying the simulation model are summarized. The following two subsections give an overview of inputs and outputs of the SIMBA model. Finally, the modelling of lockage operations and the calibration of the SIMBA model are discussed.

Discrete event simulation
In a discrete event system, one or more phenomena of interest change value or *state* at discrete points in time. These points in time are moments at which an *event* occurs. An event is defined as an instantaneous occurrence that may change the state of the system (Fishman (2001); Law (2007)). The players or *entities* in our discrete event simulation model are barges which sail through the waterway network. The simulation model is constructed in Arena, a simulation software based on queuing theory. Entities are defined as barges which originate from each inland terminal. Barges queue for handling at locks along waterway connections. Locks may be considered as a first group of service facilities in the network. Opening hours of locks are introduced in the simulation software as schedules for the availability of resources. Barges are collected in batches to go through the lockage process. After lock passage, batches are split into the original entities. When arriving in the port area, barges queue for handling at the quays of sea terminals. A second group of service facilities are thus the quays in the port area. The concept of shared queues is applied to model queueing at sea terminals throughout the model logic. Figure 4 depicts the flow of entities through the simulation model.

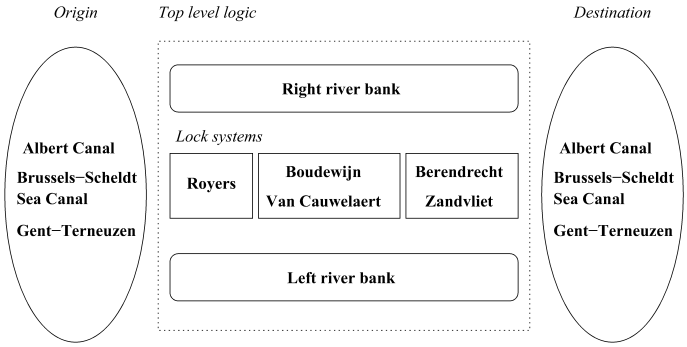


Figure 4: Flow of entities through the simulation model

The top level logic represents the port area. The model logic describes the two clusters of sea terminals on the right and left river bank, separated by three lock systems. Locks are constructed in separate submodels. Submodels are also applied for the three regions of origin in the hinterland network, namely the Albert Canal, the Brussels-Scheldt Sea Canal and the canal Gent-Terneuzen. Barges originating in the western part of the hinterland may sail through the canal Gent-Terneuzen and the Scheldt estuary to the port of Antwerp. After visiting all required terminals in the port area, barges return to their inland terminal and leave the simulation system. Stations and Route modules are introduced to keep the simulation model manageable. Examples of state variables in this discrete event system are the status of the servers (idle or busy), the number of barges waiting in a queue for handling at a lock or the time of arrival of a barge waiting in a queue for handling at a sea terminal. Events are for example the completion of service of a barge at a lock or the arrival of a barge at a sea terminal.

Assumptions
A number of assumptions are made to translate the conceptual model of the intermodal network into a discrete event simulation model. The emphasis lays on inland waterway transport. Rail connections in the hinterland network are not taken into account. All main waterway connections between inland terminals and the port area are incorporated in the simulation model. Small waterways without inland terminals are not included in the simulation model of the current situation. Pre- and end-haulage by road is also not incorporated.

In the first group of service facilities, the stochastic lockage times are represented by a triangular distribution. Sailing times on the network connections are assumed to be stochastic and also follow a triangular distribution. The arrival process of barges is based on real data input collected from the inland terminals, the waterway operators and the port authority.
The second group of service facilities consists of the quays at sea terminals. A fixed quay length is assumed for handling inland barges at each sea terminal. In reality the layout of sea terminals is aimed at handling seagoing vessels. In the port of Antwerp no dedicated quay sides are provided for inland navigation. Inland barges are handled with the same infrastructure and equipment and priority is given to handling seagoing vessels. However, no data is available on the arrival pattern and length of maritime vessels at the sea terminals. Therefore maritime vessels are not introduced into the simulation when modelling the handling at sea terminals. Instead, a given percentage of total available quay length is assigned to serving inland barges. In order to take the variability in available quay length into account, the handling of barges is modelled as a stochastic process. The handling of inland barges consists of mooring and loading or unloading containers. Both elements are modelled stochastically. The model further assumes a homogeneous container type. The same probability distribution is used for modelling the handling time of each container.
The variance-reduction technique of common random numbers is applied to synchronize various scenarios. A separate random number stream is assigned to each source of randomness. The basic idea is to compare alternative bundling networks under similar experimental conditions so that observed differences are due to differences in the system configuration rather than to fluctuations of the experimental conditions (Law (2007)). A stream of random numbers is dedicated to the lockage times, sailing times, handling times at terminals and choice of lock in the port area.

Data Requirements
All intermodal terminals in the inland waterway network are asked for information to identify current container flows in the network. Real data on shuttle services is used as an input for the simulation model. For each shuttle service the following information is required: which type of barge is used, which destinations are visited and what is the average number of import and export containers for each destination. Table 1 lists the attributes of each barge entering the network. In the second column an example is given. The simulation is run over 28 days or 672 hours. In this example a barge arrives in the simulation system at 16.43 hours, meaning it departs from the inland terminal Genk and sails to the port area of Antwerp. The barge has a width of 11.5 metres and a length of a hundred metres, leading to a surface area of 1150 m^2. In the port area first the cluster of sea terminals on the right river bank is visited. 57 containers need handling (loading or unloading) at two sea terminals. Next, the barge moors at four sea terminals on the left river bank and requires handling of 85 containers.

Attribute	Example
Departure time	16.43
Origin	Genk
Destination1	Antwerp: right river bank
Destination2	Antwerp: left river bank
Surface area	1150 m^2
Width	11.5 m
Length	100 m
Nb terminals right river bank	2
Nb handlings right river bank	57
Nb terminals left river bank	4
Nb handlings left river bank	85

Table 1: Entity attributes

Container transport interacts with other freight flows. Therefore, the flow of non-containerized goods on the inland waterway network is introduced as an input in the simulation model. These flows affect the waiting times at locks. Information is also necessary on the network connections. The waterway administrators provided information on the number of locks on each waterway, distances between locks, average lockage times, number of lock chambers and size of the chambers.
In the port area of Antwerp three clusters of locks connect the inner port area with the sea side. Data is required on the choice of locks when sailing in the port area. The average quay length available for handling inland navigation at sea terminals gives an indication of the service capacity in the port area of Antwerp. The port authority provided the average mooring time and time for loading and unloading in order to model service times of inland container barges in the port area. Service times in the port area include the time for mooring at each sea terminal plus the handling time of all import and export containers.

Performance measures
The simulation model allows to quantify a number of network properties resulting from the interaction of freight flows. Table 2 gives an overview of performance measures which are generated by the SIMBA model. The turnaround time of shuttles is defined as the total time necessary for a barge to sail from an inland container terminal to the port area, visit all sea terminals and return to the inland terminal. The turnaround time depends on the waiting times at locks and in the port area. The outputs measured at locks are the percentage of barges that have to wait, the number of barges that have to queue and the waiting time of barges in the

queue. In the port area the waiting time before handling is measured, as well as the number of vessels queueing for service. A final group of performance measures concerns the capacity utilization. In the port area this is expressed as the average percentage of quay length occupied. In the hinterland network the average and maximum number of barges on each network connection is recorded. Other performance measures can be added to the simulation model when necessary for future analyses.

Shuttles	turnaround time
Locks	total number waiting (%)
	number waiting in queue
	waiting time in queue
Port area	waiting time in queue
	number waiting in queue
Capacity utilization	quay length
	network connections

Table 2: Performance measures

Lockage process

The operations of locks strongly affect waiting times of barges for lockage. A number of decision rules are defined to make the operations of the locks in the simulation model reasonably realistic. A first group of decision rules relates to the assignment of barges to lock chambers, as depicted in figure 5. Barges are assigned to a lock chamber only if its size is within the allowed dimensions. The second decision rule assigns barges to the lock chamber with the smallest number of barges in queue. Thirdly, when no barges are waiting or an equal number of barges are queueing in front of each lock chamber, barges are assigned to the smallest lock chamber that is open. This decision rule focuses on a rapid lockage process of barges. Smaller lock chambers have a shorter lockage time. On the other hand, a more intensive use of larger lock chambers may reduce waiting times because more barges can be served simultaneously. A final decision rule is applied when in the latter case no lock chamber is open in the sailing direction of the barge. In this situation the barge is assigned to the lock chamber which is the first available. A second group of decision rules concerns the closing of lock chambers. A lock chamber is closed when there is not enough remaining space for the next barge in queue or when no additional barges arrive within a predefined number of time units. From interviews with waterway administrators it appears that the operations of locks are entrusted to a lockkeeper, without fixed rules. Future research could introduce more complex decision rules in the simulation model. For example, Ting and Schonfeld (1996) propose heuristic methods for the sequencing of vessels through locks, including locks with two dissimilar chambers. Theunissen and Janssens (2005) formulate a heuristic algorithm for the placement of inland vessels in a lock, with the aim to place as many vessels as possible from the arrival queue.

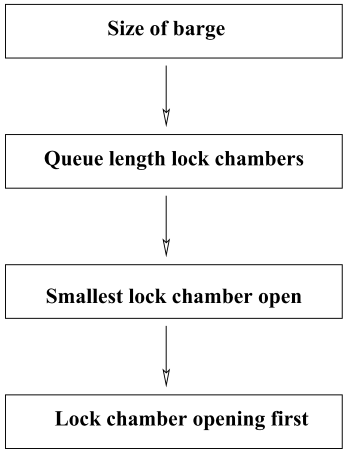


Figure 5: Decision rules for the assignment of barges to lock chambers

Calibration

Parameter settings for the description of locks are based on data input from the waterway operators. As an example, the parameter settings of the locks along the Albert Canal are described. Six lock systems are constructed on the Albert Canal, each consisting of two lock chambers for vessels up to 2000 tonnes and a third, larger lock chamber for push-towing. The standard service time for the first two lock chambers equals 45 minutes and for the third lock chamber 50 minutes. The standard service time is defined by the waterway operator as the maximum time in normal circumstances between arrival at 500 metres distance from the lock system and opening of the lock chamber to leave the lock system. This includes waiting until the lock chamber opens, sailing into the lock chamber and lockage time, but excludes sailing out of the lock chamber. From the data on lock passages provided by the waterway operator, an estimation could be made of the lockage times. For the two smaller lock chambers a triangular distribution is chosen with a mode of 16 minutes and a minimum and maximum of 12 to 20 minutes. The lockage time of the larger lock chamber is modelled with a triangular distribution with a mode of 18 minutes and a minimum and maximum of 16 to 20 minutes. The distance between locks is used together with an average speed of 10 km per hour to determine the average sailing time between locks.

The parameter setting in the port area is based on data provided by the port authority. The mooring and unmooring of barges takes 10 to 14 minutes, with a mode of 12 minutes. The loading or unloading of a single container when the inland barge has moored, is assumed

to take 2.5 minutes and varies between 2 to 3 minutes. The choice of locks is modelled with a discrete distribution for each combination of origin and destination in the port area. The same parameter settings for sailing times, lockage times and service times in the port area are made in all simulation scenarios in the following section.

During the DSSITP project, progress was regularly reported to a follow-up committee. This committee consisted of various stakeholders from the freight transport field, including waterway operators, railway operators, the Belgian railway infrastructure manager, terminal operators, the road haulage federation and the port authority. These follow-up committee meetings enabled a first verification of the model. Next, an enquiry is made into the turnaround times of vessels in order to verify the model. Table 3 summarizes transit times expressed in hours for sailing one way to the ports of Antwerp, Rotterdam and Amsterdam, as reported by the inland terminals. Some terminals mention a time interval, for example sailing from the terminal in Meerhout to the port of Antwerp may take six to eight hours. The data is based on the experience and general knowledge of inland terminal operators. Table 4 reports on the average transit times expressed in hours in the simulation model from the inland terminals to the entry point in the port area without lock passage. The transit times to the ports of Rotterdam and Amsterdam represent an inland barge sailing directly from the inland terminal to this port. As sailing times and lockage times are stochastic processes, individual transit times of vessels may deviate from the reported averages. Differences between the reported transit times of terminal operators and transit times in the simulation model may depend on the final point assumed in the port area. Furthemore, terminal operators may assume a combination of port visits. When looking at table 3, differences are also observed between estimates of various terminals. However, table 4 shows that transit times in the simulation model represent the estimates of the terminal operators.

Terminal	Antwerp	Rotterdam	Amsterdam
Deurne	3	12	
Meerhout	6-8	14-16	16-20
Genk	10-12	19-22	
Luik	14		
Gent	5-6	13	
Wielsbeke	12	18	
Avelgem	15	18	
Willebroek	4	14	
Grimbergen	5	15	
Brussel	5-6	19-20	
Herent	6		

Table 3: One way transit times (hours) - terminal operators

Finally, results of various simulation scenarios, reported in the following section, were presented and discussed with the port authority of Antwerp.

Terminal	Antwerp	Rotterdam	Amsterdam
Deurne	1.7	10.3	
Meerhout	6.6	15.2	19.2
Genk	11.9	20.5	
Luik	15.9		
Gent	6.0	14.4	
Wielsbeke	11.8	20.2	
Avelgem	15.9	21.5	
Willebroek	3.2	11.9	
Grimbergen	6.5	15.2	
Brussel	7.5	16.2	
Herent	7.3		

Table 4: One way transit times (hours) - SIMBA

ANALYSIS OF BUNDLING NETWORKS

In this section the SIMBA model is applied to investigate bundling concepts which may contribute to the improvement of intermodal barge operations. When looking at opportunities for consolidation in intermodal barge transport, two options can be discriminated. Freight may be bundled in the port area or in the hinterland of a sea port. A comparison is made between the current situation (**Current**) and these two bundling ideas by means of the SIMBA model.

First, consolidation of freight flows may be realised by providing a hub in the port area, from which cargo is distributed to the different sea terminals. Konings (2007) proposes to uncouple the collection and distribution services in the port area from the trunk haul services to the hinterland. By doing so inland barges do not have to call at multiple sea terminals. They only visit a hub in the port area. This leads to a reduction in turnaround time of vessels serving the hinterland. In the collection/distribution network containers with the same origin or destination can be bundled. This enables a more efficient and prompt handling of barges at sea terminals. In Caris et al. (2010b) the SIMBA model is applied to analyze four implementation scenarios of this bundling concept in the port of Antwerp. The most interesting scenario involves the provision of two hubs in the port area, one in each cluster of sea terminals at one side of the three lock systems. Inland barges only visit a single hub for which they do not have to pass through a lock system. The collection/distribution network is organized jointly for the two hubs. This hub scenario in the port area (**Port**) is the first bundling network reported in subsequent tables.

Second, economies of scale may be achieved by bundling load of different inland terminals destined to the same sea terminal. Inland terminals may cooperate with the

objective to create denser freight flows. In Caris et al. (2010a) cooperation between intermodal barge terminals in a hinterland network is analyzed from a network design perspective. The hinterland network is studied as a whole to see whether or not inland terminals in the network should cooperate in a corridor network. The methodology is applied to the hinterland network of inland barge terminals in Belgium. Next, selected cooperation schemes in the hinterland (**Hinter**) constitute the second bundling network simulated with the SIMBA model in tables 5 to 7. Table 5 gives the turnaround times of inland vessels, expressed in hours.

Table 5: Average turnaround times of inland terminals (in hours)

Avg turnaround time	**Current**	**Port**	**Hinter**
Deurne - Antw	15.20	9.16	33.07
	(0.47)	(0.14)	(0.33)
Deurne - Antw/Rdam	22.08	22.73	22.01
	(0.89)	(0.51)	(0.15)
Meerhout - Antw	29.24	25.64	35.06
	(0.47)	(0.39)	(0.54)
Meerhout -	41.70	38.84	42.44
Antw/Rdam/Adam	(0.38)	(0.59)	(0.48)
Genk - Antw	38.97	35.85	53.36
	(0.62)	(0.67)	(0.30)
Genk - Antw/Rdam	49.89	47.28	50.26
	(0.87)	(0.29)	(0.71)
Luik - Antw	46.46	41.90	59.68
	(0.34)	(0.23)	(0.40)
Gent - Antw	20.62	14.73	33.39
	(0.49)	(0.20)	(0.56)
Wielsbeke - Antw	38.62	28.77	40.22
	(0.42)	(0.24)	(0.37)
Avelgem - Antw	41.19	35.30	40.93
	(0.88)	(0.51)	(1.16)
Avelgem -	62.69	62.79	62.52
Antw/Rdam	(0.48)	(0.31)	(0.17)
Willebroek - Antw	14.79	11.45	23.06
	(0.17)	(0.07)	(0.16)
Willebroek -	35.59	35.81	35.37
Antw/Rdam	(0.39)	(0.25)	(0.22)
Grimbergen - Antw	20.93	16.55	32.59
	(0.21)	(0.08)	(0.28)
Brussel - Antw	21.91	19.03	33.59
	(0.34)	(0.17)	(0.28)
Brussel - Antw/Rdam	40.94	41.30	40.69
	(0.29)	(0.38)	(0.40)
Herent - Antw	21.91	18.75	21.85
	(0.19)	(0.08)	(0.20)

Inland vessels may only sail to Antwerp (Antw) or they can make a combined trip to Antwerp and Rotterdam (Rdam) or Amsterdam (Adam). Standard deviations are mentioned in brackets below the average turnaround times. In the hub scenario in the port area turnaround times of all inland terminals are significantly reduced. This reduction is explained by the fact that inland vessels only call at one hub and do not pass through any lock system in the port area. Results show that terminals involved in a corridor network in the hinterland have to take a longer turnaround time into account. The impact on turnaround times is larger as more terminals are involved.

Table 6 summarizes performance measures in the port area. The average and maximum waiting time before handling, expressed in hours, are given for the sea terminals on the right and left river bank and at the two hubs in the port area. Secondly, the average and maximum utilization of the quays on the right and left river bank and at the hubs are measured.

Table 6: Performance measures in the port area: current situation and intermodal barge hub right river bank

Port area	**Current**	**Port**	**Hinter**
Avg waiting time (in hours)			
Right river bank	0.0629	0.0000	0.0159
	(0.0306)	(0.0000)	(0.0117)
Left river bank	0.0557	0.0000	0.0255
	(0.0115)	(0.0000)	(0.0166)
Hub right	/	0.1352	/
		(0.0372)	
Hub left	/	0.0572	/
		(0.0088)	
Max waiting time (in hours)			
Right river bank	7.6128	0.0000	2.2597
Left river bank	4.3095	0.0000	5.1275
Hub right	/	8.1493	/
Hub left	/	2.7953	/
Avg capacity utilization			
Quay right river bank	0.1666	0.1583	0.1852
	(0.0017)	(0.0015)	(0.0019)
Quay left river bank	0.1741	0.1691	0.1997
	(0.0017)	(0.0018)	(0.0021)
Quay hub right	/	0.2050	/
		(0.0026)	
Quay hub left	/	0.1579	/
		(0.0011)	
Max capacity utilization			
Quay right river bank	0.9834	0.8696	0.9834
Quay left river bank	0.9850	0.5985	0.9850
Quay hub right	/	0.9660	/
Quay hub left	/	0.9100	/

Table 6 reveals that at peak moments the maximum capacity utilization in the hub scenario in the port area decreases by 38.65% on the left river bank and by 11.38% on the right river bank. Less quay length is necessary

to handle inland containers at peak hours. In the hinterland scenario less efficiency gains are recorded at sea terminals as in the hub scenario in the port area. At a hub in the port area freight is bundled of all terminals in the hinterland network, whereas in a hinterland cooperation network freight is only bundled of two to three terminals.
In table 6 the maximum waiting time over the ten simulation runs is mentioned. More details on the maximum waiting time before handling in the port area in each of the ten simulation runs may be found in table 7. Cooperation between terminals in the hinterland offers an opportunity to reduce maximum waiting times of inland barges at sea terminals.

Run	**Current**		**Port**		**Hinter**	
	Right	Left	Hub right	Hub left	Right	Left
1	3.60	3.33	3.18	2.80	1.15	1.76
2	7.61	4.07	2.31	2.52	0.57	3.49
3	2.71	4.31	2.48	1.93	2.21	1.02
4	3.72	3.81	2.86	1.94	1.21	0.54
5	3.88	3.12	3.52	2.62	0.00	1.43
6	4.74	4.15	3.60	1.54	2.06	1.13
7	2.88	3.34	8.15	1.75	2.26	5.13
8	3.50	1.30	4.00	1.89	0.54	0.46
9	3.48	3.31	3.43	1.77	1.48	1.37
10	2.09	4.17	6.32	2.05	0.90	1.53

Table 7: Maximum waiting times in port area

CONCLUSIONS AND FUTURE RESEARCH

In this paper the modelling process is presented of a discrete event simulation model for an intermodal barge transport network. The model is constructed to make a quantitative ex-ante analysis of policy measures to stimulate intermodal barge transport and is part of a larger decision support system for intermodal barge transport. The simulation model is applied to analyse opportunities of bundling freight flows in the port area and along the same river axis in the hinterland. In this study the main focus is on the inland waterway network. Potential extensions to the simulation model include the introduction of rail connections and the addition of a submodel to integrate intermodal terminal planning. The SIMBA model is also suited for other analyses, such as assessing the network wide impact of more complex decision rules for the operations of locks, the introduction of new intermodal barge terminals in the network or analyzing the consequences of growth scenarios on the network capacity.

Acknowledgement: We thank the Belgian Science Policy (BELSPO) for their support on our research project DSSITP (Decision Support System for Intermodal Transport Policy) in the research programme "Science for a Sustainable Development - call 2", under contract number SD/TM/08A.

REFERENCES

Caris A.; Janssens G.K.; and Macharis C., 2009. *Modelling Complex Intermodal Freight Flows.* In M. Aziz-Alaoui and C. Bertelle (Eds.), *Understanding Complex Systems Series: From System Complexity to Emergent Properties.*, Springer Berlin / Heidelberg. 291–300.

Caris A.; Macharis C.; and Janssens G.K., 2008. *Planning Problems in Intermodal Freight Transport: accomplishments and Prospects. Transportation Planning and Technology*, 31, no. 3, 277–302.

Caris A.; Macharis C.; and Janssens G.K., 2010a. *Modelling corridor networks in intermodal barge transport.* In *World Conference on Transport Research.* Lisbon, Portugal.

Caris A.; Macharis C.; and Janssens G.K., 2010b. *Network analysis of container barge transport in the port of Antwerp by means of simulation. Forthcoming in Journal of Transport Geography.*

Fishman G.S., 2001. *Discrete-event simulation: modeling, programming and analysis.* Springer Series in Operations Research.

Konings R., 2007. *Opportunities to improve container barge handling in the port of Rotterdam from a transport network perspective. Journal of Transport Geography*, 15, 443–454.

Law A.M., 2007. *Simulation modeling & analysis.* McGraw Hill, fourth ed.

Macharis C.; Pekin E.; Caris A.; and Jourquin B., 2008. *A decision support system for intermodal transport policy.* VUBPRESS.

Theunissen C. and Janssens G.K., 2005. *A 'less-flexibility-first' heuristic for the placement of inland vessels in a lock. Transportation Planning and Technology*, 28, no. 6, 427–446.

Ting C.J. and Schonfeld P., 1996. *Effects of tow sequencing on capacity and delay at a waterway lock. Journal of Waterway, Port, Coastal and Ocean Engineering*, 122, no. 1, 16–26.

DETERMINING OPTIMAL SHIPPING ROUTES FOR BARGE TRANSPORT WITH EMPTY CONTAINER REPOSITIONING

Kris Braekers
Gerrit K. Janssens
An Caris
Transportation Research Institute (IMOB) – Hasselt University
Universiteit Hasselt – campus Diepenbeek
Wetenschapspark gebouw 5, 3590 Diepenbeek, Belgium
E-mail: {kris.braekers,gerrit.janssens,an.caris}@uhasselt.be

KEYWORDS

Empty container management, barge transportation, service network design

ABSTRACT

Service network design for freight transportation is concerned with the selection and characteristics of routes on which services are provided. An efficient service network design should take empty container repositioning movements into account. These empty container movements are highly interrelated with loaded container transports. Unfortunately, most existing models do not consider both types of movements together. In this paper, a model for service network design in intermodal barge transportation is presented. Both loaded container transports and empty container repositioning movements are taken into account. The model is applied to the Albert Canal which connects the port of Antwerp with four hinterland ports. Several possible assumptions and empty container management scenarios are defined. The optimal shipping route under different demand conditions is calculated and the optimal location of an empty container hub in the hinterland is determined. Finally, the model is extended to a multi period model to cope with transport demand variation over periods.

1. INTRODUCTION

During the last two decades intermodal barge transport has gained market share in Northwestern Europe, with annual growth figures of 10% to 15% (Konings 2003). Currently, barge transport plays an important role in the hinterland access of major sea ports in this region. For the port of Antwerp in Belgium, the share of barge transport in the modal split rose from 22.5% to 34.8% between 1999 and 2009 (Port of Antwerp, 2009). Although many interesting contributions to literature have been made, Caris et al. (2008) indicate several intermodal planning problems that need further research attention, like service network design for intermodal barge transport.

Crainic and Laporte (1997) state that service network design is an important issue at the tactical decision level for intermodal transportation. It is involved with the selection of routes on which services are offered and the determination of characteristics of each service, particularly their frequency. State-of-the-art reviews on service network design in freight transportation are presented by Crainic (2000) and Wieberneit (2008). An overview of models for service network design in intermodal transportation may be found in Crainic and Kim (2007). Research on service network design specifically for intermodal barge transportation is scarce. Groothedde et al. (2005) discuss the design of a hub network for transporting palletized fast moving consumer goods by barge and road transport. Maras (2008) investigates the design of an optimal barge shipping route in a linear network. Caris et al. (2010) study the advantages of cooperation between hinterland terminals and different bundling strategies for barge transportation in the hinterland of the port of Antwerp.

Empty container repositioning is an important aspect that might be taken in to account when creating a service network for intermodal barge transportation and for freight transportation in general (Crainic, 2000). Because of regional imbalances between import and export volumes, some terminals or ports will create a surplus of empty containers, while others will face a deficit. In order to be able to satisfy future requests for empty containers by shippers, empty containers need to be repositioned. In barge transportation, these repositioning movements are made by using excess capacity of container ships (Choong et al. 2002; Maras 2008). Since these movements often represent a large part of the overall movements, they should be taken into account when creating service networks.

Empty container repositioning at a global level, in the context of maritime shipping network design, is studied among others by Shen and Khoong (1995), Cheung and Chen (1998), Lam et al. (2007) and Li et al. (2007). Also a large amount of research exists on empty container repositioning at a regional level, between shippers, consignees, depots and terminals (Crainic et al. 1993; Olivo et al. 2005; Jula et al. 2006; Chang et al. 2008). Unfortunately, all these papers optimize empty container flows without considering loaded container movements or by considering loaded container movements as given. Only a few papers consider the simultaneous optimization of loaded and empty container flows (Erera et al. 2005; Song and Dong 2008; Bandeira et al. 2009; Huth and Mattfeld 2009; Braekers et al. 2010).

Empty container repositioning in barge transportation is studied by Choong et al. (2002) and Maras (2008). Choong

et al. (2002) study the operational side of the empty container repositioning problem in intermodal barge transportation. The authors assume loaded container transports by barge to be known. They formulate a model that minimizes repositioning costs over multiple periods by using excess ship capacity and other uncapacitated, but more expensive transport modes (truck, rail). Experimental results show that a longer planning horizon can give better empty distribution plans because the use of slower but cheaper transport modes like barge is encouraged.

To the authors' knowledge only Maras (2008) investigates empty container repositioning in the context of service network design for barge transportation. Maras (2008) adapts a model introduced by Shintani et al. (2007) for service network design in maritime shipping. In the model of Shintani et al. (2007) the profit maximizing shipping route between a number of sea ports is determined. Not all ports have to be visited, but if a port is visited all transport demand at that port has to be satisfied. Container in- and outflow is balanced at every port, either by empty container repositioning using excess capacity or by leasing and storing empty containers. The problem is solved by a genetic algorithm.

Maras (2008) adapts the model of Shintani et al. (2007) to the context of barge transportation. The author considers the viewpoint of a logistic service provider or shipping company that wants to charter a single ship to offer a roundtrip barge service between a fixed start and end port. The objective is to maximize profit, while it is assumed that non-profitable transport requests may be turned down. The problem is to define which intermediary hinterland ports need to be visited between the start and end port, while taking both loaded container transport and empty container repositioning movements into account. Empty containers may be transported between any pair of ports and leased or stored at every port. Not all transport demand at a visited port has to be satisfied. Instead, it is assumed that any number of containers smaller than the transport demand may be transported between two ports. The model is applied to a sequence of ten ports. Since the number of possible routes is far lower in barge transportation than in maritime shipping because all ports are situated across a single axis, Maras (2008) is able find optimal solutions by using commercial software. The author finds the maximum profit for five types of ships under a single transport demand situation.

In this paper, the model of Maras (2008) is extended in several ways and applied to the management of hinterland transport chains in Western Europe. A dummy port is introduced so that the starting port should not be predefined. More realistic assumptions regarding fulfilling transport demand at hinterland ports are made. The model formulation is adapted to represent the situation of the Albert Canal which connects four hinterland ports with two clusters of terminals in the port of Antwerp. Three possible empty container management scenarios are defined and the optimal location for an empty container hub in the hinterland is determined. The model is tested for all scenarios under different transport demand conditions. Finally, the model is extended to a multi period model, including storage and leasing options, to handle situations where transport demand is not the same every period.

A detailed problem description is given in Section 2. The network structure, adapted to the Albert Canal, is presented in Section 3. The formulation of the single period model and experimental results are presented in Section 4. In Section 5, the multi period model is described. Finally, in Section 6 conclusions are drawn and future research opportunities are identified.

2. PROBLEM DESCRIPTION

In this paper, the viewpoint of a logistic service provider or shipping company that wants to offer a regular barge service between a sea port and some hinterland ports is considered. The decision maker has to determine which loaded containers are to be transported and thus which hinterland ports need to be visited. The objective is to maximize profit. Revenues are generated by loaded container transports. Costs of performing the loaded container transports and empty container repositioning costs are taken into account. Although empty containers may be transported at a marginal cost using excess capacity, these movements cause handling costs, take loading and unloading time and reduce the capacity available for loaded container transports (Choong et al. 2002; Maras 2008). At each port, the total number of containers coming in and going out needs to be balanced. In contrast with Maras (2008), no leasing and storage option is considered in the single period model. Since it is the intention of performing the same route every period, continuously leasing containers at a certain port and storing containers at another port seems not realistic in our problem context.

The ship starts its route at one of the hinterland ports, travels to the port of Antwerp and then returns to the same hinterland port. Along the route, one or more hinterland ports in-between may be visited. The intermediate ports visited upstream may differ from those visited downstream. While in the model of Maras (2008) the starting port is fixed in advance, this is not the case in this paper. A dummy port is introduced as the starting point of the ship, which means that the actual starting port is decided by the model.

As mentioned in the introduction, Maras (2008) assumes that any number of loaded containers may be transported between two ports as long as this number is smaller than the transport demand between these two ports. In this paper, two more realistic assumptions are proposed and the model is tested for both. The first assumption is that all transport demand originating from a hinterland port should be fulfilled when it is visited downstream. Likewise, all transport demand to a port should be fulfilled when it is visited upstream. Upstream and downstream decisions are made separately. The second assumption is that transport demand to and from a certain hinterland port is the sum of the transport demand of several clients. Therefore, the decision maker may choose to serve one, several or all of these clients. This means that the total transport demand at a port is made up of several blocks and that the number of

containers transported should be a combination of these blocks.
Finally, three different empty container management scenarios are considered in this paper. Scenario one represents the case where empty containers may only be transported to and from an empty container hub located in the port area. In the second scenario, an empty container hub is located in one of the hinterland ports besides the hub in the port area. In the third scenario, a hub is located in all ports which means that empty containers may be transported between all pairs of ports.

3. NETWORK

The model presented in this paper is applied to the situation of the Albert Canal which connects the port of Antwerp with hinterland ports in Deurne, Meerhout, Genk and Luik. In the port area of Antwerp, two clusters of sea terminals can be identified, one on the right river bank and one on the left river bank (see Figure 1). Both clusters are separated by three lock systems, which means ships have to pass through a lock in the port area to sail from one cluster to another.

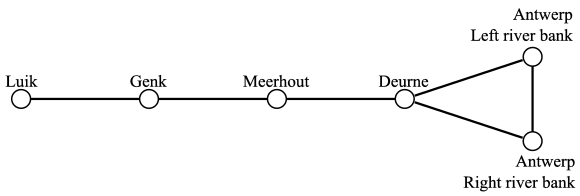


Figure 1: The Albert Canal

Because travelling from one river bank to the other may take two and a half hours, both clusters are considered as a separate port. It is assumed that a ship may visit a river bank only once each roundtrip. If it is decided to visit both river banks, the order of visiting should be free to choose since this may have an impact on the capacity available to reposition empty containers in some of the scenarios described in Section 2. In order to preserve the linear representation of the ports, a duplicate node is created for the cluster at the right river bank. Next, all hinterland ports and the dummy port are duplicated to facilitate the formulation of the problem. The final network representation is shown in Figure 2. The dummy port is represented by nodes 1 and 13, Luik by nodes 2 and 12, Genk by nodes 3 and 11, Meerhout by nodes 4 and 10, Deurne by nodes 5 and 9, Antwerp right river bank by nodes 6 and 8 and finally Antwerp left river bank by node 7. The ship starts and ends at the dummy port (node 1 and 13) and can only travel from a node to another node with a higher number.

4. SINGLE PERIOD MODEL

4.1 Problem Formulation

In this section the problem formulation for the single period model is presented. The following notation is used:

$P = 1, \dots, 13 =$ set of 13 ports
$L = \{(i,j) \mid i \in P \backslash 13, j \in P \backslash 1, i < j\}$
$c_i^e =$ entry cost at port i (€)
$c_i^h =$ handling cost at port i (€/TEU)
$t_i^a =$ standby time for arrival at port i (h)
$t_i^d =$ standby time for departure at port i (h)
$t_i^h =$ handling time per container at port i (h/TEU)
$t_{max} =$ maximum roundtrip time (days)
$c_{ch} =$ daily charter costs (€/day)
$c_f =$ fuel price (€/ton)
$c_l =$ lubricant price (€/ton)
$sfc =$ specific fuel consumption (tons/kWh)
$slc =$ specific lubricant consumption (tons/kWh)
$N_{inst} =$ engine output/propulsion (kW)
$CAP =$ ship capacity (TEU)
$Y =$ profit (€)
$R =$ total revenues (€)
$C =$ total costs (€)
$C_{char} =$ ship charter costs (€)
$C_{fuel} =$ voyage fuel costs (€)
$C_{lubr} =$ voyage lubricant costs (€)
$C_{entr} =$ total port entry costs (€)
$C_{hand} =$ total handling costs (€)
$T =$ roundtrip time (h)
$T_{trav} =$ total travel time (h)
$T_{entr} =$ total time entries (h)
$T_{hand} =$ total handling time (h)

For all combinations of ports i and j with $(i,j) \in L$, the following parameters and variables are introduced:

$r_{ij} =$ loaded container freight rate (€/TEU)
$d_{ij} =$ loaded container transport demand (TEU)
$t_{ij} =$ travel time (h)
$x_{ij} =$ loaded containers transported (TEU)
$y_{ij} =$ empty containers transported (TEU)
$z_{ij} = \begin{cases} 1 & \text{if ports } i \text{ and } j \text{ are directly connected} \\ 0 & \text{else} \end{cases}$

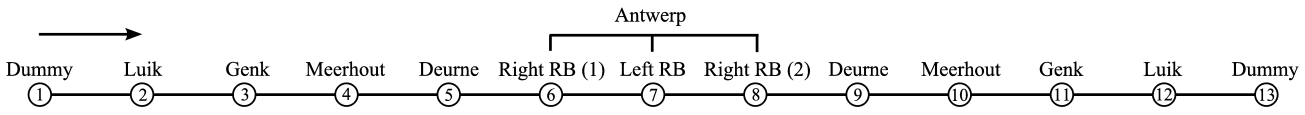


Figure 2: Network Representation

The problem is formulated as follows:

$$Max\ Y = R - C \quad (1)$$

Subject to:

$$R = \sum_{(i,j)\in L} x_{ij} \cdot r_{ij} \quad (2)$$

$$C = C_{char} + C_{fuel} + C_{lubr} + C_{ent} + C_{hand} \quad (3)$$

$$C_{char} = c_{ch} \cdot t_{max} \quad (4)$$

$$C_{fuel} = c_f \cdot sfc \cdot N_{inst} \cdot \sum_{(i,j)\in L} z_{ij} \cdot t_{ij} \quad (5)$$

$$C_{lubr} = c_l \cdot slc \cdot N_{inst} \cdot \sum_{(i,j)\in L} z_{ij} \cdot t_{ij} \quad (6)$$

$$C_{entr} = \sum_{\substack{(i,j)\in L \\ i\neq 1}} z_{ij} \cdot c_j^e \quad (7)$$

$$C_{hand} = \sum_{(i,j)\in L} (x_{ij} + y_{ij})(c_i^h + c_j^h) \quad (8)$$

$$x_{ij} \le d_{ij} \cdot \sum_{q=i+1}^{j} z_{iq} \qquad (i,j) \in L \quad (9)$$

$$y_{ij} \le CAP \cdot \sum_{q=i+1}^{j} z_{iq} \qquad (i,j) \in L \quad (10)$$

$$x_{ij} \le d_{ij} \cdot \sum_{q=i}^{j-1} z_{qj} \qquad (i,j) \in L \quad (11)$$

$$y_{ij} \le CAP \cdot \sum_{q=i}^{j-1} z_{qj} \qquad (i,j) \in L \quad (12)$$

$$\sum_{q=1}^{i} \sum_{s=j}^{n} (x_{qs} + y_{qs}) \le CAP + M(1 - z_{ij}) \qquad (i,j) \in L \quad (13)$$

$$\sum_{(j,i)\in L} (x_{ji} + y_{ji}) - \sum_{(i,j)\in L} (x_{ij} + y_{ij}) = 0 \qquad i \in P \quad (14)$$

$$\sum_{j=2}^{13} z_{1j} = 1 \quad (15)$$

$$\sum_{i=1}^{n-1} z_{i13} = 1 \quad (16)$$

$$z_{1.2} = z_{12.13} \quad (17)$$

$$z_{1.3} = z_{11.13} \quad (18)$$

$$z_{1.4} = z_{10.13} \quad (19)$$

$$z_{1.5} = z_{9.13} \quad (20)$$

$$\sum_{i=1}^{5} z_{i6} + \sum_{i=1}^{7} z_{i8} \le 1 \quad (21)$$

$$\sum_{i=1}^{q-1} z_{iq} - \sum_{j=q+1}^{13} z_{qj} = 0 \qquad q = 2, \dots, 12 \quad (22)$$

$$T = T_{trav} + T_{entr} + T_{hand} \quad (23)$$

$$T < 24 \cdot t_{max} \quad (24)$$

$$T_{trav} = \sum_{(i,j)\in L} z_{ij} \cdot t_{ij} \quad (25)$$

$$T_{entr} = \sum_{\substack{(i,j)\in L \\ i\neq 1, j\neq 13}} z_{ij} \cdot (t_i^d + t_j^a) \quad (26)$$

$$T_{hand} = \sum_{(i,j)\in L} (x_{ij} + y_{ij}) \cdot (t_i^h + t_j^h) \quad (27)$$

$$x_{ij} \text{ integer} \qquad (i,j) \in L \quad (28)$$

$$y_{ij} \text{ integer} \qquad (i,j) \in L \quad (29)$$

$$z_{ij} = \{0,1\} \qquad (i,j) \in L \quad (30)$$

The objective is to maximize profit, represented by revenues minus total costs (1). Revenues are calculated by multiplying the number of loaded containers transported between two ports and the corresponding freight rate (2). Total costs are the sum of ship charter costs, fuel costs, lubricant costs, port entry costs and container handling costs (3). Ship charter costs are determined by the maximum roundtrip time and daily charter costs (4). Fuel and lubricant costs depend on the distance travelled, engine power of the ship and the respective fuel and lubricant prices (5,6). Port entry costs are calculated in equation (7). Handling costs for both loaded and empty containers are calculated in equation (8). Constraints (9), (10), (11) and (12), together with constraint (22), ensure that no loaded or empty containers are transported between two ports when they are not connected. The capacity of the ship is controlled by constraint (13). For each port, container inflow and outflow should be the same (14). The ship must start and end at the dummy port (15,16) and the final real port must be the one corresponding with the first real port (17,18,19,20). The right river bank, represented by nodes 6 and 8, may only be visited once (21) and when a ship enters a port, it should also leave it (22). The total time of the roundtrip is the sum of the travel time, port entry time and container handling time (23) and must be lower than the maximum roundtrip time (24). The travel time, port entry time and container handling time are calculated by equations (25), (26) and (27). Finally, the number of loaded and empty containers transported should be integer values and the linking variables between nodes are binary variables (28,29,30).

Under the assumption that all transport demand at a port needs to be fulfilled when the port is visited, the following constraints are added:

$$x_{ij} = d_{ij} \cdot a_{ij} \qquad (i,j) \in L \quad (31)$$

$$a_{ij} = \{0,1\} \qquad (i,j) \in L \quad (32)$$

Under the assumption that the transport demand at each port is divided into a number of blocks, the following parameters and constraints are added:

B_{ij} = number of demand blocks between i and j

d_{ij}^b = transport demand in block b between i and j (TEU) with $\sum_{b\in B_{ij}} d_{ij}^b = d_{ij}$

$$x_{ij} = \sum_{b \in B_{ij}} d_{ij}^{b} e_{ij}^{b} \qquad (i,j) \in L \qquad (33)$$

$$e_{ij}^{b} = \{0,1\} \qquad (i,j) \in L,\ b \in B_{ij} \qquad (34)$$

Finally, for each empty container management scenario, the appropriate values of y_{ij} are set to be zero.

4.2 Experimental Results

Experimental results for the single period model are presented in this section. The model is applied to the situation of the Albert Canal for a ship with a capacity of 120 TEU. Maximum roundtrip time is set at two days. Figures for cost and time parameters are based on real values or found in other research papers (Konings 2007; Maras 2008). The model is solved using Lingo 10.0.

Artificial but realistic data are used for the loaded container transport demand. For each hinterland port, transport demand to and from each cluster in the port area ranges between 20 and 40 TEU, between 50 and 70 TEU or was set to zero. Fifteen different demand situations are defined, varying from each other in terms of:

- balanced or unbalanced distribution of demand over the hinterland ports,
- balanced or unbalanced downstream and upstream demand,
- transport demand to and from one or both clusters at the port.

For each situation, two instances are generated. Together with ten random instances, this results in a set of 40 problem instances.

In Section 2, two assumptions for fulfilling transport demand are described and three empty container management scenarios are presented: an empty container hub in the port area, a hub in the port area and in the hinterland or a hub at every port. A distinction is made whether the hub in the port area is located on the left or right river bank, or on both river banks.

For each of the 40 problem instances, the optimal shipping route is determined for all combinations of demand assumptions and empty container management scenarios. Results are shown in Table 1. The rows represent the different empty container management scenarios. For scenario two the hub in the hinterland may be located in Deurne (2-D), Meerhout (2-M), Genk (2-G) or Luik (2-L). The columns distinguish between the two demand assumptions and the three possibilities for hub location in the port area. Results for the third empty container management scenario are always the best. For the other scenarios the relative gap with the profit of scenario three is shown.

Table 1 shows that results for the first empty container scenario are worse than those of scenarios two and three, especially for the case where all transport demand needs to be fulfilled at a port. When comparing scenarios one and two, it seems that a network with a single empty container hub in the hinterland already gives substantially better results compared with a network with only a hub in the port area. Results are even better when a hub is located at every port.

The model formulation allows to determine the optimal inland location of an empty container hub. It can be seen that on average Genk is the best location under all circumstances. However, the difference with other locations is often small. When using the model in practice, the decision where to locate a hub should therefore be based on accurate cost and demand information.

An empty container hub on the right river bank in the port area gives on average better results than a hub on the left river bank. Probably this is due to the fact that the travel time from the right river bank to the hinterland ports is smaller than for the left river bank. When a hub is located at both river banks, results are even better.

Table 1: Overview of Results

Scen.	All			Blocks		
	Left	Right	Both	Left	Right	Both
1	16.39	13.05	8.66	8.80	7.78	4.95
2-D	7.14	4.99	4.01	4.85	4.77	3.75
2-M	4.72	3.20	2.10	4.54	4.45	3.39
2-G	4.35	2.74	1.74	3.76	3.88	2.82
2-L	9.66	7.23	5.74	4.24	4.19	3.47
3	-	-	-	-	-	-

No large difference in the average loading degree of the ship is found between the different demand assumptions and empty container management scenarios. When travelling between two ports, on average 70% of the capacity is taken up by loaded containers, while 6% is used for empty container repositioning. The number of hinterland ports visited is on average 2.41. Roundtrip time is almost always equal to the maximum of two days, implying that time represents a restriction on the number of containers that can be transported. Only in 65% of the experiments both clusters in the port area are visited. The time restriction may be a reason for this, since travelling between both clusters takes a considerable amount of time.

5. MULTI PERIOD MODEL

In the previous sections it is assumed that transport demand is the same in every period. A single period model is used since the optimal shipping route does not change and each period the same number of loaded and empty containers are transported. In reality, transport demand will not be constant over periods. For example, some shippers may want containers to be transported every week, while others only require containers to be transported once every two weeks. In order to handle these kind of situations, the model described in the previous sections is extended to a multi period model.

For each period, the optimal shipping route is defined. This route is not necessarily the same for each period. However, if the transport demand of a certain client is fulfilled in one period, it should also be fulfilled in the other periods.

Short term container leasing and storage options are introduced in the multi period model, while they are not

considered in the single period model. Since the number of loaded containers transported between two ports may differ from period to period, empty container repositioning needs will also change. A port may have a surplus of empty containers in one period and a deficit in another. In that case, temporarily storing empty containers at a port and leasing empty containers elsewhere could be an interesting option. Empty container repositioning costs and handling time are saved and more capacity is available to transport loaded containers. The extra costs due to leasing and storing containers are introduced in the model. These costs depend on the duration of the lease and storage. The model decides which option, repositioning empty containers or leasing and storing empty containers, is the best in each situation.

Each port has a starting stock of empty containers available (this can be zero). At the end of the planning period, the same amount of containers should be located at that port. Finally, it is assumed that empty containers can only be leased and returned at ports where an empty container hub is located. Storing containers is possible at every port.

To test the multi period model, ten random two period problem instances are generated with transport demand at each port varying between both periods. The length of a period is assumed to be a single week and empty container inventory at every port is assumed to be zero at the beginning of the first week. Results are obtained for all three empty container management scenarios. The empty container hub in the port area is assumed to be located on the right river bank. For the second scenario the hinterland hub is assumed to be located in Genk, since this gave the best results for the single period model. Only the assumption where all transport demand in a port has to be satisfied is considered.

Results show that on average the relative gap between the profit of the different empty container management scenarios increases substantially. Scenarios one and two are on average respectively 23.10% and 9.96% worse than scenario three, while this was only 13.05% and 2.74% in the single period case. The gap between scenario one and two is 14.59% (10.60% in the single period case), which shows again that a single empty container hub in the hinterland has a large positive effect on profit in comparison with the situation where there is only a hub in the port area.

Another interesting finding is that increasing the planning horizon may offer better results. Choong et al. (2002) show that using a longer planning horizon may offer better empty container distribution plans because slower but cheaper transport modes like barge may chosen over faster but more expensive modes like road transport. The multi period model described in this section shows that a longer planning horizon may also offer benefits when optimizing loaded and empty container movements in barge transportation simultaneously. When the planning period for the two period problem instances is increased to four weeks, with the transport demand of weeks three and four being the same as those of weeks one and two, slightly more profit is generated. Average weekly profit increases for empty container management scenario one by 0.02%, for scenario two by 1.15% and for scenario three by 0.83%. Further research is needed to investigate the cause of these improvements.

6. CONCLUSIONS

Empty container repositioning should be taken into account when designing a service network for barge transportation. The model presented in this paper simultaneously decides on which loaded container transports and empty container repositioning movements should be performed. The profit maximizing route for a ship is determined while taking time and capacity restrictions into account.

The model is applied to the Albert Canal in the hinterland of the port of Antwerp. Two assumptions regarding fulfilling transport demand and three empty container management scenarios are defined. Finally, the model is extended to a multi period model that can cope with demand variations over periods.

Results clearly show the advantage of an empty container hub in the hinterland, both in the single and multi period case. For the single period case, results for a network with a hub in the port area and one in the hinterland are even close to those of a network with a hub in all ports. Besides, the presented models allow to determine the best location of an empty container hub in the hinterland.

Future research could focus on the effect of changing the maximum roundtrip time of the ship. Also the tradeoff between the capacity of the ship and the frequency of service may be investigated. Finally, more insight in the results of the multi period model may be gained.

REFERENCES

Bandeira, D.L.; J.L. Becker; and D. Borenstein. 2009. “A DSS for Integrated Distribution of Empty and Full Containers.” *Decision Support Systems* 47, No.4, 383-397.

Braekers, K.; G.K. Janssens; and A. Caris. 2010. “A Deterministic Annealing Algorithm for Simultaneous Routing of Loaded and Empty Containers.” In *Proceedings of the Industrial Simulation Conference 2010 (ISC 2010)* (Budapest, Hungary, June 7-9). Eurosis, Ostend, Belgium, 172-176.

Caris, A.; C. Macharis; and G.K. Janssens. 2008. “Planning Problems in Intermodal Freight Transport: Accomplishments and Prospects.” *Transportation Planning & Technology* 31, No.3, 277-302.

Caris, A.; C. Macharis; and G.K. Janssens. 2010. “Modelling Corridor Networks in Intermodal Barge Transport.” In *Proceedings of the World Conference on Transport Research (WCTR 2010)* (Lisbon, Portugal, July 11-15).

Chang, H.; H. Jula; A. Chassiakos; and P. Ioannou. 2008. “A Heuristic Solution for the Empty Container Substitution Problem.” *Transportation Research Part E: Logistics and Transportation Review* 44, No.2, 203-216.

Cheung, R.K. and C.-Y. Chen. 1998. “A Two-Stage Stochastic Network Model and Solution Methods for the Dynamic Empty Container Allocation Problem.” *Transportation Science* 32, No.2, 142-162.

Choong, S.T.; M.H. Cole; and E. Kutanoglu. 2002. “Empty Container Management for Intermodal Transportation Networks.” *Transportation Research Part E: Logistics and Transportation Review* 38, No.6, 423-438.

Crainic, T.G.; M. Gendreau; and P. Dejax. 1993. “Dynamic and Stochastic Models for the Allocation of Empty Containers.” *Operations Research* 41, No.1, 102-126.

Crainic, T.G. and G. Laporte. 1997. “Planning Models for Freight Transportation.” *European Journal of Operational Research* 97, No.3, 409-438.

Crainic, T.G. 2000. “Service Network Design in Freight Transportation.” *European Journal of Operational Research* 122, No.2, 272-288.

Crainic, T.G. and K.H. Kim. 2007. “Intermodal Transportation.” In *Handbooks in Operations Research and Management Science Volume 14: Transportation,* C. Barnhart and G. Laporte (Eds.). North-Holland, Amsterdam, 467-537.

Erera, A.L.; J.C. Morales; and M.W.P. Savelsbergh. 2005. “Global Intermodal Tank Container Management for the Chemical Industry.” *Transportation Research Part E: Logistics and Transportation Review* 41, No.6, 551-566.

Groothedde, B.; C. Ruijgrok; and L. Tavasszy. 2005. “Towards Collaborative, Intermodal Hub Networks: A Case Study in the Fast Moving Consumer Goods Market.” *Transportation Research Part E: Logistics and Transportation Review* 41, No.6, 567-583.

Huth, T. and D.C. Mattfeld. 2009. “Integration of Vehicle Routing and Resource Allocation in a Dynamic Logistics Network.” *Transportation Research Part C: Emerging Technologies* 17, No.2, 149-162.

Jula, H.; A. Chassiakos; and P. Ioannou. 2006. “Port Dynamic Empty Container Reuse.” *Transportation Research Part E: Logistics and Transportation Review* 42, No.1, 43-60.

Konings, R. 2003. “Network Design for Intermodal Barge Transport.” *Transportation Research Record: Journal of the Transportation Research Board* 1820, 17-25.

Konings, R. 2007. “Opportunities to Improve Container Barge Handling in the Port of Rotterdam from a Transport Netwrok Perspective.” *Journal of Transport Geography* 15, No.6, 443-454.

Lam, S.-W.; L.-H. Lee; and L.-C. Tang. 2007. “An Approximate Dynamic Programming Approach for the Empty Container Allocation Problem.” *Transportation Research Part C: Emerging Technologies* 15, No.4, 265-277.

Li, J.-A.; S.C.H. Leung; Y. Wu; and K. Liu. 2007. “Allocation of Empty Containers between Multi-ports.” *European Journal of Operational Research* 182, No.1, 400-412.

Maras, V. 2008. “Determining Optimal Transport Routes of Inland Waterway Container Ships.” *Transportation Research Record: Journal of the Transportation Research Board* 2062, 50-58.

Olivo, A.; P. Zuddas; M. Di Francesco; and A. Manca. 2005. “An Operational Model for Empty Container Management.” *Maritime Economics & Logistics* 7, No.1, 199-222.

Port of Antwerp. 2009. *Annual Report 2009.* Antwerp Port Authority. Antwerp. (Available at www.portofantwerp.com)

Shen, W.S. and C.M. Khoong. 1995. “A DSS for Empty Container Distribution Planning.” *Decision Support Systems* 15, No.1, 75-82.

Shintani, K.; A. Imai; E. Nishimura; and S. Papadimitriou. 2007. “The Container Shipping Network Design Problem with Empty Container Repositioning.” *Transportation Research Part E: Logistics and Transportation Review* 43, No.1, 39-59.

Song, D.-P. and J.-X. Dong. 2008. “Empty Container Management in Cyclic Shipping Routes.” *Maritime Economics and Logistics* 10, No.4, 335-361.

Wieberneit, N. 2008. “Service Network Design for Freight Transportation: A Review.” *OR Spectrum* 30, No.1, 77-112.

TRANSPORT MODELLING AND SIMULATION

A NINE HUNDRED VARIABLE NONLINEAR TRANSPORTATION PROBLEM WITH EXCESS SUPPLY

William Conley
Departments of Business Administration and Mathematics
Austin E. Cofrin School of Business
University of Wisconsin-Green Bay
Green Bay, Wisconsin 54311-7001
U.S.A.
Conleyw@uwgb.edu

KEYWORDS
Nonlinear transportation problem, cost, Monte Carlo Simulation

ABSTRACT

The classic applied mathematics transportation problem results in a linear system of equations and a linear cost equation. It can be solved by the well known linear programming simplex technique or other modified linear solution techniques. However, many practical shipping problems in the business and industrial world involve nonlinear cost equations, where there are discounts and returns to scale for shipping in substantial quantities. These are generally more difficult to solve mathematically. Therefore, presented here is an attempt to solve a large nonlinear transportation problem with a computer statistical optimization technique (multi stage Monte Carlo optimization) that is followed by a discussion of linear and nonlinear optimization and goal programming applied to practical real world problems.

1. INTRODUCTION

Many applied mathematics books (for example, Barnett and Ziegler 1993) present the standard linear transportation problem as a practical way of delivering a product in bulk from n locations to m destinations where the n times m costs associated with shipping units along the n times m different routes are all linear. These linearity assumptions make the problems easier to solve with classical techniques.

However, desktop PC computers are so fast and affordable in the twenty-first century that it is possible to at least approximate the solution to these transportation problems when the linear cost equation is replaced by a nonlinear one that allows for discounts for shipping in quantity. A one hundred variable problem of this type where supply exactly equaled demand, was worked on in (Conley, 2007) using the multi stage Monte Carlo optimization (MSMCO) solution technique. Presented here is an attempt to use MSMCO on a much larger nonlinear transportation problem where there is an excess of supply in relation to current demand.

2. A NINE HUNDRED VARIABLE EXAMPLE

A company wants to ship 23565000 units of a product from 30 warehouses to 30 stores and, if possible, have the total shipping costs be less than 15 million dollars. The 30 stores currently individually require 771000, 772000, 773000, . . . up to 800000 units of the product. However, the 30 warehouses have a total of 29500000 units of the product in amounts of 970000, 971000, 972000 . . . up to 1000000 units of the product available to ship. The nonlinear cost equation (reflecting returns to scale for larger shipments from each warehouse to each store) is

$$C(i,j) = \sum_{i=1}^{30}\sum_{j=1}^{30} .06 * (i+j) * x(i,j) ** (.7+.005 * (i+j)) \quad (1)$$

(where * is multiply and ** is raise to a power) and Xij is the amount shipped from warehouse i to store j.

Therefore, the cost equation is set equal to 14500000 (in an attempt to reduce costs to under 15 million dollars). The other thirty equations are the summations of the sets of thirty variables that represent the amounts shipped from all of the warehouses to each individual store, set equal to their store requirement amounts.

Consequently, the MSMCO computer simulation attempts to minimize the sum of the absolute values of the differences between the left and right hand side of these 31 equations subject to all Xij ≥ 0 i=1,2, . . . 30 and j = 1, 2, . . .30 and the thirty less than or equal excess supply constraints in the warehouses.

Note that because the cost equation is so big in relation to the other equations, a weight of .065 is multiplied times its absolute value difference to make for a smoother simulation. This is done because when using MSMCO to

solve a system of equations, if all of the right hand side constants are about equal MSMCO can find a useful solution using less computer time.

3. THE SOLUTION

Nine hundred random numbers are read in, inside the warehouse constraints limits. If they collectively (in appropriate groups of 30) go over the warehouse limits they are shrunk back inside these bounds. Then the 31 equation 900 variable transformed function is evaluated. Every time this evaluation produces a better answer, it is stored and the next nine hundred "random" numbers are read in centered about this best answer so far. Again, if a better answer is produced it is stored and the simulation is re-centered about this new best answer. This is done 40000 times (called stage one). Then the search region is reduced and another 40000 solution tries are similarly attempted (called stage two). Fifty such reduced stages (50 x 40000 = 2000000 solution tries) yields fairly useful answers after about 12 minutes of run time on a desk top HP Compaq with Intel chip PC.

Ten such solution attempts were tried and the best one is reported here. Notice in Table 1 the printout of the errors in stage 1, 254 million; stage 2, 181 million, stage 3, 133 million, etc. However, by stage 25 the total error is under 30000 and less than 12000 by stage 50.

The 900 variable values are in Table 2. Also, notice that the errors in the 30 store amount (Table 3) are about all zero. There is one error of five units in store 14. Also, there are errors of 1.3 and 1.6 units in stores 27 and 28. All of the other store errors are less than one unit. Also, the cost equation of $14500000 is off by 11177/.065 = $171954 which is comfortably below the $15000000 goal.

4. THE PRINTOUT

Table 1 contains the 50 stage errors, indicating the convergence to a useful solution.

Table 1: The Fifty Stage Errors

1	253974336.0000
2	181450112.0000
3	133090536.0000
4	94968104.0000
5	70488912.0000
6	51905732.0000
7	36831688.0000
8	26046976.0000
9	3345744.2500
10	3345744.2500
11	3345744.2500
12	3345744.2500
13	3184015.0000
14	3151988.7500
15	2673863.0000
16	2108587.0000
17	1665868.8750
18	849757.6875
19	439612.0000
20	270740.6250
21	166167.6406
22	84773.9375
23	55832.6758
24	49117.0273
25	29335.9063
26	26139.8066
27	21585.8906
28	16624.4883
29	15956.9268
30	14079.1348
31	13353.2119
32	12594.8350
33	11974.5166
34	11717.8066
35	11607.6670
36	11590.0449
37	11529.3145
38	11375.1318
39	11300.5674
40	11271.6221
41	11242.2969
42	11233.2549
43	11218.2441
44	11208.3799
45	11207.1445
46	11197.9072
47	11192.2119
48	11191.0869
49	11190.7021
50	11190.2646

Table 2: The Nine Hundred Variable Shipping Amounts

1	6902.72	48732.15	16518.37	24822.07	34181.18
1	57288.98	13325.69	23032.69	10291.73	43842.20
1	32068.20	26863.63	29766.60	20159.00	32865.77
1	6889.74	20335.66	6495.61	17433.54	17966.55
1	8356.81	26758.70	33158.61	4158.15	29525.55
1	19307.50	21457.02	9614.47	72321.43	69109.24
2	1608.24	1314.26	32030.46	8742.75	14343.28
2	1697.40	25258.54	21634.71	33455.79	10378.09
2	17577.21	39570.62	11628.57	17616.45	11970.27
2	6479.13	9154.26	11600.00	1467.05	10846.29
2	49960.49	84356.55	80563.47	21946.33	16585.24
2	26402.03	35634.09	8552.89	56905.40	58892.53
3	33846.85	38736.83	66069.27	53053.11	34640.66
3	18826.15	55009.40	41206.88	23995.58	34378.34
3	15349.17	1358.09	26281.54	53685.39	79407.00
3	33477.68	29623.26	28054.74	8579.70	6726.84
3	6955.95	15780.19	9102.27	5739.42	26926.65
3	15314.80	269.02	54706.67	23089.40	6665.12
4	51546.25	1891.33	11603.35	43525.53	5702.09
4	12962.24	50665.24	32117.01	27584.38	53501.39
4	18414.60	20992.31	3593.53	161.37	46592.78
4	28759.41	50020.75	50466.33	8881.13	29805.47
4	6154.47	944.70	56726.21	5190.64	18627.91
4	2725.12	37105.00	30743.39	82162.60	36903.50
5	41623.94	45897.61	22005.05	2081.09	4714.18
5	53533.57	59798.43	18836.52	9560.16	38634.83
5	41365.98	16819.64	27060.41	32728.27	23187.91
5	12350.78	71007.64	10042.56	9762.32	43966.56
5	566.51	62680.51	1541.36	61132.66	1317.38
5	24747.53	1187.21	6710.87	10505.65	7086.44
6	35320.12	32794.18	56984.60	4295.57	16293.18
6	23720.23	16147.88	14922.10	35326.34	450.77
6	8732.99	715.92	20007.81	20399.68	42458.99
6	2353.67	11685.25	3216.62	19949.14	20346.73
6	3319.18	61617.99	13949.11	32114.32	40777.41
6	14184.82	13465.45	83009.01	45605.32	166.17
7	6511.62	48404.81	8708.64	9571.88	17742.40
7	14017.81	1474.66	15543.85	30064.52	1949.31
7	44036.56	13873.26	43269.98	4746.45	729.61
7	33837.16	63666.24	17687.79	13514.36	77668.48
7	23622.41	10592.48	74898.59	3694.60	64202.77
7	83287.21	12220.38	28547.08	6277.14	1532.77
8	17119.69	55993.89	54616.26	1483.78	11456.84
8	38357.40	6851.41	64721.58	6092.29	12977.52
8	11305.95	22659.20	6098.33	66613.13	92012.44
8	28618.64	2855.11	38763.62	11001.34	23341.66
8	18776.80	5595.15	41616.52	4062.42	48407.31
8	2116.01	15620.70	10698.92	63786.22	39573.88
9	2343.12	7606.90	46970.12	9713.86	48148.30
9	5751.44	33769.38	3084.90	18391.87	66571.63
9	51148.73	49634.39	7828.53	25815.13	41578.33
9	24017.76	30460.27	9168.41	32276.92	34001.70
9	103447.21	23110.97	14733.05	13593.51	37610.59
9	1011.94	57093.03	760.01	6323.32	63386.35
10	15836.77	8344.50	15529.71	2859.89	45001.49
10	25899.87	21566.84	23594.57	3977.67	14460.82
10	29104.00	23086.98	2642.31	13962.44	39559.15
10	27796.72	17307.88	25183.77	32256.56	34142.48
10	35786.30	41327.89	14713.60	18752.42	26098.11
10	79100.79	436.45	71734.57	29064.19	6288.69
11	1988.60	3412.60	3084.25	47936.21	50480.23
11	37525.03	11261.91	1546.55	60437.64	11269.10
11	10321.31	71033.41	41720.49	5368.94	60182.04
11	6686.82	46883.69	75483.37	49042.66	47169.36
11	10428.6	513437.24	58950.34	27271.53	15376.60
11	15142.27	43592.20	19725.11	27202.50	8778.19
12	3937.22	33027.48	11241.92	10489.28	20317.91
12	47254.06	30802.02	8451.97	5303.42	42283.44
12	3255.08	31124.14	47895.18	72043.02	32499.97
12	11667.58	7941.13	40218.91	13319.57	1060.75
12	37573.99	69113.04	20933.66	5607.29	82334.28
12	6699.55	22213.59	2972.31	37250.00	3083.51
13	1792.36	27603.03	27781.79	3575.22	9570.24
13	66205.57	11215.29	5258.60	13603.50	3841.34
13	8057.45	44464.30	29316.41	124.89	17718.66
13	58791.54	18542.30	6127.96	88162.62	36404.67
13	15585.25	26524.54	45077.15	11357.26	64161.94
13	9595.41	2260.94	23083.51	25847.46	4661.09
14	13952.00	23809.82	4930.82	56809.01	15167.43
14	29834.43	27271.70	13406.60	38047.09	48182.31
14	55667.70	6864.46	90154.47	17999.48	2064.56
14	20939.64	43970.27	20726.20	15293.47	12445.79
14	29603.47	1912.06	22236.83	60373.38	16822.63
14	30820.17	15729.52	11896.79	895.51	27327.30
15	22056.89	10187.33	26323.79	8576.98	35987.55
15	1839.93	118.37	51453.66	71945.52	5840.92
15	6321.10	3347.57	51318.33	30318.19	4881.57
15	17529.10	10379.56	30820.32	89513.94	36428.44
15	50745.48	22364.26	16595.34	52891.28	12407.11
15	27936.60	20857.65	12838.31	1335.69	75212.29
16	4140.10	12804.00	2142.25	10099.87	6626.06
16	9172.60	82110.76	31159.58	30627.39	17858.08
16	38439.25	27550.36	4309.59	31037.60	4345.99
16	6318.98	66048.73	30564.89	14650.81	32235.27
16	10411.12	52065.86	10274.10	45775.89	2071.84
16	67461.23	8308.83	7844.60	11880.53	20515.14
17	32970.99	76997.63	5663.54	58734.11	60704.49
17	5237.51	10633.83	9624.48	37656.78	42486.77
17	6425.74	3311.89	16744.93	17238.31	50598.16
17	5572.39	8860.96	10385.55	15666.12	1360.99
17	34316.52	25003.50	7752.04	51713.41	95.68
17	114162.67	20341.61	21748.97	23068.14	6877.10
18	7163.30	9786.08	8290.73	10512.44	1537.51
18	2760.36	33460.63	17469.24	42172.90	7365.01
18	39439.31	35254.66	8757.35	3069.10	19179.20
18	15040.79	23598.40	41202.09	57365.54	18747.93
18	16747.50	30230.33	30267.89	35010.59	17040.19
18	10943.31	98524.04	24720.33	3627.87	5077.22
19	1430.44	51704.95	28737.01	4739.83	5862.30
19	51837.16	155.12	61684.49	27640.09	6383.73
19	376.32	57353.72	5897.63	54886.10	9400.28
19	41424.22	32304.31	12142.35	34368.05	13091.11
19	13838.82	8859.26	25808.36	37556.89	26618.82
19	26287.39	20244.62	4000.49	37066.76	67273.70
20	36461.35	39860.76	7810.15	64738.86	7340.29
20	28568.88	3022.84	4459.18	5675.43	37891.45
20	54697.58	64085.63	41365.90	1702.48	23802.65
20	70638.36	1756.72	13732.79	43081.28	21854.75
20	80126.42	76078.52	11262.54	17681.39	40638.15
20	16570.21	24829.32	40884.82	24329.90	9371.35
21	10734.96	17654.00	41866.04	57852.64	13046.19
21	2227.44	44821.98	5154.97	27750.45	15274.96
21	71815.95	15217.08	55079.57	31119.27	1256.39
21	91601.88	1020.81	26807.51	3378.55	32553.92
21	25864.52	3745.58	33668.46	14179.05	30009.43
21	5943.88	66794.78	36466.85	54072.88	36766.45
22	3423.10	37713.74	11650.82	22348.07	34088.09
22	53007.52	10377.01	51504.48	1845.58	33729.07
22	56490.30	30965.30	24640.57	16857.17	55245.21
22	6532.73	4177.88	33297.89	23913.66	31765.64
22	11765.22	6567.95	2365.16	6593.18	1553.45
22	4803.99	60878.27	9171.15	30944.71	67349.51
23	109475.77	2240.71	20029.80	20122.04	63883.19

23	31895.69	38372.61	9900.59	5197.52	27974.96
23	109475.77	2240.71	20029.80	20122.04	63883.19
23	31895.69	38372.61	9900.59	5197.52	27974.96
23	15831.70	8540.94	5970.78	6550.96	5613.36
23	22777.96	48963.00	62712.55	11368.46	23441.04
23	69349.80	3866.81	10886.48	12460.86	5658.96
23	6757.95	23666.87	71940.84	29077.53	33893.11
24	54811.06	11374.78	41410.84	8549.85	10672.88
24	9000.80	51642.16	76414.84	59102.93	18576.27
24	38224.54	19936.85	22625.78	23446.87	6311.03
24	59582.01	2532.86	5269.63	35978.51	33505.25
24	6673.26	20050.29	34998.00	32624.36	10229.86
24	79419.21	26944.21	52208.46	5106.97	22013.89
25	51839.52	22198.28	12528.90	26934.12	35771.65
25	23640.97	44590.13	29952.08	3034.74	20851.95
25	12249.01	57016.28	8434.93	20113.31	17510.16
25	29733.39	27215.13	26989.99	46279.83	5487.26
25	35352.77	16517.41	2097.07	43370.16	5758.22
25	9378.15	34498.88	68768.90	9453.41	12594.59
26	41463.66	6387.22	9076.95	66864.23	10020.02
26	39423.03	66729.92	26363.27	31759.38	46943.21
26	8599.50	27827.21	45975.95	7403.73	2955.96
26	6841.11	38624.86	29889.60	31600.26	1174.13
26	13569.74	6797.18	13639.05	61374.44	28841.45
26	6584.72	26126.42	5558.62	25035.52	95299.44
27	91437.83	2569.93	29909.54	1488.73	46559.14
27	2839.21	6356.78	84412.88	17605.90	62709.63
27	9631.07	35735.06	21230.06	3933.69	3860.36
27	26660.62	20833.52	20947.43	6173.28	38939.73
27	52662.95	42621.45	42635.00	32529.87	22922.87
27	50960.91	2000.50	3696.94	6890.64	2596.35
28	35569.82	20699.92	79719.34	24106.97	54180.88
28	58489.09	8212.36	8230.36	46254.05	2686.46
28	23037.47	13235.53	6017.56	30378.48	23786.11
28	66892.88	53725.61	47043.93	22386.45	49188.97
28	2665.38	11350.11	13298.02	1110.33	33521.08
28	22284.79	33099.19	1205.38	3502.33	2792.81
29	33559.95	64690.11	58894.86	34291.98	22955.95
29	9582.47	1056.33	2202.10	46349.62	41880.77
29	2519.44	2312.15	3943.96	54619.22	12411.70
29	12802.88	21269.25	28863.84	3576.91	1434.40
29	11013.30	1407.06	36956.96	17313.69	23121.23
29	3791.53	22205.05	41309.93	10942.57	4292.29
30	131.94	7561.22	10870.93	75080.09	38004.66
30	13603.08	10920.76	20655.79	8249.68	8826.18
30	50496.70	11249.42	73423.00	99907.27	21014.77
30	3384.91	2234.81	2409.31	28757.92	4229.95
30	5759.98	20722.55	12294.90	56820.71	45737.82
30	12258.30	29396.49	32881.53	35428.95	4620.19

Table 3 contains the 31 store and cost equation errors

Table 3: The 31 Store and Cost Equation Errors

31 Error Terms	Right Hand Side Equation Value
0.12500	771000.00000
0.00000	772000.00000
0.00000	773000.00000
0.25000	774000.00000
0.25000	775000.00000
0.12500	776000.00000
0.00000	777000.00000
0.50000	778000.00000
0.06250	779000.00000
0.37500	780000.00000
0.06250	781000.00000
0.00000	782000.00000
0.06250	783000.00000
5.37500	784000.00000
0.25000	785000.00000
0.50000	786000.00000
0.18750	787000.00000
0.50000	788000.00000
0.06250	789000.00000
0.06250	790000.00000
0.25000	791000.00000
0.12500	792000.00000
0.12500	793000.00000
0.00000	794000.00000
0.37500	795000.00000
0.00000	796000.00000
1.31250	797000.00000
1.62500	798000.00000
0.31250	799000.00000
0.25000	800000.00000
0.00000	800000.00000
0.00000	799000.00000
0.00000	798000.00000
0.00000	797000.00000
0.00000	796000.00000
0.00000	79000.000000
0.00000	794000.00000
0.00000	793000.00000
0.00000	792000.00000
0.00000	791000.00000
0.00000	790000.00000
0.00000	789000.00000
0.00000	788000.00000
0.00000	787000.00000
0.00000	786000.00000
0.00000	785000.00000
0.00000	784000.00000
0.00000	783000.00000
0.00000	782000.00000
0.00000	781000.00000
0.00000	780000.00000
0.00000	779000.00000
0.00000	778000.00000
0.00000	777000.00000
0.00000	776000.00000
0.00000	775000.00000
0.00000	774000.00000
0.00000	773000.00000
0.00000	772000.00000
0.00000	771000.00000
11177.13961	14500000.00000

5. GOAL PROGRAMMING AND OPTIMIZATION

One could make the point that this is really a goal programming problem, where each of the 31 equations was a goal, the most important one to get the shipping costs under 15 million dollars. However, if the company has enough computer time, it could then lower the cost equation to, for example, 11 million dollars, and keep rerunning the simulation to try to lower the cost even more. Then some of the store requirement equations might have slight or large errors in them and management would have to choose the "best" of these solutions.

The linear programming approach (with a linear cost equation) will produce an exact solution to an oversimplified model. The multi stage Monte Carlo (MSMCO) simulation approach produces sometimes exact, but usually approximate, solutions to more accurate (nonlinear costs) models of real world problems. Either or both approaches may have value to the manager, depending on the application one is pursuing.

6. CONCLUSION

Presented here was a discussion of the classic linear transportation problem, followed by making the case for a nonlinear transportation cost analysis (when appropriate) in order to have a more accurate model for practical applications, even if it is more challenging to solve.

Then the multi stage Monte Carlo optimization (MSMCO) technique was used to approximate the solution to a 900 variable nonlinear transportation problem involving 31 equations and 30 major, less than or equal, constraints that arose because of excess supply (over demand). A brief discussion of goal programming and optimization followed.

It could also be noted that linear programming is an exterior algorithm (that goes along the edges of the feasible solution spaces looking for answers). Whereas, the MSMCO algorithm employed here is an interior algorithm that searches the entire feasible solution space (looking for solutions) and hence is more general purpose for applications than is linear programming. This is demonstrated in Conley (2008), Wong (1996) and many other publications. Both can be useful depending on the engineering, science or business optimization problem at hand.

REFERENCES

Barnett, R.A. and Ziegler, M.R. 1993. *Applied Mathematics for Business Economics, Life Sciences and Social Sciences,* 5th edition, New York, Dellen MacMillan.

Conley, W. C. 2008. "Ecological Optimization of Pollution Control Equipment and Planning from a Simulation Perspective," The International Journal of System Science, Vol. 39, pp. 1-7.

Conley, W.C. 2007. "A Nonlinear Transportation Problem with one Hundred Variables," in Proceedings of the 2007 European Simulation and Modeling Conference ESM2007, St. Julian's Malta Eurosis, Belgium, pages 287-290.

Wong, J.Y. 1996. "A Note on Optimization in Integers," International Journal of Mathematical Education in Science and Technology, Vol. 27, No. 6, 865-874.

BIOGRAPHY

WILLIAM CONLEY received a B.A. in mathematics (with honors) from Albion College in 1970, an M.A. in mathematics from Western Michigan University in 1971, an M.Sc. in statistics in 1973 and a Ph.D. in mathematics - computer statistics from the University of Windsor in 1976. He has taught mathematics, statistics, and computer programming in universities for over 30 years. He is currently a professor emeritus of Business Administration and Statistics at the University of Wisconsin at Green Bay. The developer of multi stage Monte Carlo optimization and the CTSP multivariate correlation statistics, he is the author of five books and more than 200 publications world wide. He is a member of the American Chemical Society, a fellow in the Institution of Electronic and Telecommunication Engineers and a senior member of the Society for Computer Simulation.

CREATING AN INNOVATIVE ACTIVITY-BASED FREIGHT TRANSPORTATION FRAMEWORK

Tabitha Maes
Katrien Ramaekers
An Caris
Tom Bellemans
Gerrit K. Janssens

Transportation Research Institute, Hasselt University
Wetenschapspark 5 - Bus 6
B-3590 Diepenbeek, Belgium
E-mail:{tabitha.maes;katrien.ramaekers;an.caris;tom.bellemans;gerrit.janssens}@uhasselt.be

INTRODUCTION

In a growing globalised context and consumption economy freight transport is of crucial importance. Activities of firms are expanding, even across borders. This causes an increase in logistics activities of firms as they become more dynamic. Public and private decision makers need to take these trends into consideration with regard to their decisions and a better projection of freight traffic flows becomes necessary. Being able to understand the drivers of freight flows makes it possible to forecast freight flows in the future and to calculate the impact of different policies on freight traffic. It will put policymakers in the position to get a better insight in the way the transport of goods comes about. Still, freight demand modelling is lacking behind on the efforts made in passenger transport models.

Hence, there is a growing need for models that can predict freight flows more accurately. Here the category of activity-based models comes into play, as they are able to better represent the link with the economy, interactions between different actors and the logistic elements inherent of freight movement. The development of a comprehensive and reliable freight transport model is needed. This paper is structured as follows. First, the need of activity-based models in freight transport will be elaborated in the next section. The following two sections take a closer look at the main differences with passenger transport. Starting with the different actors involved and then the logistic elements, inherent at freight transport, are presented. Next an overview of the options for an innovative activity-based freight transportation framework is given. Finally, some conclusions will be drawn.

ACTIVITY-BASED MODELLING IN FREIGHT TRANSPORT

To given an idea of the emerging trends in freight transportation modelling and the aspects of activity-based models, an overview of the main developments in literature is given.

Today, most state-of-the-practice models in freight transport are still four-step models, where the focus is on individual trips. These models have as main disadvantage, that they are looking at the aggregated flows between zones and cannot model flows at a more detailed level. For that, they are missing out on the behavioural aspects behind transport and are having errors due to aggregation. More importantly they are lacking elements of logistics decision making. The importance of incorporating logistics decisions and behavioural aspects in a freight transportation model is widely recognized (Tatineni and Demetsky (2005), Tavasszy et al. (1998), MOTOS (2006) and Liedtke (2009)). Some of the more recently developed four-step models are already incorporating logistic components (Tavasszy et al. (1998), SCENES Consortium (2000) and Yin et al. (2005)). However, these models are on an aggregated level and are not taken into account aspects of the different agents.

Recent trends in freight modelling are moving to agent-based models, which are part of the group of activity-based models and focus on each freight agent separately. Therefore they are better able to model their individual operational decisions and their interactions concerning logistics and transports. Furthermore a disaggregated approach is applied, by looking at trips and decisions on a microscopic scale and no longer to aggregate flows between different zones. This enables the understanding and representation of roles that each actor plays in the freight transportation system, as also

the interactions between actors. Besides, it is possible to incorporate changes in actors and their interactions over time. These elements are of fundamental importance in the development of more behavioural models for the freight system (Roorda et al., 2010). The disaggregated approach of these models, together with the representation of the different actors, enables better modelling possibilities for logistics decisions.

One of the main differences between modelling freight and passenger transport is that there are more actors involved in the decision making process. First there are firms who are sending and receiving goods, shippers who are responsible for the organization of the consignment and modes and the last group are carriers who undertake the movement (Ortúzar and Willumsen, 2001). Next to this, there are several other firms responsible for the transhipment, storage and custom facilities. The transportation of goods may follow a network of shippers, carriers, forwarders, terminals, distribution centres and others to arrive at its destination. These logistics chains are typical for the movement of freight and need to be taken into account when it comes to modelling freight flows. The economic transactions between suppliers and consumers, and the logistics operations that actually deliver the goods, are the two main drivers behind the rapidly evolving patterns of freight movements (Yin et al., 2005). Therefore, more attention has to be paid to the different actors.

Another difference with passenger transport is the dynamic nature of the freight logistics system, therefore trends in industry supply chains need to be considered. Trends like just-in-time (JIT) logistics are having an impact on the modes used, and size and frequency of shipments. Two of the main trends stated in Hesse and Rodrigue (2004) are:

- Demand-side orientation of activities. While traditional delivery was primarily managed by the supply side, current supply chains are increasingly managed by demand.
- Logistics services are becoming complex and time-sensitive. This has led to the point that many firms are now sub-contracting parts of their supply chain management to third-party logistics providers. These providers benefit from economies of scale and scope.

This leads to the need for agent-based models at a microscopic level. Roorda et al. (2010) gives several reasons for implementing agent-based modelling. First of all, there are diverse actors involved in the production and distribution of goods, none of which may have full control or even knowledge of all decisions made throughout the supply chain. Secondly, the interactions between firms are diverse and finally business models are changing over time. So a close follow up of all these interactions is requested to have a more realistic image of freight transport flows. Due to the modelling at a micro level, it is possible to look at individual instead of aggregated flows. This gives the opportunity to include individual firm characteristics and detailed representation of commodity groups. When looking at single movements of goods, more information of a shipment may be represented that would go lost in aggregated data. As stated in Liedtke and Schepperle (2004), the activity-based approach can explain the effects of individual behaviour changes on the whole transport system. This allows improving the quality of forecasts for public and private planners.

The following two sections elaborated more on the freight actors involved in the process and the logistic elements that have to be included in a comprehensive model.

THE FREIGHT ACTORS INVOLVED

In this section the role of the agents or actors is presented, hereby looking at which agent are defined and what their responsibilities are in the decision making process.

The groups of agents that are mostly used are shippers, receivers/customers, carriers/transporters and forwarders (Boerkamps et al., 1999; Liedtke, 2009; Wisetjindawat et al., 2007) and may be expanded to include politics as in the GoodTrip model. The role of these different actors is briefly discussed in the next paragraph.

The receiver or customer initiates the demand and chooses a supplier to deliver the required goods. After the shipper is chosen the receiver decides on the delivery moment, shipment size and whether he conducts the transport himself or not (Boerkamps et al., 2000). In Wisetjindawat et al. (2007) shippers play a major role in the selection of carrier and vehicle choice. Boerkamps et al. (2000) state that shippers are often responsible for transportation and therefore have to decide on mode choice, vehicle type and vehicle size. Furthermore, they decide on grouping of goods types, product range to offer, location of facilities, availability of distribution channels and whether or not to maintain own transport services. The carriers or transporter are responsible for the actual movement of the shipments and the tour planning problems. In Liedtke (2009) forwarders have the extra responsibility to build and coordinate transport chains. Freight movements are also indirectly influenced by politics (Boerkamps et al., 1999). Politics have an influence on the market structure, as they are responsible for an optimal spatial-economic organization. On the transport

market they may make a difference by regulating the accessibility and mobility of the transport.

Many companies act as both shipper and receiver. Ultimately, a simple actor may fulfil all roles in the supply chain, that is, as receiver of goods deliveries, as shipper and/or as transporter of shipments. An actor may be active in different activity types, for example: consumer, supermarket, distribution centre, production factory, etc. (Boerkamps et al., 2000). At the same time, they may own a private fleet with which they deliver their own goods and may provide transportation services to other companies as well (Roorda et al., 2010).

The framework of Roorda et al. (2010) and the TAPAS model (Davidsson et al., 2008) differentiate themselves with regard to the choice of agents they have made. The models are working at an even more detailed level and are able to incorporate more of the interactions between, and the decisions of, the different agents.

Roorda et al. (2010) established a new set of agents in his framework. The main agents are business establishments, firms and facilities (commodity, business service and logistics service).

- Business establishment: an organization at a specific location that produces, processes, or stores commodities, or provides business or logistics services. A business establishment may include several different facilities.
- Firm: an organization that owns or operates one or more business establishments. Within a logistics firm, business establishments at different locations may be integrated into a logistics network.
- Commodity production facility: one of the internal resources of a business establishment. The function of a commodity production facility is to produce or process commodity inputs.
- Business service facilities: provide services instead of commodities.
- Logistics service facility: provides logistics services, including transportation and inventory.
- End consumers: initiates demand for commodities.

The TAPAS model uses six different agents (Davidsson et al., 2008): Customer, Transport Chain Coordinator (TCC), Product Buyer (PB), Production Planner (PP), Transport Buyer (TB), Transport Planner (TP). Many possible options are available for the location of the different decision making agents. The customer agent might be a retailer or a producer. The TCC might be a planner within a larger company or a third or fourth party logistics operator. The PB is often connected to the organization which hosts the TCC, but can be independent when the TCC is a third party logistics operator. The PP belongs to the producing company. The TB might belong to the same organization as the customer or as the TCC. The TP typically belongs to the organization owning and controlling the transport carriers (Bergkvist et al., 2005).

LOGISTIC DECISIONS IN FREIGHT TRANSPORT

A disadvantage of many of the freight transportation models is that they are completely lacking elements of logistic organization (Ben-Akiva and de Jong, 2008). A better link with the freight distribution industry is required to overcome this weakness and some models have made progress in this respect by modelling logistic processes such as the number and location of distribution centres, the choice of shipment size and travel mode (Rand Europe, 2002). Furthermore, the choice of receiver or sender could also be modelled using disaggregate random utility models. This leads to the opportunity of simulating changes in the logistic chain, for example: these days many goods are delivered from distribution centres to the retailers, rather than from manufacturers (Kuzmyak, 2008). The delivery patterns that are optimal for distribution centres are different from when they were shipped directly by the producer. Those movements are often made by truck fleets whose travel is organized into tours with many stops.

One of the problems where firms are confronted with is the choice of an appropriate inventory level and transport mode. To make this decision most authors are referring to the inventory-theoretic model, which use the total logistic costs to determine which transport mode is most appropriate for the desired inventory level. This is done by taking into account all costs in the supply chain that are influenced by the mode choice. An integration of the inventory-theoretic concept may be found in the ADA model (Ben-Akiva and de Jong, 2008). Here the total logistic cost is used in the logistic module to make the 'transport chain choice', which is composed of shipment size and frequency, number of legs in the transport chain, transport mode and vehicle type. In Roorda et al. (2010) a similar approach to the ADA model is proposed.

In table 1 an overview is given of some of the main logistic decisions of the reviewed models. These models were examined whether they explicitly simulated shipment size, mode choice, vehicle type and the logistic distribution chain chosen. If one of these items is not represented in the model it is denoted with '0', if it is not know a '?' is inserted.

Table 1: Logistics Decisions

Model	Shipment size	Mode choice	Vehicle type	Logistic chain
Tavasszy et al. (1998)	0	X	?	X
SCENES Consortium (2000)	?	X	X	X
Yin et al. (2005)	?	X	X	X
Ben-Akiva and de Jong (2008)	X	X	X	X
Hunt and Stefan (2007)	?	0	X	0
Fischer et al. (2005)	X	X	?	X
Hunt (2003)	X	0	X	?
Boerkamps et al. (1999)	X	X	X	X
Liedtke (2009)	X	0	0	0
Wisetjindawat et al. (2007)	X	0	X	?
Davidsson et al. (2008)	X	X	X	0
Roorda et al. (2010)	X	X	X	X

Looking at the different transport modes used in the models of table 1, a major distinction that may be made is between the urban or more regional models and the national and international models. Where urban or regional models focus more on road and rail transport, the national models are defining more transport modes. Besides rail and road national models mostly also include inland waterways and transport via sea. In some cases even air transport is considered. An exception to this is the INTERLOG model (Liedtke, 2009) that only considers road transport although it is a national wide freight transport model.

Although the first inventory-theoretic models date from 1970 (Baumol and Vinod, 1970) and were able to state the importance of integrated consideration of logistics and transportation in decision making, latter developed models are lacking this logistical insight (Liedtke, 2009). The more recent developed models are again taking the interaction of logistics decisions and transportation into their development. It is crucial to take logistics decisions into consideration while modelling freight transport to come to a more realistic image of freight movement today. Still work has to be done to fully grasp the logistic impact on freight transport. The options for a comprehensive activity-based freight transportation framework are further elaborated in the next section.

OPTIONS FOR AN INNOVATIVE FREIGHT TRANSPORTATION FRAMEWORK

As stated earlier, there is a need for a more comprehensive model that includes logistical elements. The objective is to develop an activity-based micro simulated model, where the focus lies on the agents. Liedtke and Schepperle (2004) state that having a model for the transport of goods at a microscopic level, would be a significant improvement for transport forecasts and the assessment of policy measures at any point in process, due to its ability to map individual reactions.

First of all, the characteristics of freight transport have to be taken into account. The main characteristics are heterogeneity, physical factors, operational factors and dynamic factors (Ortúzar and Willumsen, 2001). When modelling at a micro level, it is possible to look at individual instead of aggregated flows. This gives the opportunity to include individual firm characteristics and detailed representation of commodity groups. When looking at single movements of goods, more information of a shipment may be represented that would go lost in aggregated data. Furthermore, production rates of firms may be included to take account for the changes in the demand pattern of customers, like in the TAPAS model (Davidsson et al., 2008).

As a starting point the relationship with the economy has to be included. Disaggregate models start from a detailed microeconomic background of the different commodity groups. The behaviour of shipper and carrier that is modelled helps to determine how much and in what way commodities will be moved. Transport can be considered as a part of the logistics process and a production factor. Companies consider their output as the arrival of finished goods at their destination. For this not only labour and capital is necessary, but also transport becomes important as production factor (Meersman and Van de Voorde, 2008). This allows the analysis of the relation between an economic activity and the resulting transport movement.

When developing an activity-based model great care has to be paid to the choice of agents involved in the model, as discussed earlier. The way these agents interact with each other and how they are involved in the decision making process is of key importance in developing a micro simulated activity-based model. This allows to include pricing mechanisms and to take into account long- or short-term contracts between agents. An opportunity exists to simulate market interactions and pricing negotiations. Furthermore, more attention has to be paid to logistic decision making. What are the responsibilities of each agent and on what may he have an influence?

By explicitly simulating the different agents involved in the decision making process of an activity-based model, the logistics decisions and chains may be represented. When it comes to logistical processes some main items have to be included, like modelling shipment size and an appropriate mode and vehicle type choice. It gives the opportunity to incorporate inventory management at the customer and vendor site, to include warehouse management at distribution centres and to simulate terminal operations. To optimize distribution chain flows the location of distribution centre may be included in the modelling process. A close follow up of all these interactions is requested to have a more realistic image of freight transport flows.

CONCLUSIONS

There is a great need for better forecast models of freight transport. Some attempts to improve model results have already been made and freight modelling is moving to more activity-based models, like in passenger transport. In this paper a contribution to this process is been made, by exposing the main issues in freight transport modelling. First, the different agents that are involved in the decisions making process were given. These agents are necessary to construct distribution channels and supply chains, which may facilitate freight transport. Secondly, the many logistics elements and decisions that come with this entire process were discussed and have to be taken into account when modelling freight flows. Furthermore, some recommendations have been made to arrive at a more comprehensive activity-based freight model.

Further research has to be conducted to study the possible agents that may be introduced and how they interact with each other. Also the possibilities for incorporating the logistic elements have to be examined. Finally, research had to be done to collect the necessary data.

REFERENCES

Baumol W. and Vinod H., 1970. *An inventory theoretic model of freight transport demand. Management Science*, 16, no. 7, 413–421.

Ben-Akiva M. and de Jong G., 2008. *The Aggregate-Disaggregate-Aggregate (ADA) Freight Model System.* In M. Ben-Akiva; H. Meersman; and E. Van de Voorde (Eds.), *Recent Developments in Transport Modelling: Lessons for the Freight Sector*, Emerald, UK. First ed.

Bergkvist M.; Davidsson P.; Persson J.; and Ramstedt L., 2005. *Multi-Agent and Multi-Agent Based Simulation*, Springer, vol. 3415, chap. A Hybrid Micro-Simulator for Determining the Effects of Governmental Control Policies on Transport Chains.

Boerkamps J.; van Binsbergen A.; and Bovy P., 2000. *Modeling behavioral aspects of urban freight movements in supply chains.* Transport research board, Washington. 79th annual meeting.

Boerkamps J.; van Binsbergen A.; Taniguchi E.; and Thompson R.G., 1999. *GoodTrip—A new approach for modelling and evaluating urban goods distributrion.* In T. E. and R. Thompson (Eds.), *City Logistics.* Institute of Systems Science Research, Kyoto, Japan, 175–186. 1st Internat. Conf. City Logistics.

Davidsson P.; Holmgren J.; A. Persson J.; and Ramstedt L., 2008. *Multi agent based simulation of transport chains.* In Padgham; Parkes; Müller; and Parsons (Eds.), *Proc. of 7th Int. conf. on Autonomous Agents and Multiagent Systems.* AAMAS 2008, Estoril, Portugal, 1153–1160.

Fischer M.; Outwater M.; Cheng L.; Ahanoto D.; and Calix R., 2005. *Innovative framework for modeling freight transportation in Los Angeles County, California. Journal of the transportation research board*, , no. 1906, 105–112. Transportation Research Board of the National Academies, Washington.

Hesse M. and Rodrigue J.P., 2004. *The transport geography of logistics and freight distribution. Journal of Transport Geography*, 12, 171–184.

Hunt J., 2003. *Agent Behaviour Issues Arising with Urban System Micro-Simulation. European Journal of Transport Infrastructure and Research*, 2, no. 3/4, 233–254.

Hunt J. and Stefan K., 2007. *Tour-based microsimulation of urban commercial movements. Transport research part B 41*, 981–1013.

Kuzmyak J., 2008. *Forecasting Metropolitan Commercial and Freight travel: A synthesis of Highway practice.* Tech. Rep. NCHRP Synthesis 384, National Cooperative Highway Research Program.

Liedtke G., 2009. *Principles of micro-behavior commodity transport modeling. Transportation Research Part E*, 45, 795–809.

Liedtke G. and Schepperle H., 2004. *Segmentation of the transportation market with regard to activity-based freight transport modelling. International journal of logistics: Research and applications*, 7, no. 3, 199–218.

Meersman H. and Van de Voorde E., 2008. *The Relationship between Economic Activity and Freight Transport.* In M. Ben-Akiva; H. Meersman; and E. Van de Voorde (Eds.), *Recent Developments in Transport Modelling: Lessons for the Freight Sector*, Emerald, UK. First ed.

MOTOS, 2006. *Transport Modelling: Towards Operational Standards in Europe.*

Ortúzar J. and Willumsen L., 2001. *Modelling Transport.* John Wiley & Sons, Chichester, UK, third ed.

Rand Europe, 2002. *Review of freight modelling.* Tech. rep., ME&P. Report B2 - Review of models in Continental Europe and elsewhere.

Roorda M.; Cavalcante R.; McCabe S.; and Kwan H., 2010. *A conceptual framework for agent-based modelling of logistics services. Transportation Research Part E*, 46, 18–31.

SCENES Consortium, 2000. *SCENES European transport forecasting model and appended module.* Technical description deliverable D4, ME&P, Cambridge. SCENES for the European Commission DG-TREN.

Tatineni V. and Demetsky M., 2005. *Supply chain models for freight transportation planning.* Research report UVACTS-14-0-85, University of Virginia.

Tavasszy L.; Smeenk B.; and Ruijgrok C., 1998. *A DSS for modelling logistic chains in freight transport policy analysis. International Transactions in Operational Research*, 5, no. 6, 447–459.

Wisetjindawat W.; Sano K.; Matsumoto S.; and Raothanachonkun P., 2007. *Micro-simulation model for modeling freight agents interactions in urban freight movement.* 86th annual meeting of the Transportation Research Board, Washington D.C.

Yin Y.; Williams I.; and Shahkarami M., 2005. *Integrated regional economic and freight logistics modeling, results from a model for the Trans-Pennine Corridor, UK.* Paper presented at the European transport conference 2005, Strasbourg.

MODELLING SHORTEST PATH DECISIONS USING AN ACTIVITY-BASED SEGMENTATION

Katrien Ramaekers, Mario Cools, Sofie Reumers and Geert Wets
Transportation Research Institute
Hasselt University
Wetenschapspark 5 bus 6
B-3590 Diepenbeek, Belgium
E-mail: katrien.ramaekers@uhasselt.be; mario.cools@uhasselt.be;
sofie.reumers@student.uhasselt.be; geert.wets@uhasselt.be

KEYWORDS: route choice modelling, shortest path, activity-based approach, trip purpose

ABSTRACT

The aim of this research is to identify the relationship between activity patterns and route choice decisions. The focus is turned to the relationship between the purpose of a trip and whether or not the shortest path is chosen for the relocation. The data for this study were collected in 2006 and 2007 in Flanders, the Dutch speaking and northern part of Belgium. To estimate the relationship between the choice for the shortest path or not and the corresponding activity-travel behaviour a logistic regression model is developed. The results point out that, when analyzing the relationship between the activities of the people and whether or not the shortest path is chosen, there is no significant influence by the activity-based segmentation. However, when the deviation from the shortest path is related to the activities people perform, a significant relationship is found. Furthermore, next to trip-related attributes (trip distance), also socio-demographic variables and geographical differences play an important role.

INTRODUCTION

To support policy makers, traffic and transportation models can be used to make better long-term decisions. On an international level, activity-based models have become the norm to model travel behaviour (Davidson et al., 2007). The most important characteristic of these models is that the travel behaviour of persons or families is a product of the activities that they wish or have to perform, procuring a more realistic description and a better understanding of people's travel behaviour. Because of these advantages, researchers and policy makers in the United States have switched from conventional models to activity-based models. Although this trend is most visible in the United States, a similar evolution can be noticed in Europe.

Governments require reliable predictions of travel behaviour, traffic performance, and traffic safety to support long-term decisions. A better understanding of the events that influence travel behaviour and traffic performance will lead to better forecasts and consequently policy measures that are based on more accurate data.

An important issue that the models should account for is the decision process that one undergoes when conducting a trip. One of many considerations is the route choice. Therefore, it is important to examine to what extent route choice is related to the type of trip. The term 'type of trip' indicates the purpose of the trip and submits a link to the pattern of human activities. The term 'route choice' includes the many attributes of the path chosen to conduct the trip and establishes the link with the behaviour pattern of an individual.

The aim of this research is to identify statistically significant relationships between activity patterns and the behaviour regarding route choice. The focus is on the relationship between the purpose of the trip and whether or not the shortest route is chosen.

Studies regarding the relationship between the purpose of the trip and the travel distance are frequently available. This indicates that travel distance is considered an important attribute of route choice. Ramming (2002) states that car travellers want to minimize their travel time, regardless of the purpose of the trip. Zhang and Levinson (2008), however, state that the shortest path is chosen less when the purpose of the trip is shopping or paying a visit rather than travelling to work or to a leisure activity. Goldenbeld et al. (2007) study the route choice of car travellers in the Netherlands. Almost half of the respondents indicates 'shortest path' as one of the main reasons for choosing a route. When the purpose of the trip is work, 'shortest path' is considered more important in route choice than for other trip purposes. In a study of Papinski et al. (2009), route directness is indicated as the second most important factor when choosing a route.

In addition, literature shows that other factors besides route attributes (personal, household and situational characteristics) play a role in the route selection, e.g. Bayarma et al. (2007), Zhang and Levinson (2008) and Scheiner (2009). Therefore, personal characteristics as age, gender, income, profession and province and the situational characteristic time of day are considered in the analyses.

In the next section, the data is described. In the third section, the adopted methodology approach is explained in further detail and the results are discussed. Finally, in the last section, the most important findings are summarised and directions for further research are highlighted.

DATA

The data for this study were collected in 2006 and 2007 in Flanders, the Dutch speaking and northern part of Belgium, in the context of a large scale survey, conducted on 2500 households in the study area. In this section, first, more information concerning the large scale survey is provided. Next, some of the important data processing steps are highlighted and the variables that are considered for the analysis are described.

Data Collection

Traditionally, travel surveys have been collected by paper and pencil or over the phone. The introduction of activity-based analysis, which prompts the need for considerably more detailed data on travel behaviour, identified the advantages of collecting activity or time use diary data (see Ettema et al. (1997) for an overview). At the same time, however, the use of diary data virtually precluded the use of telephone interviews and in addition substantially increased respondent burden and error proneness (see e.g. Dowling and Colman (1995) and Sun et al. (1995)). To avoid or at least reduce such error, computer-assisted diary instruments were developed.

The data for this study stem from a large scale activity-based data collection effort conducted on households since the household context, in which individuals operate, has a very strong influence on individuals' decisions, particularly when household resources are shared, there are shared household responsibilities and there are decisions that are made jointly by multiple household members. The survey used a mixed-mode survey design, using a PDA application on the one hand, and using traditional paper and pencil diaries on the other hand. Cools et al. (2009) demonstrated that the use of this mixed survey design turns out to be a suitable way of collecting detailed information about planned and executed activity-travel behaviour of households as the survey mode has no direct impact on the quantities investigated.

The PDA application, called PARROTS (PDA (Personal Digital Assistant) system for Activity Registration and Recording of Travel Scheduling) has been developed in such that respondents could easily provide information about their activity-travel behaviour (Bellemans et al., 2008). Whenever an activity or trip is registered in PARROTS, a number of attributes for this activity or trip were collected using a customized GUI. The most important activity and trip attributes PARROTS collected are: activity type, date, start and end time, location, mode of transportation, travel time and travel party. Besides PARROTS uses the integrated Global Positioning System (GPS) to automatically record location data. This combination of GPS and diary responses provides great insight into the route choice decision-making process (Papinski et al., 2009). Jan et al. (2000) showed that GPS is a viable tool to study travellers' route choice decisions as GPS can reveal important travel behavioural information that is impossible to discern with earlier conventional survey methods such as interviews, respondent-administered questionnaires, or driver simulators. Moreover, conventional methods have proved burdensome, time consuming, and error prone (Wolf et al., 1999).

Data Processing

In order to analyze the reported and recorded travel data, advanced post-processing is necessary to make the information usable for route choice modelling (Schuessler et al., 2010). In this research only displacements made by car are taken into account. Displacements made with any other mode of transport are filtered out of the database. Next, the GPS-data are compared to the data reported by the respondents in the diaries. If there is a mismatch between both data sources, the displacements are not used in the analyses since it is possible that the reported displacements are incorrect. Furthermore, only respondents that filled in all personal characteristics are considered because these characteristics are used in the analyses. Given the network that is used to analyze the trips is a national network, cross-border displacements are removed from the database.

The data processing step leads to a dataset containing car displacements on the Belgian road network for respondents of whom the personal characteristics are known and for whom the GPS-data is consistent with the data reported in the diaries. The dataset contains 1423 car displacements, made by 299 different respondents.

Data description

This study focuses on the relationship between the purpose of a trip and whether or not the shortest path is chosen. In this paragraph, the variables that are used in the analyses are described.

Roads are divided in three categories, following the functional road classification of Weijermars et al (2008), namely through-roads (primary roads), distributor roads (secondary roads) and access roads (local roads).

Furthermore, trip-related attributes are considered. In literature these attributes are often pinpointed as predominant variables including: trip purpose (de Palma and Picard, 2005), trip distances (Scheiner, 2010) and congestion (Jan et al., 2000). Five types of trip purposes are distinguished: work, leisure, shopping, home and other. Congestion is coded as a dummy taking value one for trips made during congested periods (6:00-9:00 and 16:00-19:00) and taking value zero during other periods of the day.

Besides trip-related attributes, other factors such as socio-demographic and geographical characteristics play an important role in the route selection, as discussed by de Palma and Picard (2005), Bayarma et al. (2007) and Li et al. (2005). Therefore, the personal characteristics age, gender, net personal income, profession and the geographical characteristic province are considered in the analyses.

METHODOLOGY

Recall that the focus of this study is to assess the relationship between the choice for the shortest path or not and the corresponding activity-travel behaviour. To estimate this relationship a logistic regression is developed (Agresti, 2002). In the previous section, an elaborate description of the considered variables is provided. To assess the significance of the various trip-related and non-trip related predictors, a type III analysis of the effects is made, displayed in Table 1.

There is no significant influence of the activity-based segmentation: there is no significant relationship between whether or not the shortest path is chosen and the activities people perform. Furthermore, in line with Parkany et al (2006), congestion has no significant impact on the modelled route choice decisions. In accordance with literature (see e.g. Abdel-Aty and Huang (2004) and Parkany et al (2006)), next to trip-related attributes (trip distance), also socio-demographic variables and geographical differences play an important role.

TABLE 1 Type III Analysis of Effects

Effect	DF	Chi-Square	P-value
Road type	2	0.317	0.853
Purpose	4	2.526	0.640
Distance	1	164.372	<0,001
Congested	1	0.159	0.690
Age	3	9.734	0.021
Sex	1	1.516	0.218
Profession	5	9.849	0.080
Net personal income	5	19.690	0.001
Province	4	17.912	0.001

The parameter estimates of the logistic regression model, presented in Table 2, provide more insight in the factors that explain whether or not the shortest path is taken. To detect potential multicollinearity problems the Variance Inflation Factors (VIFs) are calculated. In general, VIF values exceeding 10 indicate the presence of serious multicollinearity undermining the validity of the results (Marquardt, 1980). Other authors consider this boundary too liberal and suggest that the variance inflation factors should not exceed 4 (Montgomory and Runger, 2003). The VIFs for the model presented in this paper indicate that there is no problem of multicollinearity.

With regard to the trip distance, the parameter estimates indicate that the longer the trip distance is, the less likely one takes the shortest path for this trip. When the trip distance would increase by 1 km, the odds of travelling by the shortest path decreases by 14 % (the odds are multiplied by 0.86 (=exp(-0.150)).

Concerning the effect of the socio-demographic variables, one can observe the clear difference between 65+ and the remaining age categories. When analyzing the results with respect to profession, it is clear that the choice for the shortest path or not is influenced by the profession of the respondent. Regarding the net personal income one could note that income has a decreasing effect on the likelihood of choosing the shortest path. The odds of choosing the shortest path are 57.2% (the odds are multiplied by 0.428 (=exp(-1.005-(-0.157))) lower for the lowest income class when compared to the highest income class. Finally, the parameter estimates also show that interprovincial differences exist. The likelihood of choosing the shortest path is smallest for Limburg and largest for Flemish Brabant.

TABLE 2 Maximum Likelihood Parameter Estimates

Parameter	Est.	St. Er.	VIF
Intercept	1.850	0.302	
Road type			
- Primary (through) road	-0.084	0.154	1.403
- Secondary (distributor) road	-0.060	0.205	1.234
- Local (access) road	0.000	n.a.	n.a.
Purpose			
- Home	0.019	0.200	1.948
- Work	0.000	n.a.	n.a.
- Leisure	-0.139	0.220	1.702
- Shopping	0.214	0.234	1.655
- Other	0.055	0.221	1.724
Distance	-0.150	0.012	1.290
Congested			
- During peak	-0.052	0.130	1.057
- Off-peak	0.000	n.a.	n.a.
Age			
- 18-25	-0.311	0.367	1.880
- 26-40	-0.167	0.162	1.338
- 41-64	0.000	n.a.	n.a.
- 65+	0.998	0.347	1.271
Sex			
- Female	0.194	0.157	1.315
- Male	0.000	n.a.	n.a.
Profession			
- Blue-collar worker	0.325	0.279	1.255
- White-collar worker	0.000	n.a.	n.a.
- Independent	-0.276	0.319	1.120
- Student	0.218	0.482	1.618
- Not professionally active	-0.484	0.204	1.814
- Other	-0.214	0.290	1.217
Net personal income			
- 0-1250 Euro	-1.005	0.246	1.716
- 1250-1750 Euro	0.000	n.a.	n.a.
- 1750-2250 Euro	-0.296	0.189	1.737
- 2250-2750 Euro	-0.082	0.270	1.528
- More than 2750 Euro	-0.157	0.341	1.275
- No answer	-0.601	0.224	1.744
Province			
- Antwerp	0.000	n.a.	n.a.
- East Flanders	-0.176	0.203	1.513
- West Flanders	0.107	0.273	1.270
- Flemish Brabant	0.157	0.191	1.741
- Limburg	-0.608	0.195	1.655

In addition, the relation between the deviation from the shortest path and the corresponding activity-travel behaviour is studied. To estimate this relation a linear regression is developed (Neter et al., 1996). To assess the significance of the various trip-related and non-trip related predictors, a type III analysis of the effects is made, displayed in Table 3.

Important to underline is the importance of the activity-based segmentation: there is a clear relationship between the deviation of the shortest path and the activities people perform. Next to trip-related attributes (trip distance), also socio-demographic variables and geographical differences play a noticeable role.

TABLE 3 Type III Analysis of Effects

Effect	DF	Chi-Square	P-value
Road type	2	1.314	0.519
Purpose	4	10.493	0.033
Distance	1	943.490	<0,001
Congested	1	0.426	0.514
Age	1	4.816	0.028
Sex	1	0.586	0.444
Profession	5	19.945	0.001
Net personal income	5	6.180	0.289
Province	4	10.219	0.037

The parameter estimates of the logistic regression model, presented in Table 4, provide more insight in the factors that explain the deviation of the shortest path. Again, the Variance Inflation Factors (VIFs) were calculated to detect potential multicollinearity problems. The VIFs calculated for the model presented in this paper indicate that there was no problem of multicollinearity.

With regard to the trip distance, the parameter estimates indicate that the longer the trip distance is, the higher the deviation from the shortest path. When the trip distance would increase by 1 km, the deviation from the shortest path will increase by 112 metres. Concerning the effect of the socio-demographic variables, one can observe that if age increases, the deviation from the shortest path decreases. When analyzing the results with respect to profession, a clear difference between blue-collar workers and the remaining categories is observed. Finally, the parameter estimates also show that interprovincial differences exist. The deviation from the shortest path is smallest for Limburg and largest for Eastern Flanders.

DISCUSSION AND CONCLUSIONS

In this study the relations between route choice decisions (i.e. shortest path or not), activity patterns and other influencing variables have been assessed. When analyzing the relationship between whether or not the shortest path is chosen and the activities people perform, there is no significant influence of the activity-based segmentation. However, when the deviation from the shortest path is related to the activities people perform, a significant relationship is found. Furthermore, in accordance with international literature, next to trip-related attributes (trip distance), also socio-demographic variables and geographical differences play a noticeable role.

In future research, instead of studying the shortest path decision, it might be interesting to study the relationship between the fastest route and the activities of the people. Another potential pathway for further investigating route choice decisions might lie in the roots of more psychological underpinnings. Besides, factors describing the situational context such as weather conditions (Cools et al., 2010a, 2010b) or public holidays (Cools et al. 2007, 2009a, 2010) can also be taken into account. Moreover, future research should extent to other transport modes such as walking, bicycle use, public transport and carpooling.

TABLE 4 Maximum Likelihood Parameter Estimates

Parameter	Est.	St. Er.	VIF
Intercept	0.019	0.331	
Roadtype			
- Primary (through) road	0.146	0.130	1.401
- Secondary (distributor) road	0.105	0.179	1.232
- Local (access) road	0.000	n.a.	n.a.
Purpose			
- Home	0.199	0.164	1.949
- Work	0.000	n.a.	n.a.
- Leisure	0.573	0.182	1.700
- Shopping	0.259	0.196	1.656
- Other	0.198	0.185	1.720
Distance	0.112	0.003	1.285
Congested			
- During peak	0.073	0.111	1.056
- Off-peak	0.000	n.a.	n.a.
Age	-0.014	0.007	2.010
Sex			
- Female	-0.097	0.127	1.250
- Male	0.000	n.a.	n.a.
Profession			
- Blue-collar worker	-0.340	0.227	1.211
- White-collar worker	0.000	n.a.	n.a.
- Independent	0.623	0.265	1.119
- Student	0.318	0.365	1.353
- Not professionally active	0.418	0.194	2.174
- Other	-0.386	0.245	1.239
Net personal income			
- 0-1250 Euro	-0.029	0.212	1.690
- 1250-1750 Euro	9.000	n.a.	n.a.
- 1750-2250 Euro	-0.101	0.151	1.581
- 2250-2750 Euro	-0.229	0.223	1.448
- More than 2750 Euro	-0.419	0.285	1.230
- No answer	0.177	0.179	1.635
Province			
- Antwerp	0.000	n.a.	n.a.
- East Flanders	0.382	0.172	1.511
- West Flanders	-0.051	0.229	1.267
- Flemish Brabant	0.034	0.161	1.664
- Limburg	-0.198	0.161	1.590

REFERENCES

Abdel-Aty, M. A., and Y. Huang (2004). Exploratory Spatial Analysis of Expressway Ramps and Its Effects on Route Choice. *Journal of Transportation Engineering*, Vol. 130, No. 1, pp. 104-112.

Agresti, A. (2002). *Categorical Data Analysis*. Second ed. Hoboken, NJ: Wiley.

Bayarma, A., Kitamura, R., and Y. Susilo (2007). Recurrence of Daily Travel Patterns: Stochastic Process Approach to Multiday Travel Behavior. *Transportation Research Record: Journal of the Transportation Research Board*, *2021*(-1), 55-63. doi: 10.3141/2021-07

Bellemans, T., Kochan, B., Janssens, D., Wets, G. and H. Timmermans (2008). Field Evaluation of Personal Digital Assistant Enabled by Global Positioning System: Impact on Quality of Activity and Diary Data. *Transportation Research Record: Journal of the Transportation Research Board*, No. 2049, Transportation Research Board of the National Academies, Washington, D.C., pp. 136-143.

Cools, M., Moons, E., Bellemans, T., Janssens, D. and G. Wets (2009). Surveying activity-travel behavior in Flanders: Assessing the impact of the survey design. *Proceedings of the BIVEC-GIBET Transport Research Day*, VUBPress, Brussels, pp. 727-741.

Cools, M., Moons, E., Creemers, L. and G. Wets (2010a) Changes in Travel Behavior in Response to Weather Conditions: Do Type of Weather and Trip Purpose Matter? *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2157, pp. 22-28.

Cools, M., Moons, E. and G. Wets. (2007) Investigating Effect of Holidays on Daily Traffic Counts: Time Series Approach. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2019, pp. 22-31.

Cools, M., Moons, E. and G. Wets. (2009a) Investigating the Variability in Daily Traffic Counts 16 through Use of ARIMAX and SARIMAX Models: Assessing the Effect of Holidays on Two Site Locations. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2136, 2009, pp. 57-66.

Cools, M., Moons, E. and G. Wets. (2010) Assessing the Impact of Public Holidays on Travel Time Expenditure: Differentiation by Trip Motive. *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2157, pp. 29-37.

Cools, M., Moons, E. and G. Wets (2010b) Assessing the Impact of Weather on Traffic Intensity. *Weather, Climate, and Society*, Vol. 2, No. 1, 2010, pp. 60-68.

Davidson, W., Donnelly, R., Vovsha, P., Freedman, J., Ruegg, S., Hicks, J., Castiglione, J. and R. Picado (2007). Synthesis of first practices and operational research approaches in activity-based travel demand modeling. *Transportation Research Part A*, vol. 41, pp. 454-488.

de Palma, A. and N. Picard (2005). Route choice decision under travel time uncertainty. *Transportation Research Part A: Policy and Practice*, Vol. 39, No. 4, pp. 295-324.

Dowling, R.G. and S.B. Colman (1995). Effects of increased highway capacity: Results of household travel behaviour survey. *Transportation Research Record* 1493, pp. 143-150.

Ettema, D.F., Timmermans, H.J.P. and L. van Veghel (1997). Effect of data collection methods in travel and activity research. European Institute for Retailing and Services Studies, Eindhoven University of Technology, Eindhoven, The Netherlands.

Goldenbeld, C., Drolenga, J., and A. Smits (2007). *Routekeuze van automobilisten. Resultaten van een vragenlijstonderzoek* (No. R-2006-33) (p. 116).

Jan, O., Horowitz, A. and Z.-R. Peng (2000). Using Global Positioning System Data to Understand Variations in Path Choice. *Transportation Research Record: Journal of the Transportation Research Board*, No. 1725, Transportation Research Board of the National Academies, Washington, D.C., pp. 37-44.

Li, H., Guensler, R. and J. Ogle (2005). Analysis of Morning Commute Route Choice Patterns Using Global Positioning System-Based Vehicle Activity Data. *Transportation Research Record: Journal of the Transportation Research Board*, No. 1926, Transportation Research Board of the National Academies, Washington, D.C., pp. 162-170.

Marquardt, D. W. (1980). You should standardize the predictor variables in your regression models. *Journal of the American Statistical Association*, Vol. 75, No. 369, pp. 74–103.

Montgomery, D. C., and G. C. Runger (2003). *Applied Statistics and Probability for Engineers*, Fourth Ed. John Wiley and Sons, New York.

Neter, J., Kutner, M.H., Wasserman, W. and C.J. Nachtsheim (1996). *Applied Linear Statistical Models*. Fourth edn. Burr Ridge, IL: McGraw-Hill/Irwin.

Papinski, D., Scott, D.M., and S.T. Doherty (2009). Exploring the route choice decision-making process: A comparison of planned and observed routes obtained using person-based GPS. *Transportation Research Part F: Traffic Psychology and Behaviour*, *12*(4), 347-358.

Parkany, E., Du, J., Aultman-Hall, L. and R. Gallagher (2006). Modeling Stated and Revealed Route Choice: Consideration of Consistency, Diversion, and Attitudinal Variables. *Transportation Research Record: Journal of the Transportation Research Board*, No. 1985, Transportation Research Board of the National Academies, Washington, D.C., pp. 29-39.

Ramming, M.S. (2002). *Network Knowledge and Route Choice* (Submitted to the Department of Civil and Environmental Engineering in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Transportation). Massachusetts Institute of Technology. Retrieved November 12, 2009, from http://web.mit.edu/its/papers/ramming_phd_final.pdf

Scheiner, J. (2009). Social inequalities in travel behaviour: trip distances in the context of residential self-selection and lifestyles. *Journal of Transport Geography*, *In Press.*

Schuessler, N., Balmer, M. and K.W. Axhausen (2010). Route Choice Sets for Very High-Resolution Data. *Proceedings of the 89th Annual Meeting of the Transportation Research Board*. CD-ROM. Transportation Research Board of the National Academies, Washington, D.C.

Sun, A. Sööt, S., Yang, L. and E. Christopher (1995). Household travel survey nonresponse estimates: The Chicago experience. *Transportation Research Record* 1493, pp. 170-178.

Weijermars, W., Gitelman, V., Papadimitriou, E. and C. Lima de Azevedo (2008). Safety performance indicators for the road network.

Wolf, J., Hallmark, S., Oliveira, M., Guensler, R. and W. Sarasua (1999). Accuracy Issues with Route Choice Data Collection by Using Global Positioning System. *Transportation Research Record: Journal of the Transportation Research Board*, No. 1660, Transportation Research Board of the National Academies, Washington, D.C., pp. 66-74.

Zhang, L. and D. Levinson (2008). Determinants of Route Choice and Value of Traveler Information: A Field Experiment. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 086, pp. 81-92.

BIOGRAPHY

Katrien Ramaekers graduated as Master of Business Economics at the Limburg University Centre in 2002. In 2007, she obtained her Ph.D. in Applied Economic Sciences at Hasselt University. Currently, she is a post-doctoral researcher at Hasselt University and is working on the modelling of freight transport. She is a member of the Transportation Research Institute (IMOB) of Hasselt University.

Mario Cools obtained the degree of master in applied economic sciences at Antwerp University in 2004. In 2005 he graduated at Hasselt University as a master of science in applied statistics. Since August 2005 he has been working as a PhD candidate at the Transportation Research Institute of Hasselt University. In November 2009, he obtained his Ph.D. and now he is working as a post-doctoral researcher at the Transportation Research Institute (IMOB) of Hasselt University.

Sofie Reumers graduated as a master in transportation sciences at the Transportation Research Institute (IMOB) of Hasselt University in 2010.

Geert Wets received a degree as commercial engineer in business informatics from the Catholic University of Leuven in 1991 and a PhD from Eindhoven University of Technology in 1998. Currently, he is a full professor at Hasselt University where he is director of the Transportation Research Institute (IMOB).

Lead Time Analysis of Passengers and Baggage at Amsterdam Airport Schiphol

Tim ter Horst [1)], Jaap A. Ottjes [2)], Marianne N. van Scherpenzeel [1)] and Gabriel Lodewijks [2)]

[1)] Schiphol Group
Passenger Business Development
P.O. Box 7501 1118ZG
Schiphol, The Netherlands
e-mail: scherpenzeel_m@schiphol.nl

[2)] Delft University of Technology
Faculty of Mechanical, Maritime and
Materials Engineering
Mekelweg 2, 2628 CD, Delft
e-mail: j.a.ottjes@tudelft.nl

KEYWORDS

Discrete event simulation, stochastic simulation, TOMAS, airport logistics, required lead time.

ABSTRACT

This paper discusses the development of a stochastic simulation tool which is used to determine the required lead times of passengers and their baggage for different airport scenarios. The tool is applied to determine the impact of a specific development plan on the required lead times at Amsterdam Airport Schiphol. The output of the tool is a visualization of the simulation and required lead time distributions for different connections. By comparing these distributions the impact is determined.

INTRODUCTION

Amsterdam Airport Schiphol (AAS) is an important hub in the global aviation network. The time needed to transfer, the so called required lead time[1], is very important for the position of AAS with respect to other hub airports, as airlines can offer more and better connections. This is an advantage for transfer passengers, who prefer a shorter total travel time. Because of the same reason the required lead time is also important for originating passengers.

The expected growth of passengers at AAS has lead to an airport development plan called 'Master Plan Southern Development' (MPSD). This plan describes, amongst others, an expansion of the terminal with an additional pier, pier A (see Figure 1). Schiphol Group, the company which is exploiting AAS, is intending to use an Automated People Mover (APM) system as a transportation device between the root of pier A and the root of pier D in order to safeguard the required lead times of passengers. The baggage system will also be changed according the Master Plan Southern Development: a new baggage hall will be constructed and a high speed transportation line called the 'backbone' will be built to connect this new hall with the existing baggage halls.

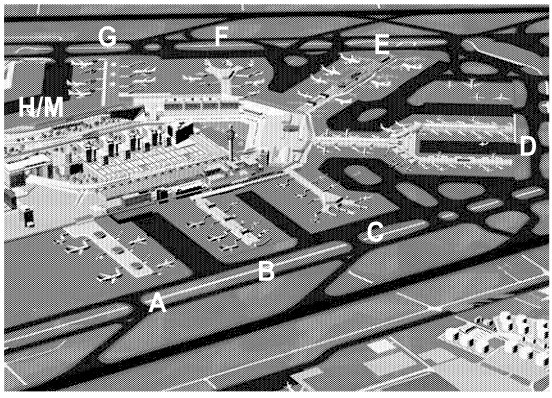


Figure 1 Artist impression of Amsterdam Airport Schiphol expanded with pier A

In order to determine the impact of the projects in the MPSD on the required lead times, it is needed

[1] Required lead time is defined as the lead time needed for a passenger or baggage piece from a specific location to another specific location. The time needed for optional processes (i.e. visiting shops or baggage buffering) is not included in the required lead time.

to gain insight in the individual required lead times of passengers and their baggage in both the current and in the future scenario. As these required lead times depend on many concurrent processes with time varying parameters and stochastic influences in the complex airport system, it is decided to design a tool based on discrete event simulation to determine this impact.

The structure of this paper is as follows: first the approach is discussed. This is followed by the system analysis, simulation tool development and results of some experiments. Finally the conclusions are presented.

APPROACH

The goal of this research is to determine the impact of the projects included in the MPSD on the required lead times of passengers and their baggage. The approach consists of analysis of the system, the development of a generic simulation tool and the execution of several experiments with different airport scenarios. Based on the results of the experiments conclusions are drawn.

ANALYSIS

In the analysis phase the required lead time standards, airport scenarios and simulation software are analyzed.

Required lead time standards
Schiphol Group is using standards with respect to the required lead times of originating and transfer passengers and baggage, namely respectively the Minimum Originating Time (MOT) and the Minimum Connecting Time (MCT). The time span of the MOT as well as the MCT can be found in Figure 2.

The numbers, corresponding to the MOT and MCT, depend on the (origin and) destination of the flight(s) in question. For confidential reasons, these numbers can not be discussed in this paper.

Airport scenarios
Both current and the future airport scenarios are analyzed in terms of passenger and baggage processes and their configuration, process characteristics, flight schedules, passenger occupations and transfer rates.

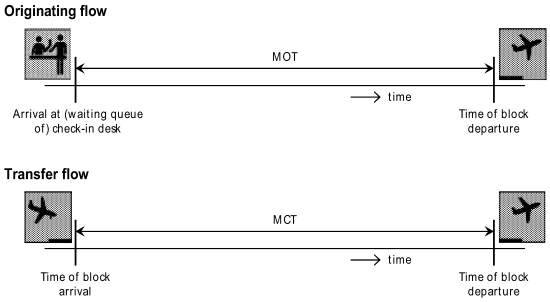


Figure 2 Visual representation of the MOT and MCT

The analyzed passenger processes include check-in, ticket check, security screening, passport control, customs, gate processes, (de)boarding, transfer desk use, baggage reclaim, walking and other transportation processes. Baggage processes included in the analysis are transshipment processes (baggage make up, loading and unloading of the aircraft and baggage unloading at the transfer unloading quay), system transportation and platform transportation.

Software selection
As stated in the introduction, a stochastic simulation model is needed to determine the required lead times of passengers and baggage in both airport scenarios. As it must be able to model different airport scenarios the simulation model should be generic. Due to the logistical nature of the complex airport system, a discrete-event simulation is chosen. After analyzing available simulation software it is decided to use TOMAS (Veeke, H.P.M., J.A. Ottjes, 2000) to model the passenger and baggage processes, as it is a discrete-event simulation tool which makes use of the process interaction modeling theorem (Zeigler, B.P, et al, 2000). This allows the active elements in the simulation to have a process, which makes the modeling process very communicable to non simulation experts (Ottjes, J.A., H.P.M. Veeke, 2002) Furthermore it is very flexible, as its implementation is based on the object-oriented programming functionality of Delphi.

SIMULATION TOOL DEVELOPMENT

Structure and explanation of the model
The process interaction modeling theorem contains the following three stages (Zeigler, B.P., et al, 2000):

1. Decompose the system into relevant classes of elements;
2. Identify the attributes of each element class;

3. Distinguish the element classes which have a process (the active elements)

The developed simulation model consists of the following active elements: aircrafts, passengers, bags and services. Examples of services are the check-in, security screening and boarding process for passengers and system transportation and baggage make up for bags. As the required lead time of every individual passenger and bag should be recorded, the required lead time is saved as an attribute of the passenger or baggage class.
Arrival flight elements create originating passengers with their bags at the departure hall, where departure flight elements create passengers before the deboarding service and bags before the baggage unloading service. Passengers and bags move from service to service with use of a route matrix. The appropriate service elements handle passenger and baggage elements.
The arrival time of a passenger at the departure hall is determined with use of arrival time distributions which depend on the airline, day of the week, time of the day and destination (Europe or intercontinental).
The staying times of passengers in the lounge are not included in the required lead time of passengers, but should be modeled in order to obtain realistic arrival patterns of passengers at the gate. Therefore the staying time is calculated with use of an arrival time at the gate distribution for flights with a European or an intercontinental destination.
As the exact transfer connections are unknown, arriving transfer passengers are connected to a departure flight using connection time data accumulated by continuous sampling connection times, (Schiphol Group, 2009) carried out under the authority of the marketing division of Schiphol Group. This is a random sample survey executed all year long at the gates. Approximately 40.000 transfer passengers are annually involved in this research. Furthermore, it is taken into account that transfer connections are only allowed for airlines within the same airline alliance.

Model input
The simulation model is implemented in a user friendly Delphi/TOMAS application.
The tool user interface includes a menu and several windows (see as example Figure 3 and Figure. 4). By means of this user interface services, routes and areas can be created, edited and removed.

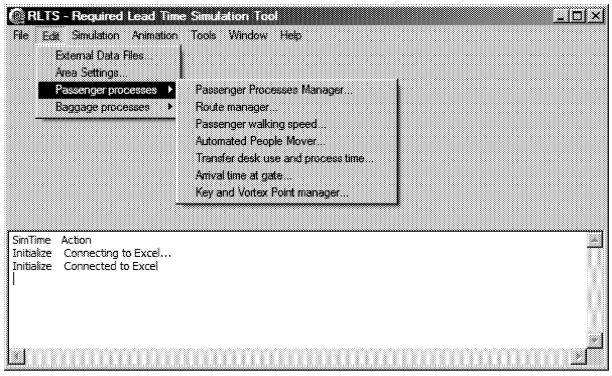


Figure. 3 Screenshot of the main window of the simulation tool

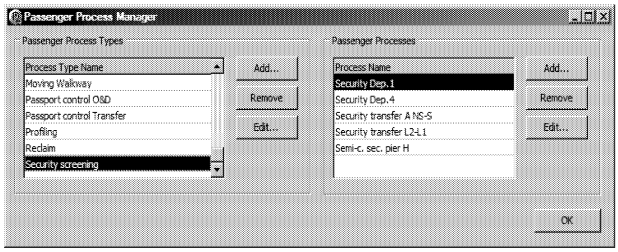


Figure. 4 Screenshot of the Passenger Service Manager window

The remaining input data (amongst others the flight schedule, aircraft stand planning and passenger occupations and transfer rates) is imported from Microsoft Excel files which can be specified in the simulation tool.

Model output
The simulation tool records the required lead times and their composition of all individual passengers and baggage pieces. Frequency distributions and their composition are saved into a Microsoft Excel file, which can be used for analysis.

Figure 5 Screenshot of the animation window

The passenger processes are visualized during the simulation run by means of a 2D-animation. Screenshots of the animation window can be found in Figure 5 and Figure 6.

Figure 6 Screenshot of the animation window

EXPERIMENTAL RESULTS

The simulation model is used to determine the required lead times of passengers and baggage in the current and the future airport scenario. As an example, the required lead time distributions of the passengers transferring within Europe can be found in Figure 7. Note that for confidential reasons the results in this paper are adapted and normalized.

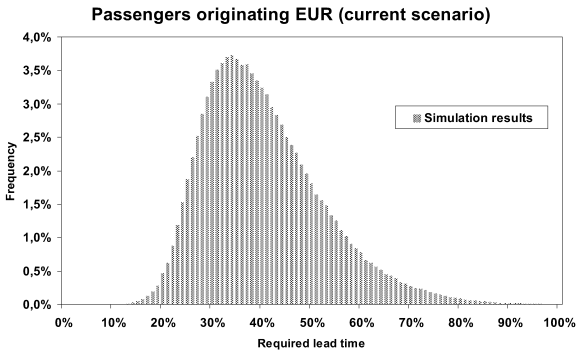


Figure 7 Required lead time distribution of originating passengers to an European destination (current airport scenario)

In order to be able to identify the process(es) which cause the lead times in the higher range, the tool generates graphs of the composition of the required lead times per time interval. An example can be found in Figure 8.

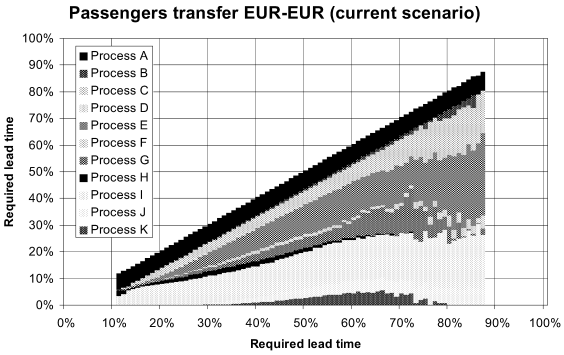


Figure 8 Required lead time composition of transferring passengers within Europe (current airport scenario)

By interpreting this type of graph, it shows that, for passengers having the most frequent required lead times, the time needed for process E and process I are the most time-consuming processes. The increasing waiting times of process C, E, G and K are the main cause of the higher required lead times of passengers.

The impact of the future scenario is measured by comparing the percentage of passengers and bags which exceed the lead time standards used by Schiphol Group, and the results have given useful insights for Schiphol Group.

CONCLUSIONS

A discrete-event simulation tool is developed which is able to simulate the passenger and baggage processes in a complex airport system, and generates required lead time distributions of a user specified airport scenario. The tool is used to determine the impact of a development plan at Amsterdam Airport Schiphol and the results have given useful insights as input for the decision process. The simulation tool can be used in the future for comparable problems at Amsterdam Airport Schiphol or other airports.

REFERENCES

Ottjes, J.A., H.P.M. Veeke, *Prototyping in process oriented modeling and simulation*. Proceedings of the 16th European Simulation Multiconference (ESM 2002), Darmstadt, June 2002.

Schiphol Group, *Continuous research connection times*. Schiphol, 2009.

Veeke, H.P.M., J.A. Ottjes, *TOMAS: Tool for Object-oriented Modeling And Simulation*. Proceedings of the Business and Industry Simulation Symposium (ASTC 2000), Washington D.C., April 2000.

Zeigler, B.P., H. Preahofer and T.G. Kim, *Theory of Modeling and Simulation: integrating discrete event and continuous complex dynamic systems*. 2nd edition, Academic Press, San Diego, 2000.

LOGISTICS SIMULATION

CHANGE TO GREEN IN INTRALOGISTICS

Orhan Altintas
Daimler AG
Rather Str. 51
D-40476 Düsseldorf, Germany
E-Mail: orhan.altintas@live.de

Cengiz Avsar
Daimler AG
Rather Str. 51
D-40476 Düsseldorf, Germany
E-Mail: cavsar@gmx.com

Matthias Klumpp
FOM Institute for Logistics and Service Management (ild)
Leimkugelstraße 6
D-45141 Essen, Germany
E-Mail: matthias.klumpp@fom-ild.de

KEYWORDS
Green Logistics, Green Intralogistics

ABSTRACT
This research contribution shows existing potentials in greening intralogistics as one major part in a green supply chain management concept. Different categories such as warehousing, buildings, layouts, transport systems and even IT are discussed and described contributing to an overall green intralogistics scheme. These outlines are enriched by an expert survey in intralogistics showing awareness among German automotive and machinery companies concerning green intralogistics concepts and measures. Though there are some hurdles and anxiousness about such green investments, a general trend towards such concepts in intralogistics is obvious and will attract further research and attention from manufacturing companies.

1. INTRODUCTION

Increasing economic growth and the accretive globalization lead to an increasing demand for transportation services. This causes a growing transport volume as well as an increase of ecological damages. At the same time ecological attention increases in society and people become aware of how entrepreneurial behavior effects the environment. Hence, it is necessary to change business strategy in order to accomplish an environmental responsible behavior and to keep up competitiveness simultaneously. To achieve sustainability it is necessary to pursue sub-goals as economic and social action (Lange, 2008, p. 41).
Only by regarding all three aspects (ecology, economy and social), short term aims can be reached. Economic sustainability is important to sustain natural livelihood for future generations. Therefore it is important to deal with the environment in a responsible way (Grunwald et al., 2006, p. 33).
Economic sustainability is reached by economic objectives, which ensure the long-term success of a company and the ability to compete (Pfohl, 2004, p. 151). Preservation of a future and worth living society counts to the aims of sustainability (Kraemer, 2008, p. 33). The three categories must be balanced. Thereby the weight of the categories cannot be reduced due to demand of another category. Unfortunately this is regarded quite often in practice: A concept of economy always meaning increasing input counts to the prejudices in business practice (Scharlau, 2009, p. 34).
The aspects of green logistics lead to an effective change. Because of that the question occurs which aims should be followed and if these categories can be combined. These can be subdivided and analyzed in the reduction of emissions and the reduction of costs as well as energy. The ability of a company to compete is depending on the ability to react to costumer's demands, which change faster and faster. This brings a much higher multiplicity of parts along. Through that demands towards production logistics increase, which has to achieve more with less resources.
An analysis of the warehouse transportation systems supplier 'VanDerLande Industries' showed that 24% of total logistics costs are caused by intralogistics (Kranke, 2008, p. 28). Because of that it has become obvious that an improvement on sustainability is necessary and would have positive impact.
This research bothers with the question, where rooms for improvements can be found in the categories of intralogistics and how they work out economically. The single rooms for improvements with the aims of a reduction of emissions, energy savings and savings of costs are analyzed and evaluated trough an expert survey.

2. POTENTIALS IN INTRALOGISTICS

In manufacturing it can be observed that the ability to compete with economical aims will be reached (Cansier, 1996, p. 278). Thereby all categories in production become obvious and can be seen as potentials. They refer to production processes, so that the manufactured products should have low costs.
A comprehensive analysis shows that further costs have to be regarded when calculating sales price. This includes the costs for resources, production expenditures, costs for development as well as transport costs. Intralogistics can be influenced directly and meets the categories hall layout, means of transport, stock, the assignment of new technologies respectively systems as well as transport packaging, which will be explained below.

2.1. Hall layout

In the area of intralogistics the layout of production halls determines logic costs extensively, which cannot always be influenced. Lean production approaches, which also demand optimal choice of locations for the particular steps of manufacturing, are preferable (Reuter, 2009, p. 63). This appendage can be pursued, when a new production location is planned on green grassland, for example. Here, locations can be chosen in an early stage, so that between particular production locations and warehouse only rare logistical activities are necessary. It is more difficult for already existing halls, which cannot be changed concerning their structural engineering. With the help of an analysis of value streams important spots can be found. The analysis of the value stream diagnoses the inventory level between two

subsequent manufacturing locations and generates measures for optimization (Liker et al., 2007, p. 74).
In assembly plants, which can be seen quite often in the automobile industry, the product is manufactured out of several modules of external suppliers. The routes of transport could be reduced dramatically, if suppliers would settle in industry parks (Klug, 2010, p. 16). For several reasons suppliers can't settle their headquarters in the proximity of assembly plants, so they accept longer transport distances.
A further reduction of total transport distances can also be reached by choice of production type as the assembly line work for example. Routes of transport can be optimized, when employees handle two manufacturing locations one after another instead of always processing equal parts at one location. These changes lead to the fact, that the processed part must not be transported by conveyance, because the employees transport the parts (Thonemann, 2009, p. 378).
Further potential develops by manufacturing lines, which are very close to each other. Here, the processed parts attain the next line by a common slide without extra energy (Reuter, 2009, p. 63).
Routes of transport can also be decreased by constructive changes on the product (Erlenspiel et al., 2007, p. 321). For example a module is produced with two instead of three parts, so that one manufacturing step can be omitted. With the help of such constructive changes also customer-friendly products, which have to be detected and considered, can be generated. For example, illuminists of flood lights can only be accomplished in garages.
One part of resource protection can also be an optimal arrangement of driveways. It is often the case that longer driveways have to be taken, because the direct ways are blocked through other manufacturing lines. Sometimes even optimally designed halls are changed for the worse by later extension. Because of that a new analysis regarding all driving ways is often necessary (Erlenspiel et al., 2007, p. 323).

2.2. Means of transport

2.2.1. Fixed feed systems

Continuous conveyor count to the in-plant transport systems and are generally used for transporting greater amounts of material or continually used materials on fixed routes.
The investment costs and costs of operation can be very high. Because of the fixed transport ways and difficult rebuilding possibilities they cannot be fitted to changing processes. Nevertheless, in the past years these systems have been further developed, so that flexibility is possible at low costs. The costs of operation could be reduced by new technological developments. For example the effectiveness of electric engines could be boosted and therefore a reduction of the energy use about 8% could be achieved, as it is shown in the article of VDMA (Association of German Mechanical Engineering Companies).
Moreover, the energy use can further be reduced by 12 % by the use of arranged actuations. Despite the high saving potential, currently only a third of the electrical actuations are arranged (Volz, 2008, p. 19).
Continuous conveyors generally move at constant speed. In practice, the continuous conveyors are so displayed that the requested limit load is covered and due to the constant use a fixed speed is received. Anyhow, there are times, when no goods have to be transported. In this period the use of the conveyor is not necessary and could automatically switch into standby mode with a low speed (Viastro, 2008).

Furthermore the speed of the upstream and downstream working zones should be proofed. As a result of that a reduction of speed can be deflected and therefore energy can be saved.
One additional point, which counts as potential by continuous conveyors, is the use of fully integrated discharges. For the division of material streams discharge elements as pneumatically processed pusher, which have a high requirement of compressed air and thereby have a high use of energy, are used in practice. To reduce this high use of energy meanwhile integrated solutions dominate, as lifted and canted role discharger for example. They require only 10% of the usual energy use (Materialfluss, 2008).

2.2.2. Conventional feed system

The forklift is one of the most well-known and frequently used discontinuous transport techniques in intralogistics. Possible green savings for forklifts could be achieved by the use of effective processes. This can be achieved by optimized transport processes, as well as by the use of new control strategies and systems. In automatic forklift systems industrial trucks are controlled via funk so that empty drives are avoided. Therefore energy consumption and by excess of a certain forklift fleet also the total number of forklifts can be reduced (Voigt, 2008, p. 36).
Stock costs could be lowered for about 10 to 20 percent in the long term, despite relatively high investments (Günthner, et al., 2009, p. 207).
Another saving concerning industrial trucks could be achieved by the use of environmentally friendly and efficient drive engineering. The deployment of alternative propulsion mechanisms is pushed in the sector of PKW since several years. In 1997 with the Toyota Prius, the first series of hybrid vehicles were brought on the market. After the change of the emission levels of work machines at the latest, this theme reached the attention of forklift producers. The industrial trucks can be suited for cross breeding because of their dynamic drive and load profiles, because of the fact that the percentage of stop and go process for short distances driving in a constant speed is very high (Biermann, et al., 1998, p. 2).
The manufacturers of mobile work machines develop vehicles with composite engines, fuel cells, hydrogen engines or free piston engine in doing so the concept Mild-Hybrid has got the highest market opportunities. The vehicle has got an additional combustion engine across an electronic engine, which can be used as a starter and generator at the same time. Trough the cross-linking of combustion and electronic engine a reduction of up to 25 percent can be achieved (Günthner et al., 2009, p. 208).
Beside the already mentioned approach there are further ideas, with which the need of energy and the adoption of material can be optimized. Trough the adoption of consumption-driven routing in contrast to the classical transports savings can be generated. The consumption-driven routing is linked with the Kanban method (filling method). Thereby small charge carriers are brought to the lines by trailer in fixed journey times. Trailers are small, movable drags, which are pulled by electronic vehicles, and which can be changed in their capacity by either changing the

amount of trailer or the amount of levels and height. The trailer drives between the stock area and consumer location (mostly in one hall). Route and time are fixed so that short ways and security of supply can be achieved with low stock (e. g. 1 time per hour). The charges of the trailer are mixed, which means that several part numbers are delivered by each driving round. The following advantages can be generated by their implementation (Baudin, 2004, p. 113; Takeda, 2002, p. 88):

- Movement at the assembly working spaces will be abated and thereby the contingent of added value increases.
- Lower costs trough less assets at the lines and in the hall.
- Improves the efficiency of the material supply trough the elimination of material movement.
- Decreases the amount of forklifts and thereby decreases costs.
- Improves the safety in the plant by correct organization.
- Supports the method of small carriers.

2.3 Stock keeping

The stock of manufacturing plants is part of the activities in the category of intralogistics. They contain potentials with great impact. As you can see on the basis of the analysis of VanDerLande Industries, about 50 % of intralogistics costs, in particular 35 % of heating- and ventilation engineering and 15 % of lighting engineering is caused by the storage area (Günthner et al., 2009, p. 206).

On closer inspection of these areas in most of the factories potentials do exist and could be changed without bigger efforts and investments.

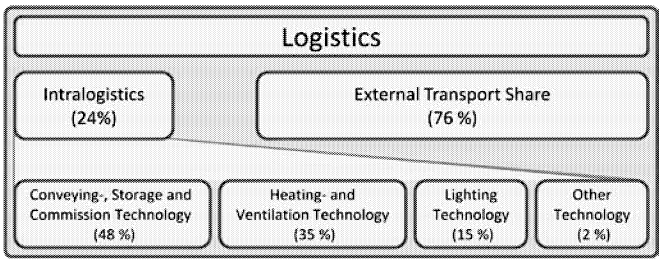


Figure 1: Energy consumption in logistics

2.3.1 Stock locations for articles and materials

Depending on the material group and the product it can be necessary to have several stocks available. One of the reasons for appliance of multiple stocks is the specific product which has to be stored. For example, for flammable materials strict rules have to be followed and thereby these have to be stored separately. Furthermore, an additional stock can be necessary, if several manufacturing halls exist and if these are located in great distance. At this point an integral analysis, which has to answer basic questions, is necessary. In practical it sometimes happens that production halls are planned and built up without considering intralogistics aspects.

When planning stock locations also external logistics should be regarded besides the intralogistics aspects. It would not be effective, if trucks of suppliers would have to drive long distances on factory premises, because this would interfere with internal transport. When choosing stock locations it is to make sure that ways are short and do not cross each other. During the last years the trend has moved towards high bay racking, which reduce the energy demand in combination with software applications (Arnold et al., 2008, p. 569). This is necessary because the demand of space increases, while at the same time available space decreases. This trend requires transport vehicles with other attributes, such as forklifts which can extract boxes out of a height of 20 meters. These vehicles are not appropriate for the transport from stock to production hall, because they have higher energy consumption due to their size. A partly automated feed system can help, by extracting the boxes out of all warehouse areas automatically after demand and paring it at a certain place in stock for the internal transport. The energy use can be reduced even more, if the stored articles are subdivided into material groups. For this purpose an ABC analysis can be used in consideration of the turnover ratio, whereas it should be reassessed continually based on the changing demand (Lasch, 2005, p. 259). Not all material groups are accessed in the same frequency, because for example the amount of boxes is not equal due to the geometry of the parts. Choosing the exact stock locations for particular material groups can reduce transport distances and therefore also energy use.

Through optimizing the boxes turnover ratio and therefore also the driveways can be reduced. Partly the amount of boxes can be increased about 10 to 20%, if these are parked differently (Jünemann, 2008, et al., p. 335). This method automatically reduces the driveways from supplier to factory.

There are also potentials, which do not directly result from driveways. These potentials concern necessary energies for ventilation, cooling, heating and lighting. The food industry requires stocks, which provide constant temperatures. These stocks are cooled with complex controls, so that the required temperature can be reached. The interface to the outside world is considered as disturbing factor, which always occur when new products have to be stored or taken out of the stock. Warm air from outside comes into the warehouse and makes additional cooling necessary. This mixture of air can partly be reduced trough the adjustment of alleys. Here, new zones between stock and the outside world are arranged. After the transport vehicle has entered this zone, the door to the stock opens not until this zone has been approach closed. The warm air can only mix with the volume of the small alley. The energy which is used for cooling is therefore lower (Günthner et al., 2009, p. 210).

Much energy can also be saved in the commission area for B and C articles in stock. For these goods was needed larger storage area. Nevertheless, the complete area is lighted, although fewer employees working there. The guiding idea of the effective design of intralogistics systems should avoid this and light should only be switched on in areas where commissioners work. This could be realized with the adoption of motion detectors as well as light sensors. Hereby up to 40% of energy can be saved for lighting (Günthner et al., 2009, p. 211).

The need for compliance of certain temperatures cannot be essential for certain stocks, if certain products do not have to be stocked. It is possible to establish a new area for those goods. Sometimes a heating of stock halls can be necessary, for example to reach an adequate temperature in winter. The costs to heat this hall are higher than they would be if the forklifts would have encapsulated cam assemblies which have to be heated. This assumes that only employees with forklifts can work in that hall.

In the range of lighting there are also potentials to reduce energy. In most of the stocks the lighting can only totally be switched of or on. Stock areas, which are passed over very rarely, only need an emergency light. At this point intelligent motion detectors could be useful to reduce the use of energy.

2.4 Green IT

According to a study of the federal environmental agency about 10% of the electric power consumption is spent by information and communication technique. Thereby about 33 million tons of CO_2 emissions are discharged each year. By using innovative and environmentally friendly IT-infrastructure savings of energy can be generated. In practice for nearly every application a separate server is installed, which only uses its own performance level. Moreover, each commission working space is arranged with its own computer system, which boosts the amount of computers. By using visualizations of servers and thin clients the amount of computer can be reduced drastically (Viastore, 2008).

Furthermore, with software supporting systems further can be captured, regarding the packaging of particular goods on the one hand, and transport or rather stock locations on the other hand. Therefore different solutions on the basis of a warehouse management system (WMS) are offered by different software suppliers. With simultaneous consideration of different variables this system assesses the optimal packaging for each product.

Not only the amount of air but also the optimal ways for transportation are regarded. That implies that packaging have to be optimized as far as possible, so that loading spaces can be utilized at an optimum. Therefore the sum of lengths and widths of the different packaging needs to match the length and width of the loading space.

By adoption of a WMS system an optimized loading area is achieved, which leads to a reduction of CO_2 emissions, as well as a reduction for packaging (mmlogistik, 2009).

3. EXPERT SURVEY

3.1. Method of collecting data

As part of the study, 11 logistic experts from different companies had been surveyed. When choosing the different companies, medium-sized and large enterprises with strong relationship to production and logistics activities had been focused. Based on the topic experts from the logistic division were chosen. Duration of employment, gender and age of the experts were not taken into account. The questionnaire was sent in form of a word document via e-mail to the experts. The questionnaire contains 8 (mostly) multiple choice questions. Therefore the required time to answer the questionnaire could be restricted to maximum 15 minutes. The answers of the different questionnaires were evaluated and graphed afterwards.

3.2. Results

To be able to make a final statement, the different questions have to be evaluated first. At the beginning some basic data about the surveyed experts are presented, which were requested with the first three questions of the survey.

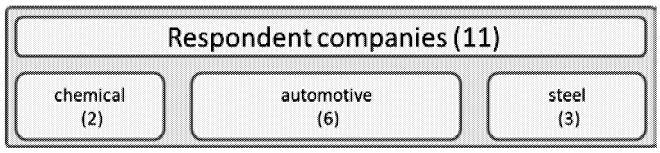


Figure 2: Expert group characteristics

The term ‚green logistics' is mostly associated with ecological awareness and a positive image for a company, as the answers demonstrate, followed by high investments and a negative influence on a company's profitability. Therefore it is to deduce that most companies would be willing to implement green logistics, but they are afraid of the necessary investment. This mental attitude is even intensified by the present state of the economy.

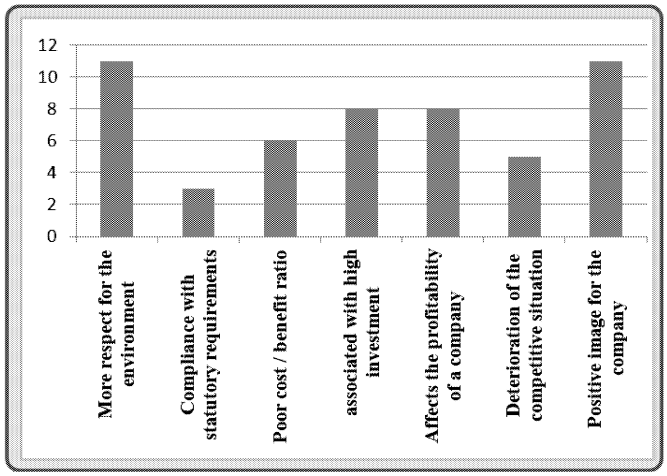


Figure 3: Sustainability motivation

This mental attitude is even mirrored in the answers referring to the *requirements* a structural change has to meet. Therefore an organization's willingness to put changes into practice depends on a positive cost-value-ratio and a short payback period.

It is noticeable that companies hesitate to conduct changes based on trend developments or costumer requests.

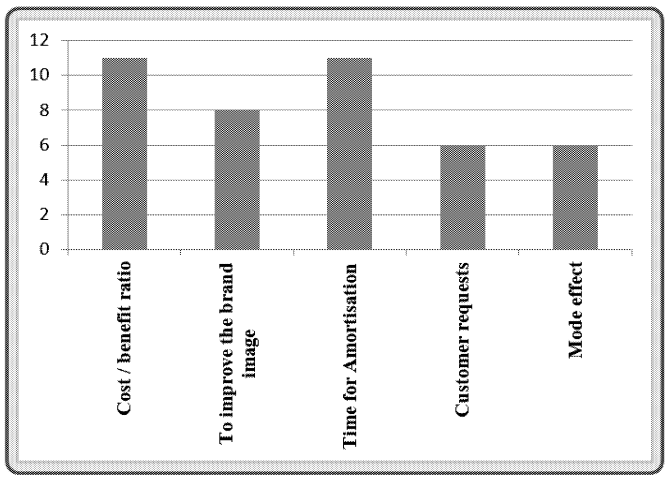


Figure 4: Decision criteria

Subject to constant conditions companies would be willing to implement green techniques, as it is to deduct from the different answers of one question. This statement supports the attitude towards measures which lead the company to an

environment-friendly image, which is thereby given a high priority.

Therefore, companies are basically willing to implement those 'green' measures. When the surveyed experts were asked in which fields of the company 'green' measures are implemented, they named the means of transport and storage field. In these fields existing potential is approached first, followed by the hall layout and Green IT.

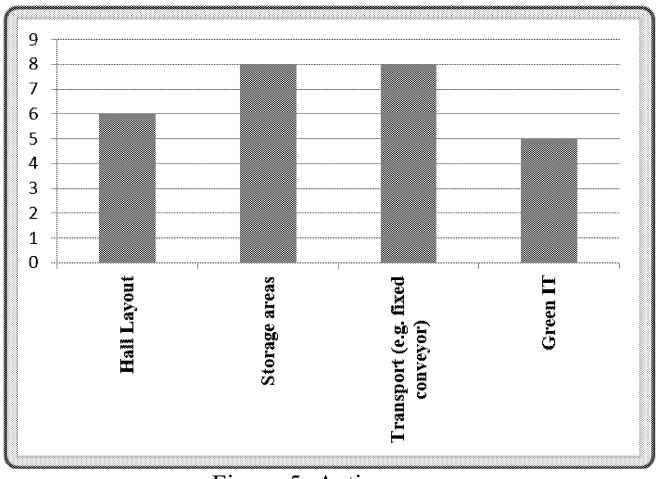


Figure 5: Acting areas

On closer inspection it is noticeable that measures in the different fields are prioritized differently, when they are supposed to meet ecological and economical demands. In particular measures which are supposed to lead to a decrease in transport route lengths are favored.

4. CONCLUSION

Especially in the field of intralogistics great potential for resource-saving and at the same time environmentally friendly processes are available. These are multifaceted and concern different fields. In the scope of this research some potential was identified and elucidated. Some of these potentials do not require large investments even though they do have a positive effect on a company's image and make savings possible. The answers of the survey lead to the conclusion that most companies are willing to put those measures into practice. Especially green logistics measures are favored to influence the company's image in a positive way and to support the environmental protection, but they are not classified as cost-saving measures – through they usually are.

On closer inspection it is noticeable that green logistics measures are even used to save costs. The payback period for these measures is not much worse than for alternative investments. Therefore the way to energy-saving and resource-saving processes is open. In the long run companies that go this way and implement these measures will stand out from their competitors, especially as resource prices are expected to rise sharply.

REFERENCES

Arnold, D., Isermann, H., Kuhn, A., Tempelmeier, H. and Furmans, K. (2008).“Spezielle Logistikprozesse”. In: *Handbuch Logistik*. Springer: Berlin and Heidelberg, 525-580.

Voigt, S. (2008). “Sparsamer stapeln”. In: LOGISTIK inside, 12/2008, 35-38.

Baudin, M. (2004). “Lean logistics: the nuts and bolts of delivering materials and goods”. Productivity Press: New York.

Biermann, J. W. and Bady, R. (1998). ”Hybridantrieb - Strukturvarianten, Betriebsstrategien sowie deren Vor- und Nachteil”. http://www.ika.rwth-aachen.de/forschung/veroeffentlichung/1998/26.-27.03/, date 02.06.2010.

Cansier, D. (1996). “Umweltökonomie” Lucius & Lucius Verlagsgesellschaft: Stuttgart.

Jünemann, R. and Schmidt, T. (2000). “Materialflußsysteme: Systemtechnische Grundlagen”. Springer: Berlin, Heidelberg and New York.

Erlenspiel, K., Kiewert, A. and Lindemann, U. (2007). “Kostengünstig entwickeln und konstruieren: Kostenmanagement bei der integrierten Produktentwicklung”. Springer: Berlin, Heidelberg and New York.

Günthner, W. A., Galka, S. and Tenerowicz, P. (2009). „Roadmap für eine nachhaltige Intralogistik“. In *Sustainable Logistics*, 14/2009, 205-219

Grunwald, A. and Kopfmüller, J. (2006). „Nachhaltigkeit”. Campus Verlag: Frankfurt/Main.

Klug, F. (2010). “Logistikmanagement in der Automobilindustrie: Grundlagen der Logistik im Automobilbau”. Springer: Heidelberg, Dordrecht, London and New York.

Kraemer, K. (2008). “Die soziale Konstitution der Umwelt”. VS Verlag für Sozialwissenschaft/GWV Fachverlage: Wiesbaden.

Kranke, A. (2008). “Effizienz statt Leistung”. In: LOGISTIK inside, 12/2008, 28-29.

Lange, H. (2008). “Nachhaltigkeit als radikaler Wandel: die Quadratur des Kreises?”. VS Verlag für Sozialwissenschaft, GWV Fachverlage: Wiesbaden.

Lasch, R. (2005). “Logistik Management: Innovative Logistikkonzepte”. Deutscher Universitäts-Verlag/GWV Fachverlage: Wiesbaden.

Liker, J. K., and Meier, D. P. (2008). “Praxisbuch: Der Toyota Weg für jedes Unternehmen”. FinanzBuch Verlag: München.

Materialfluss (2008). http://www.materialfluss.de/forder-und-hebetechnik/fordertechnik-und-komponenten/technik-trend-fordertechnik-energiesparen-beim-fordern-6/, date 08.06.2010.

mmlogistik, (2009). http://www.mmlogistik.vogel.de/lagertechnik/articles/ 22871/, date 15.06.2010.

Pfohl, H.-C. (2004). “Erfolgsfaktor Kooperationen in der Logistik: Outsourcing – Beziehungsmanagement – Finanzielle Performance”. Erich Schmidt Verlag: Berlin.

Reuter, C. (2009). “Logistikrelevante Lösungen auf der Basis von Lean-Management bei kleinen Losgrößen und hoher Variantenvielvalt”. Jost-Jetter Verlag: Heimsheim.

Scharlau, J. (2009) „Socially Responsible Investment” Schriften zum Europäischen und Internationalen Privat-, Bank- und Wirtschaftrecht Band 30. De Gruyter: Berlin.

Takeda, H. (2002). “Das synchrone Produktionssystem. Just-in-Time für das ganze Unternehmen”. Moderne Industrie Verlag: München.

Thonemann, U. (2009). “Operations Management: Konzepte, Methoden und Anwendungen”. Pearson Studium: München.

Volz, G. (2008). “Elektrische Motoren und Antriebssysteme“. Infoblätter Fördertechnik, DENA: Berlin.

Viastore (2008). www.viastore.de/.../Energieeffizienz_Hochregallage_Intralogistik,pdf, date 10.06.2010.

OPERATIVE SUSTAINABLE LOGISTICS MANAGEMENT SIMULATION

Matthias Klumpp
Sascha Bioly
Institute for Logistics and Service Management (ild)
FOM University of Applied Sciences
Leimkugelstraße 6

D-45141 Essen, Germany
E-Mail: matthias.klumpp@fom-ild.de

Alexandra Mai
Dachser GmbH & Co. KG
Memminger Str. 140
D-87439 Kempten, Germany
Hella Abidi
Dachser GmbH & Co. KG
Hansestr. 52,
D-51149 Cologne, Germany
E-Mail: hella.abidi@dachser.com

KEYWORDS
Sustainable Logistics, Operative Sustainable Logistics, Sustainable Logistics Simulation.

ABSTRACT
Sustainable logistics concepts currently lack an *operative* transmission scheme: Strategic and customer requirements are increasingly prompting green concepts but on an operative level still quality, service and especially cost criteria are usually valued more important than sustainability concerns. Therefore this research article argues that further management and simulation models have to be developed, tested and implemented in order to help operational decision making in transport chains. This research contribution helps in this development by suggesting an operative sustainable logistics management scorecard and matching this with operational data from the DACHSER company, Germany.

1. INTRODUCTION

In general green and sustainable logistics concepts are focused on *strategic* decisions as e.g. location and general transport mode decisions. But nevertheless for future improvements and concept development also the dimension of *operative* logistics decisions is an important field of improvement. A significant environmental impact in this decision arena can be assumed as in most cases environmental impact worsens as time pressure is increasing and speed needs to be enhanced.
Therefore the concepts of supply chain event management have to be evaluated regarding sustainable information and decision making. As a guiding principle such day-to-day decisions (even automated ones in Supply Chain Event Management [SCEM] systems) should consider sustainable aspects as for example the implicated CO_2 emission by change in transport modes (e.g. air instead of seaway). Therefore existing calculation models addressing transport mode comparisons with emission criteria (Carter et al., 2008; Klumpp et al., 2009; Seuring et al., 2008; Zelewski et al., 2009) have to be merged with existing SCEM concepts. This is the topic of the following research paper.

2. SUSTAINABILITY AND LOGISTICS

The development of concepts in supply and logistics management towards more sustainability is driven by a *multitude* of factors, e.g. p*olitical influences* as e.g. the Kyoto Protocol of 1997, *media influences* expecting data, concepts and reactions from companies in order to prove their sustainable management policy and m*anagement influences* integrating the expected future raw material prices driven by shortages in raw materials due to restricted resources.
Literature regarding logistics, supply management and supply chain management is in many cases *cost* driven (Wiedmann et al., 2008, p. 63), *quality* (Bogaschewsky et al., 2008, p. 244) and *risk* oriented (Goll et al., 2008, p. 150). Sustainability concepts are to date only implemented as *subfactors* in concepts within these three specific perspectives or for a specified industry sector (e.g. the food sector; Hamprecht, 2005, p. 2). Even optimization models with a per se *integrated* approach are missing sustainable parameters in their objectives (Kohler, 2008, p. 10).
From the literature it can be stated that the destinction between strategic and operative levels is not yet clearly established for sustainable logistics. Therefore table 1 is providing a first draft of such a distinction in order to provide further fields and topics of research. Operational sustainable logistics management deals with single transports, whether the decision time frame being short (event management) or long (contingency planning).

Table 1: Sustainable Logistics Dimensions

	Transport volume and interval: ***high/long***	Transport volume and interval: ***low/short*** (one-time)	
Decision timeframe: ***long***	Strategic Green Logistics (e.g. locations, transport mode)	**Green Logistics Contingency Planning (e.g. accidents)**	**Opertative Sustainable Logistics Management (OSLM)**
Decision timeframe: ***short***	Green Project Logistics Management	**Green Supply Chain Event Management (GSCEM)**	

3. INFORMATION MANAGEMENT IN SUPPLY CHAINS

Operational, event-driven systems allow the monitoring of stocks, orders and deliveries of goods along the supply chain. They identify expected events and unplanned incidents and inform the decision makers about their status with the aim of early identification of disorders and the states of emergency (Okhrin, 2008, p.111). Tracking and tracing of carriers and vehicles will be bundled provided and equipped with additional functions (e.g. warning functions) for better control and decision-support. Supply Chain Event Management requires for an efficient functionality a seamless flow of information within the corporate network. It allows the permanent monitoring of materials and goods flow along the entire value chain and implemented a coordinated management in the event of supply disruptions and emergency situations (Beckmann 2004, p. 113). The task of SCEM is an active and customer-oriented monitoring of the supply chain to the disturbances and variations in the value creation process in good time and to propose possible solutions. This increases the SCEM, the flexibility and responsiveness of the entire supply chain.

The SCEM is an interface between the created supply chain planning and pre-planned process and the course of the process in the operational process of supply chain execution (Arnold et al. 2008, p. 481). If deviations between the current actual state of the process and the planned course observed, the SCEM shall immediately initiate a series of reaction steps, which serve to address the malfunction and a planned continuation of the process and alternative solutions to pre-strike.

A SCEM system extends the functionality of tracking and tracing applications. The generic status messages are forwarded to the decision makers in real time. Later, the SCEM leads the target and actual analysis converts the signals into planned events or unplanned disruptions. The biggest advantage of the SCEM their transparency across multiple levels of the supply chain, since, ideally, all the individual processes are constantly monitored and controlled. However, making a high complexity and dynamics of the processes in a value chain, the effective implementation of the SCEM is a difficult task. SCEM has to realize thereby a permanent monitoring of material and goods flows along the entire chain and additionally has to make coordinated management action possible in case of supply disturbances and exceptional cases (Beckmann, 2004, p. 113). The task of SCEM is an active and customer oriented monitoring of the delivery chain to recognize disturbances and give possible solutions. Thus SCEM increases the flexibility and capacity of reaction of the entire supply chain.
The first theoretical bases of SCEM were already compiled in the form of elaboration about management by exception (MBE) in the middle of the last century, whereby beginnings of practical field use can be found in the bases of the tracking and tracing (Hunewald, 2005, p. 9; Wildemann, 2007, p. 13). Characteristic for the MBE approach is the fact of reducing control and steering activities of the responsible person. An intervention is only necessary if an event cannot be processed and/or settled independently by the SCEM system (Bittel et al., 1964, p. 5). With SCEM an interface is provided between the pre-defining supply chain planning (SCP), the planned process and the real operational sequence along the supply chain execution (Arnold et al., 2008, p. 481; Nissen, 2002, p. 477).
If deviations between the current actual condition of the process and the planned process are observed, SCEM introduces immediately a set of reaction measures, which serve for the recovery of the disturbances and a regular continuation of the process and/or suggest alternative solution types. Things like tracking and tracing, the traceability and on-line arrangement of goods, charge carriers and vehicles are bundled with additional functions (e.g. warning functions) for the better control and decision support provided. SCEM cannot replace the fundamental SCP but builds further on this (Bretzke, 2002, p. 28).
A SCEM system can extend the functionality of tracking and tracing applications. The generated status messages are passed on to the decision maker in real time. These of all participants of the supply chain collected data are supervised and interpreted by the event management system. In the further process SCEM accomplishes a comparison of nominal and actual values and designates the signals in planned events or unplanned disturbances (Klaus, 2004, p. 13).
This is the main task of the SCEM system. If the system registers an incident, thus a plan deviation from the defined specified condition, it tries to make a rapid reorganization of the process available on the basis of pre-defined solution alternatives. However this is only possible if a potential scenario is programmed in the event system and possible alternative solutions are implemented (Karrer, 2003, p. 188).

The biggest advantage of SCEM is transparency over several stages of the delivery chain, since ideally all individual shipments can be constantly supervised and steered. High complexity and dynamics of the processes in a value chain, however, make the effective implementation of SCEM difficult. Thus it is required that all processes are integrated along a supply chain in the event management system, because only with completely integrated and not partly omitted processes can an optimal reaction of the event management take place (Wildemann, 2007, p. 41).
As a condition for this a greatest possible information transparency along the entire supply chain has to be ensured (Wildemann, 2007, p. 44). In the course of constantly growing requirements of participants in the supply chain (shipper - service provider - customer) gains in particular the use of SCEM increasingly in meaning. Tracking and tracing, warning functions and further telematic components increase the quality and topicality of information about the whole supply chain. In direct consequence these tools permit optimal planning extending reactivity and effectiveness with deviations and exceptional cases.
SCEM can improve the efficiency and security of logistics processes. SCEM deployment optimizes the yield situation and customer satisfaction. The information gain concerning business processes secures a positive prognosis for the enterprise used by SCEM.

In order to *integrate* all the specified perspectives needed in a operative sustainable logistics a holistic management model is drafted in figure 1.

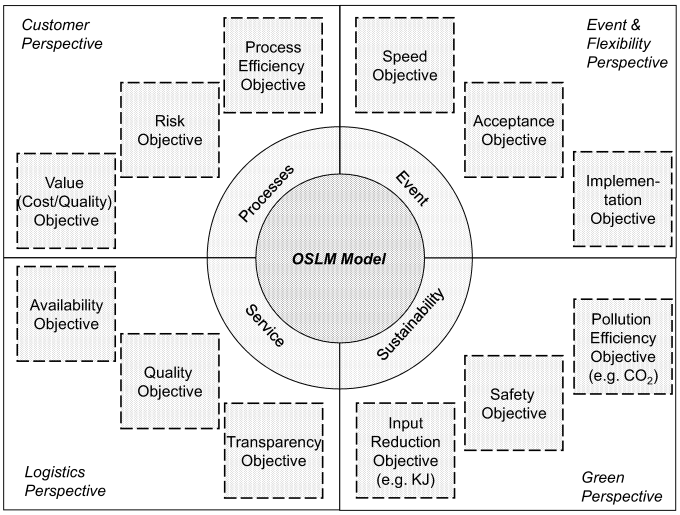


Figure 1: Operative Sustainable Logistics Management Model

First there is a *sustainability* or *green perspective* (Anderson et al., 2009; Archel et al., 2008; Darnall et al., 2008; Krause et al., 2009; Middendorf, 2008; Rodrigue et al., 2001; Straube et al., 2008) containing the following:

- *Input Reduction* objective calling for lower inputs of non-renewable materials as e.g. energy and raw materials needed for transport equipment and transport and logistics services.
- *Safety* objective describing the absence of harmful events such as oil and other dangerous goods spills in natural habitats or human injuries.
- *Pollution efficiency* objective determining a reduction of emissions of e.g. greenhouse gases or other pollutants in relation to logistics service outputs.

A *logistics perspective* (Bowen et al., 2001; Fleischmann et al., 1997; Tate, 1996) is underpinned by the following three important factors:

- *Availability* objective describing the basic function of logistics to ensure availability of the right goods at the right place and on time.
- *Quality* objective addressing the need for unharmed goods transport and smoothness of logistics *services* (service orientation, security awareness).
- *Transparency* objective in logistics depicting the aim to provide accurate and real-time information about transport, goods status and overall logistics performance for customers and other partners in the supply chain.

The *customer perspective* (Christopher et al., 2004; Giunipero et al., 2004) is described further by the following three objectives:

- *Value* objective addressing the ratio of costs and *product* quality in purchasing to be guarded and improved.
- *Risk* objective defining an overall risk management approach in order to avoid situations threatening company existence.
- *Process efficiency* objective determining the process time and internal process costs to be reduced in supply management e.g. by E-Procurement.

The *event and flexibility perspective* (Wagner et al., 2006; Wang et al., 2007) is outlined by the following three factors:

- *Speed* objective according to standard events in supply chains as usually disruptions in the transport chain create a need for higher speed.
- *Acceptance* objective mentioning that with increasing technology impact on all steps, persons and companies in a supply chain there has to be more emphasizing of acceptance.
- *Implementation* objective determining the fact that future technologies will need even more education and training efforts in order to fledge their full potential in supply chain event management (*implementation hurdle*).

All these factors together build an integrated view and might enable an integrated information modeling for logistics decisions – for the first time including sustainable decision facts. This could lead to more sustainable operational logistics decisions e.g. in transport modes and event reactions.

But still this model is missing quantitative measurement, simulation and controlling data. Therefore the authors developed in a logistics research project the following scorecard (figure 2), drafting operational measurement criteria for all the four perspectives of operational sustainable logistics management for the first time.

customer perspective		
objectives	measures	targets
process	customer satisfaction	school grade using survey better than 2,0
risk	% of all 'events'	< 5 %
cost	% cost overrun in LSP-responsibility	< 25 %

event & flexibility perspective		
objectives	measures	targets
speed	% on time in face of event	despite event on schedule > 50 %
implementation	total cost of all events in ratio to returns of all events	< 150 %
acceptance	average response time upon events	minimize, e.g. 23 h today => 21,5 h in future

logistics perspective		
objectives	measures	targets
quality	% on time of all	> 95 %
	low damage share	< 2 %
transparency	% of parcels in T+T- and/or SCEM-system	> 80 %
availibility	delivery time	e.g. 24 h run-time in germany / 48 h europe / 72 h worldwide

green perspective		
objectives	measures	targets
pollution	reduction absolute	minimize
	reduction below average	below 80 g (road transport)
output	certificate	ISO 14001, etc.
safty	net worth of all damages: damage costs / total turnover (alternative: per tkm)	< 1%

Figure 2: Operative Sustainable Logistics Management Scorecard

4. OPERATIVE DATA SIMULATION

The following operative transport data are used as an example in order to show an aggregated view as indicated on the OLSM model and scorecard draft above. The presented data are operational transport data from DACHSER in July 2010 concerning transports in Germany and Europe, usually consumer goods as e.g. coffee products or cosmetics. Interestingly the event descriptions provide a wide range of simulation and controlling requirements, e.g. from missing parts to consignees denying reception of goods. Today these events are handled on a single event "get it done" basis – but this should be changed by simulation and information systems as the scorecard drafted above in order to enable logistics management to react on a strategic level though operational transports are concerned. This could increase economic and sustainability gains in the supply chain.

ActiveReport DACHSER

Type Report Event	Origin of report		Division	Product	Consignee			Consignor				Order date	Quantity packaging	External remarks
	Date	Time			NC	ZIP	City	Name	NC	ZIP	City			
refusal of acceptance	19.7.10	12:19	Food Logistics	targospeed 12	D	81829	MUENCHEN	LAVAZZA LUIGI DEUTSCHLAND GMBH	D	60598	FRANKFURT	16.7.10	1 euro pallet	One CC was refused cause wrong goods has been delivered. A subsequent good will be delivered on the 21 of july till 9 o´clock a.m
New deadline arranged	20.7.10	12:00	Food Logistics	classicline	D	92355	VELBURG	LAVAZZA LUIGI DEUTSCHLAND GMBH	D	60598	FRANKFURT	19.7.10	49 carton	The consignee wishes the delivery on the 22 of july
Partial delivery	20.7.10	11:46	Food Logistics	targospeed 12	D	87534	OBERSTAUFEN	LAVAZZA LUIGI DEUTSCHLAND GMBH	D	60598	FRANKFURT	16.7.10	21 carton	One Item has not been delivered cause of a mistake by the Order Picking. We will deliver the item on the 17 th of july at the arranged time slot of the consignee
Consignment does not have any cartage note status	21.7.10	10:04	European Logistics	targospeed	D	21339	LUENEBURG	NORA SYSTEMS GMBH	D	69469	WEINHEIM	20.7.10	1 one way pallet	Shipment will be delivered in the afternoon
Complete deficiency	21.7.10	10:53	European Logistics	targospeed	D	35091	COELBE	NORA SYSTEMS GMBH	D	69469	WEINHEIM	20.7.10	1 one way pallet	When would you send us the shipment- Or should we cancel this Order? Many thanks for your soon reply in advance
Booking in (Avis)	21.7.10	02:23	European Logistics	targospeed	D	45770	MARL	NORA SYSTEMS GMBH	D	69469	WEINHEIM	20.7.10	1 one way pallet	The shipment arrived too late in our branch for the delivery in time. We would like to book in the shipment for the delivery today by an express to our charge or for the delivery tomorrow in the morning.

Figure 3: Operative Sustainable Logistics Data

ActiveReport is developed by Dachser and it is among to an innovative quality tool. This supply chain event management instrument is used in the practical experience to show and report proactively and directly every irregularity in the transport and logistic process; thereby the quality in logistic chains will be increased.

In fact each shipment is monitored continuously during the whole transport process. In the case of any difference for ex. Refusal of delivery, incorrect quantities or wrong delivery address, the Active Report tool automatically create a report about the deviation in real time, so that the shipper and Dachser staff can take corrective measures immediately. The Advantages for the customer with ActiveReport are:

- It increases the quality entire the logistic chain
- All the shipments are continuously monitored
- Transparence during the process chain
- Corrective measures can be taken immediately
- The customer can define precisely at the product level which information are relevant for them.

Example 1 (shipment for Marl; the latter in the table)**:** on 20 July the transmission fetched from the sender for the feed to 21.07. (in Marl). To 21.07. around 2:23 a ActiveReport originated in, since the vehicle did not arrive in the receipt address at 2 o'clock. (The feed is endangered). The Dachser coworker (the nightshift) examine, when the vehicle in the receipt address arrives and these inform. In this case the vehicle arrived after 8 o'clock. The feed cannot be accomplished. The outlet address informs the sender about the forthcoming run time excess and offers to the customer the further setting possibilities, like feed by special trip on the same day or feed on the next day for beginning of work. (This is the service of the Dachser). The sender examines, as hasty the transmission is and communicates his decision to the Dachser coworker.

Example 2: A shipment for Hamm could not be delivered, since the address is not correct. The driver enters this difference locally and transmits the data by GPRS to Dachser. At this moment a ActiveReport is provided automatically. The receipt address examines whether it the address via Internet or local directory. Directories to determine can order and the second feed. If the new address in the proximity is and the route is assigned. In this case was the address in another geographical place. The outlet address informed the sender about the setting difference and asked for further order. On the same day the new address was conveyed. These instructions were conveyed to the receipt address. This arranged the forwarding of the transmission to the responsible receipt address on the same day. The feed took place on the subsequent day.

5. OUTLOOK AND FUTURE DEVELOPMENT

The described OSLM simulation research brought the following main results:

- A concise management model for operational sustainable logistics is necessary and drafted in this article.
- The general model needs an outline in quantitative measurement data in order to be of value to logistics management and decisionmakers.
- Operational data from the logistics service provider DACHSER showed the wide range of events in operational transports and therefore the dire need to aggregate these information into management information, simulation and decision systems.

There may be several options for further research e.g. testing the suggested objective data in the OSLM model perspectives regarding their operational value in practice. Some parameters may have to be change to to company context, specific business area or even country and regional location

and position in the supply chain (e.g. OEMs as Volkswagen: Koplin et al., 2007).
Further research may also extend the view of the four provided perspectives and the three objective areas within the perspectives if necessary.

Grant Support Notice: The research for this article was supported by research funding in the project *LOGFOR* (German NRW Department MWME as well as EU - ERDF) and WIWELO (BMBF).

REFERENCES

Anderson, M., and Skojett-Larsen, T. (2009). "Corporate Social Responsibility in Global Supply Chains". In *Supply Chain Management - An International Journal*, 14 (2), 75-86.

Archel, P., Fernandez, M., and Larringa, C. (2008). "The Organizational and Operational Boundaries of Triple Bottom Line Reporting: A Survey. In *Environmental Management*, 41, 106-117.

Arnold, D., Isermann, H., Kuhn, A., Furmans, K., and Tempelmeier, H. (Eds.) (2008). *Handbuch Logistik*. Springer: Berlin.

Beckmann, H. (Ed.) (2004). *Supply Chain Management*, Springer: Berlin.

Bittel, L. R., and Maynard, H. B. (1964). *Management by Exception: Systematizing and simplifying the managerial job*. McGraw-Hill: New York.

Bogaschewsky, R., and Müller, H. (2008). „Stand und Weiterentwicklung des E-Procurement in Deutschland". In *Best Practice in Einkauf und Logistik, 2nd edition*. BME (Eds.). Gabler: Wiesbaden, 237-254.

Bowen, F. E., Cousins, P. D., Lamming, R. C., and Faruk, A. C. (2001). "The role of supply management capabilities in green supply". In *Production and Operations Management,* 10 (2), 174-189.

Bretzke, W.-R. (2002). "SCEM – Entwicklungsperspektive für Logistikdienstleister". In *Supply Chain Management,* 2 (3). 27-31.

Carter, C. R., and Rogers, D. S. (2008). "A framework of sustainable supply chain management: Moving toward new theory". In *International Journal of Physical Distribution & Logistics Management,* 38 (5), 360-387.

Christopher, M., and Lee, H. (2004). "Mitigating supply chain risk through improved confidence". In *International Journal of Physical Distribution & Logistics Management*, 34 (5), 388-396.

Darnall, N., Jolley, G. J., and Handfield, R. (2008). "Environmental Management Systems and Green Supply Chain Management: Complements for Sustainability?". In *Business Strategy and the Environment,* 18, 30-45.

Fleischmann, M., Bloemhof-Ruwaard, J. M., Dekker, R., van der Laan, E. A., van Nunen, J. A. E. E., and van Wassenhove, L. N. (1997). *Quantitative models for reverse logistics: A review, European Journal of Operational Research*, 103, 1-17.

Goll, L., and Haupt, S. (2008). "Corporate Governance, Risk- and Compliance Management in der Beschaffung". In *Best Practice in Einkauf und Logistik*. BME (Eds.). Gabler: Wiesbaden, 149-168.

Giunipero, L.C., and Eltantawy, R.A. (2004). "Securing the upstream supply chain: a risk management approach". In *International Journal of Physical Distribution & Logistics Management*, 34 (9), 698-713.

Hamprecht, J. (2005). *Sustainable Purchasing Strategy in the Food Industry*. DiFo: Bamberg.

Hunewald, C. (2005). *Supply Chain Event Management – Anforderungen und Potentiale am Beispiel der Automobilindustrie*. DUV: Wiesbaden.

Karrer, M. (2003). "Supply Chain Event Management – Impulse zur ereignisorientierten Steuerung von Supply Chains". In *Innovationen im E-Business, series Innovative Produktion und Logistik*. Dangelmaier, W. (Ed.). Vol. 10, ALB-HNI: Paderborn, 187-198.

Klaus, O. (2004). "Geschäftsregeln im Supply Chain Event Management". In *Supply Chain Management,* 4 (2), 13-19.

Klumpp, M., Zelewski, S., and Saur, A. (2009). "Increasing Rail Cargo Transport Performance". In *High-Performance Logistics*. Blecker, T., Kersten, W., and Meyer, M. (Eds.). ESV: Berlin, 17-30.

Kohler, K. (2008). *Global Supply Chain Design*, CfSM: Estenfeld.

Koplin, J., Seuring, S., and Mesterharm, M. (2007). "Incorporating sustainability into supply management in the automotive industry – the case of the Volkswagen AG". In *Journal of Cleaner Production,* 15 (2007), 1053-106.

Krause, D. R., Vachon, S., and Klassen, R. D. (2009). "Special Topic Forum on Sustainable Supply Chain Management: Introduction and Reflections on the Role of Purchasing Management". In *Journal of Supply Chain Management,* 45 (4), 18-25.

Middendorf, K. (2008). "Logistik im Spannungsfeld zwischen Globalisierung und Nachhaltigkeit". In *Das Beste der Logistik*. Baumgarten, H. (Ed.). Springer: Berlin, 405-414.

Nissen, V. (2002). „Supply Chain Event Management". In *Wirtschaftsinformatik*, 44 (5), 477-480.

Okhrin, I. (2008). "Supply Chain Event Management". In *Handbuch Logistik*. Arnold, D., Isermann, H., Kuhn, A., Furmans, K., Tempelmeier, H. (Eds.) Gabler: Berlin, 481-485.

Rodrigue, J.-P., Slack, B., and Comtois, C. (2001). "Green Logistics (The Paradoxon of)". In *The Handbook of Logistics and Supply-Chain Management*. Brewer, A. M., Button, K. J., and Hensher, D. A. (Eds.) Pergamon: London, 339-350.

Seuring, S., and Müller, M. (2008). "From a literature review to a conceptual framework for sustainable supply chain management". In *Journal of Cleaner Production,* 16/2008, 1699-1710.

Straube, F., and Borkowski, S. (2008). *Global Logistics 2015+ - How the world's leading companies turn their logistics flexible, green and global and how this affects logistics service providers*. BVL: Berlin.

Tate, K. (1996). "The elements of a successful logistics partnership". In *International Journal of Physical Distribution & Logistics Management,* 26 (3), 7-13.

Wagner, S.M., and Bode, C. (2006). "An empirical investigation into supply chain vulnerability". In *Journal of Purchasing & Supply Management*, 12, 301-312.

Wang, E.T.G., and Wei, H.-L. (2007). "Interorganizational Governance Value Creation: Coordinating for Information Visibiliy and Flexibility in Supply Chains". In *Decision Sciences*, 38 (4), 647-674.

Wiedmann, H., and Teichmann, J. (2008). "Next Level Purchasing: Erfolgsfaktor eines aktiven Kostenmanagements". In *Best Practice in Einkauf und Logistik, 2nd edition*. BME (Eds.). Gabler: Wiesbaden, 56-65.

Wildemann, H. (2007). *Event Management in der Supply Chain – Leitfaden zur Steuerung geplanter und zufälliger Ereignisse entlang der Supply Chain*. TCW: München.

Zelewski, S., Klumpp, M., and Saur, A. (2009). "Leerfahrtenoptimierung und Kapazitätserweiterung durch Kooperationen von Eisenbahnverkehrsunternehmen". In *Höchstleistung im spurgeführten System*. Schütte, J. (Ed.). Eigenverlag: Dresden, 1-29.

IN-PLANT LOGISTICS SYSTEMS MODELING WITH SYSML

Veronique Limère
Ghent University
Technologiepark 903
B-9052 Ghent-Zwijnaarde, Belgium
E-mail: Veronique.Limere@ugent.be

Sarath Balachandran
Georgia Institute of Technology
765 Ferst Drive, NW
Atlanta, GA 30332, USA

Leon McGinnis
Georgia Institute of Technology
765 Ferst Drive, NW
Atlanta, GA 30332, USA

Hendrik Van Landeghem
Ghent University
Technologiepark 903
B-9052 Ghent-Zwijnaarde, Belgium

KEYWORDS
SysML, systems modeling language, facility logistics, kitting, bulk feeding, cost model.

ABSTRACT

Up till now Systems Modeling Language (SysML) has mostly been used to model physical systems of interest. This paper shows how SysML can also be used to represent an abstract model. In this application a mathematical cost model is represented using the SysML plugin for the software MagicDraw. ParaMagic, a plugin in MagicDraw supplementary to SysML, links to Mathematica to solve the model. SysML is a formal language and offers a very intuitive graphical representation. It is therefore a useful medium to create a domain specific language for a field of knowledge. The comprehensiveness of the language, which makes it possible to incorporate specification, analysis, design, verification, and validation of systems, makes it a very valuable tool for collaboration on large projects.

INTRODUCTION

SysML is a relatively new systems modeling language which provides the syntax for very expressive models of systems. Friedenthal et al. (2008) define Systems Modeling Language (SysML) as follows:

SysML is a general purpose graphical modeling language for systems engineering applications. It is a dialect of UML™, the industry standard for modeling software-intensive systems. It supports the specification, analysis, design, verification and validation of a broad range of complex systems, including hardware, software, information, processes, personnel, and facilities.
(Friedenthal et al., 2008)

Up till now, the focus for SysML applications has been mainly on modeling physical systems of interest and not so much on modeling abstract systems and models. In this project SysML is used to graphically represent a mathematical cost model. We deal with a compact cost model for the minimization of in-plant logistics costs for feeding parts to an assembly line. To the best of our knowledge, no such application has been published before. Block definition diagrams are used to represent the structure of the model and parametric diagrams are used to model the equations. The diagrams and their usage will be extensively discussed in two separate sections about structure and parametrics.

MagicDraw is the architecture modeling environment which is used to implement the model. SysML is packaged as a plugin to the MagicDraw tool and supports all SysML diagrams. SysML Parametrics are further supported by an additional plugin, ParaMagic, which enables computations directly from the SysML model.

PROBLEM STATEMENT AND SCOPE

Nowadays, customers put a lot of pressure on the market to obtain timely delivery and low prices. In addition, more and more variation in the product assortment is demanded and custom-made products are often requested. This current trend is explicitly perceivable in the automotive industry. Each single vehicle that comes off the line is equipped with the proper options requested by the customer. No two vehicles coming off the line consecutively are identical. This evolution towards more customization has major consequences for production organizations and their logistics department. Components do not only exist in a single variant but alternative variant parts have to be assembled. This leads to an increasing number of parts moving around on the shop floor and undoubtedly to a more complex material supply process.

In practice, different methods of material supply are practiced in the vehicle industry. Bulk feeding is the most straightforward method of line feeding. However in the automotive industry parts are often large and voluminous. Putting a full packaging unit – i.e. container or pallet or box – of each part number in the border of line would require an enormous production area and is therefore not feasible. Moreover line-operators would have to look too far to find the correct components to be assembled which would lead to a decreased productivity. To address the need for diminishing line stocks and a better organized border of line in order to facilitate the operator's task, kitting is introduced as a counterpart of bulk line feeding. To introduce the reader to the problem setting, and for the understanding of the SysML model we shortly define the two main methods of materials supply.

Bulk feeding
Feeding a line in bulk means that no parts are yet assigned to a certain end product when they are delivered to the line. No logical combinations are made between the parts and each stock keeping unit is delivered in a full container quantity.

Kitting
A *kit* is a specific collection of components and subassemblies that together (i.e. in the same container) support one or more subassembly operations for a given product or "shop order" (Bozer and McGinnis, 1992) and *kitting* is the practice of putting together a kit of components and/or subassemblies – according to a future assembly schedule – before delivery to the shop floor. Kit assembly takes place in a separate supermarket area.

Because of the increasing number of parts moving around on the shop floor and the corresponding large number of transactions, considerable amounts of money are involved. Therefore, performing parts handling activities efficiently and assigning the most appropriate line feeding methods to the parts at hand is of major importance. In this viewpoint a cost model can be used to look for an optimal assignment solution.

In this paper a compact cost model is considered. The focus is not on representing all the features of the line feeding problem, but instead on illustrating how SysML can be used to model an optimization model graphically and the benefits of doing this. More specifically, the model considers the influence of the materials supply method, i.e. bulk feeding or kitting, on the operator efficiency at the assembly line, and the labor costs for the operator that has to do the additional material handling when kitting is preferred are included. Optimal decision making will then be guided by the objective of minimizing all costs.

Notation

COH	Cost of an operator hour (€/h)
Vel	Velocity of an operator (m/h)
$YQTY_i$	Usage (units/year) of part number i
tp_i^b	The time (h) to pick a unit of part number i from a bulk container
tp^k	The time(h) to pick a unit from a kit
Δ_i^k	The distance (m) for the operator in the supermarket to pick part number i from a bulk container to kit
f_i	Percentage of end products using part number i
m_i	Number of units of part number i assembled per end product (bill of material)
B^{kit}	The batch size for assembling kits, i.e. the number of kits an operator assembles at once
tk_i	The time (h) to pick a part number i in the supermarket from its bulk container in order to kit it

A binary decision variable assigns parts to the appropriate materials supply method:

x_i	$x_i = 1$: Bulk feeding
	$x_i = 0$: Kitting

Mathematically we can represent the objective function by the following:

$$\text{Min } C_{total} = \text{Min } \left(C_{pick} + C_{kit}\right)$$

$$C_{pick} = \sum_i \left[COH \cdot YQTY_i \cdot \left[x_i \cdot tp_i^b + (1 - x_i) \cdot tp^k\right]\right]$$

$$C_{kit} = \sum_i [COH \cdot (1 - x_i) \cdot YQTY_i \cdot tk_i]$$

With:

$$tk_i = \left(2 \cdot \frac{\Delta_i^k}{Vel}\right) \Big/ f_i \cdot m_i \cdot B^{kit}$$

C_{pick} represents the cost of labor at the assembly line. The picking time for the production operator at the line is dependent on the materials supply method. If parts are kitted, the operator productivity will be higher and the picking time shorter ($\forall i: tp_i^b \geq tp^k$).
C_{kit} represents the cost of labor in the kitting area. If parts are kitted, additional labor is needed to assemble the kits in advance. Productivity of picking in a kitting area is higher because kits can be assembled in batches and kitting areas can be set up to contribute to efficient picking.

MODELING OF STRUCTURE

In order to model the problem at hand we used two of the four important pillars of SysML (Figure 1), i.e. structure and parametrics. A link between the two is obtained through value binding. In this section we start by describing the structure. The next section concentrates on parametrics.

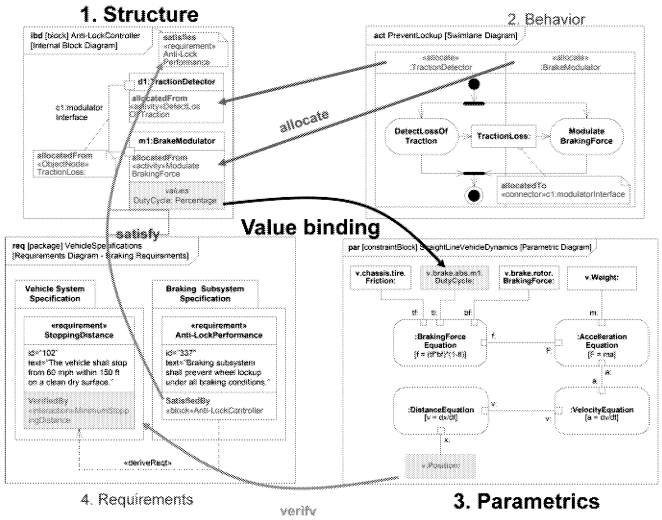


Figure 1: The Four Pillars of SysML
(Source: Friedenthal, www.omgsysml.org)

As mentioned in the previous section, the main objective is to calculate the total cost for supplying all parts to the line. This cost can be obtained using a sequence of equations. To allow for additional complexity to be easily added in the future, we use object-oriented modeling techniques. This allows a model to be built from simple, independent, and potentially reusable subsystems, and to be tied together only at the highest levels. Obviously, this object-oriented structure will contribute to an overall comprehensible model and a clear structure. SysML supports object-oriented

modeling by use of Block definition diagrams (BDD). A *Block Defintion Diagram (BDD)* describes the organization of the structure, the hierarchy of system, subsystems, and all the elements that make up the system.
Moreover, in SysML the design of complex systems is achieved in a top-down approach. We will illustrate this top-down approach for our example model starting with the composition of the overall total cost, and gradually concentrating more on details of the sub-costs.

The structure of the total cost is represented in the Block Definition Diagram 'Totalcost' (Figure 2). In the center three blocks are represented. A *block* is a very general modeling concept in SysML that is used to model a wide variety of entities that have structure such as systems, hardware, software, physical objects, and abstract entities (Friendenthal et al. 2008). The interconnections between the blocks are composite association paths that relate the whole to its parts. In this case abstract entities are represented, the total cost which is composed of a picking cost and a kitting cost. In the right upper corner a constraint is displayed. A *constraint block* is a special kind of block used to define equations (Friendenthal et al. 2008). The total cost equation, i.e. total cost being a sum of the picking cost and the kitting cost, is clarified in the constraint TotalcostEqn.

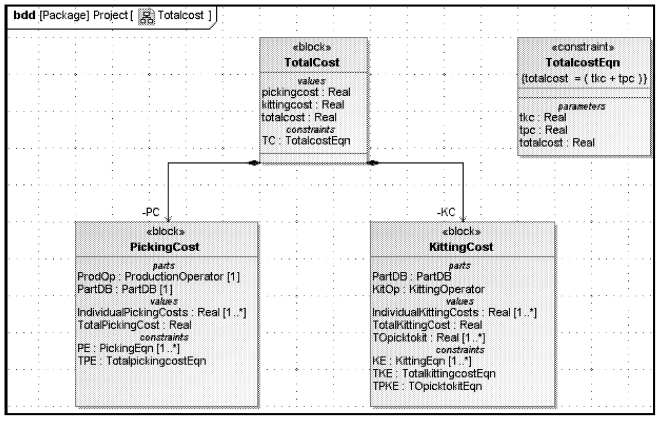


Figure 2: Block Definition Diagram – Total cost

The following Block Definition Diagrams deal with more detail of the sub-costs. PickingBDD (Figure 3) and Kitting BDD (Figure 4) structure respectively the picking cost and the kitting cost. The fact that picking cost and kitting cost are each shown in a separate Block Definition Diagram helps to maintain a clear overview.

The picking cost for the production operator at the line C_{pick} (Figure 3) is calculated from the part database and data about the production operator. The part database gives information for each of the parts that need to be supplied to the line and is structured as an array using value properties with aggregate data types. A part is characterized by its part number, its $YQTY_i$, its weight, its tp_i^b, its frequency of occurrence, its bill of material, its Δ_i^k, and its decision variable x_i. The production operator has an hourly labor cost, and a constant tp^k.

The calculation of the total picking cost is done in two steps. First, the constraint 'PickingEqn' calculates individual picking costs for each of the parts as an aggregate value property. The second constraint 'TotalpickingcostEqn' then calculates the total picking cost as a sum of the individual picking costs, by means of the aggregate sum function 'sum()'.

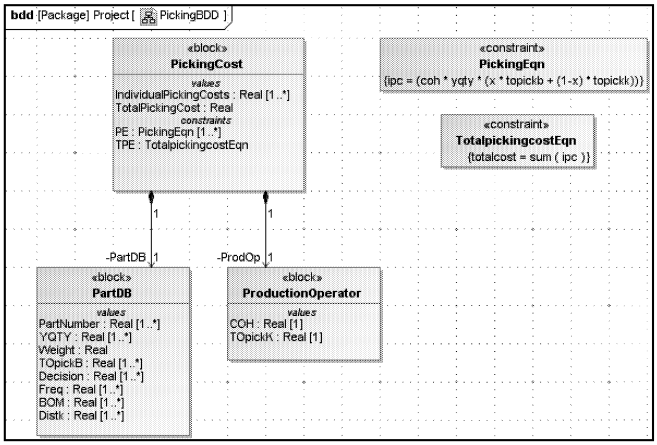


Figure 3: Block Definition Diagram – Picking cost

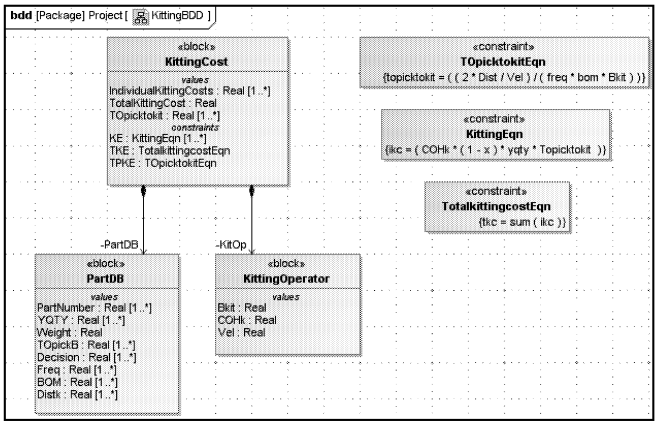


Figure 4: Block Definition Diagram – Kitting cost

The kitting cost C_{kit} (Figure 4) is modeled similarly. An extra constraint is needed for the representation of the operator time needed to pick a unit of part number i from the kitting area.

MODELING OF PARAMETRICS

With the use of Block Definition Diagrams, the structure of the model is described. Additionally Parametric Diagrams are used to relate the constraints and parameters. *Parametric diagrams* are used to create systems of equations that can constrain the properties of blocks (Friedenthal et al. 2008). For more information about Parametrics we refer to Peak et al. (2007).

For each of the constraints a Parametric Diagram is created. These Parametric Diagrams take care of the correct value binding of all parameters of the model. We will show all of the Parametric Diagrams in the same structure as the Block Definition Diagrams. The diagrams should be clear without much further explanation.

In Figure 5 at the top the two inputs for the total cost equation are linked to the constraint and below the output is displayed.

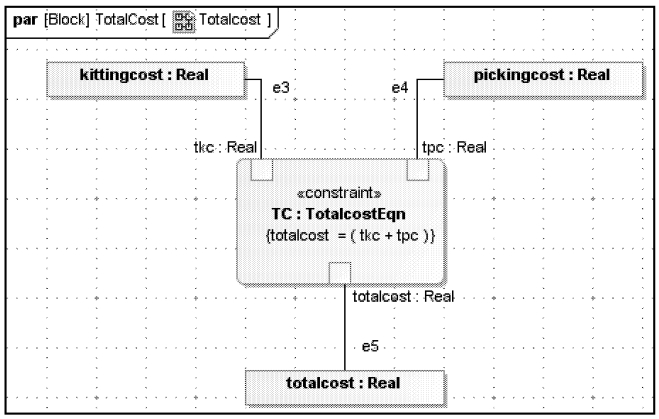
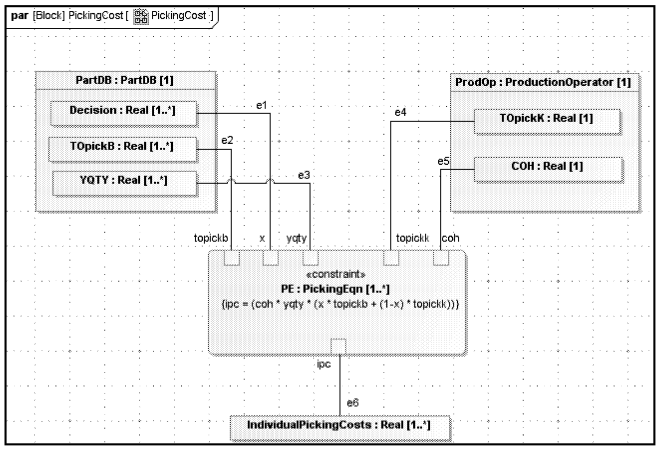


Figure 5: Parametric Diagram – Total cost

Figure 6 displays the inputs and output for individual picking costs per part in the first diagram and the transformation to a total picking cost in the second diagram.



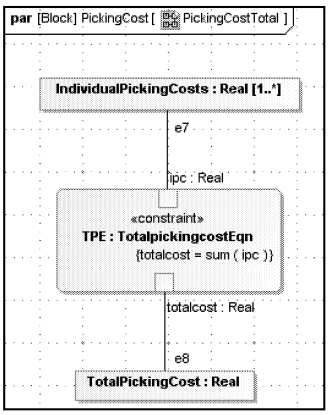
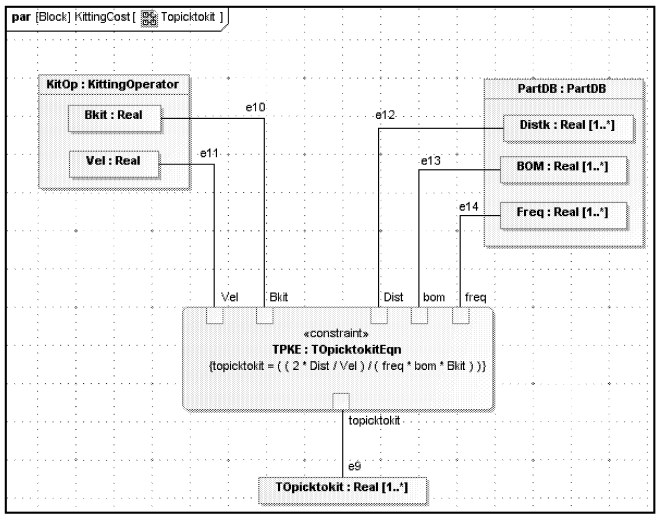
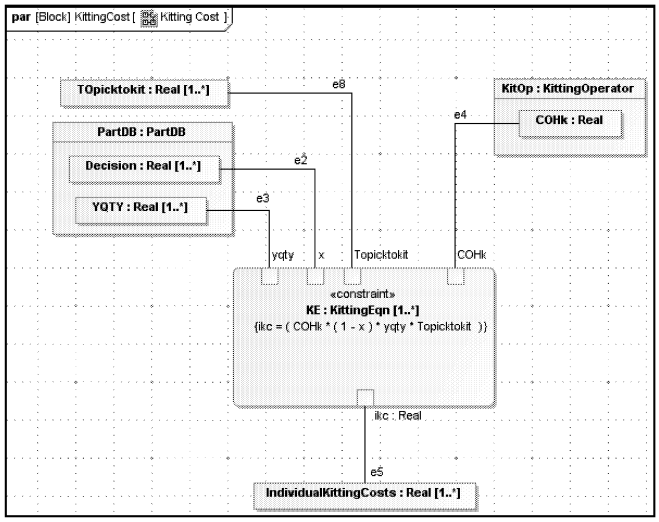


Figure 6: Parametric Diagrams – Picking cost

Figure 7 presents the same for the kitting cost. An extra first diagram is added for the representation of tk_i, an intermediate factor for the calculation of the individual kitting costs. The second diagram displays the inputs – including the previously defined tk_i – and output for individual kitting costs per part. The last diagram shows the transformation to a total kitting cost.





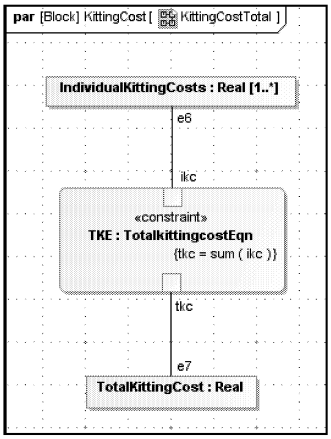


Figure 7: Parametric Diagram – Kitting cost

VALIDATION

To avoid errors, the model schema is at this point validated through the creation of a CXS_heading in MagicDraw. Bugs are revealed and removed in order to obtain a valid structure for the SysML model.

As a second tool the SysML parametrics "flattened graph" visualization tool, BuzzToys Panorama, is used to validate the model. The graphic representation (Figure 8) gives a clear overview and by walking through the tree structure on

the left it can be checked that the subsystems are linked correctly to the parameters and to each other.

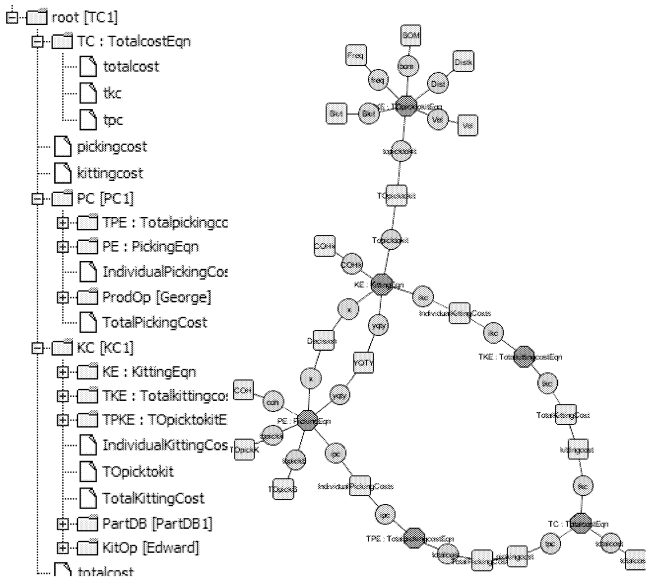


Figure 8: Panorama view – Total cost

SOLVING THE MODEL FOR A GIVEN INSTANCE

An instance is an example of the model with specific values assigned to the given parameters and which can be solved for the unknowns. In this section we will explain how the SysML model can be used to solve for total costs.

A new Block Definition Diagram is created to structure the instance. This Block Definition Diagram is populated with instance blocks and appropriate connections are made. To add values for the instance we made use of the 'read from' Excel functionality. A part database is entered in an Excel file which later is used to write the results. In MagicDraw, the slots to put the instance values need to be created beforehand and a causality type needs to be given. The picking cost, kitting cost and total cost variables are assigned target causalities. To avoid errors, the instance is validated through the creation of a CXI_heading. Bugs are removed from the model to obtain a valid structure.

The model is solved with ParaMagic. ParaMagic connects to the Mathematica solver which uses the equations built in the parametric diagrams to obtain the desired results. In Figure 9 the ParaMagic interface is shown. The input parameters are assigned a causality type 'given', and the output variables are assigned a causality type 'target'. The 'Solve' command allows to connect to Mathematica and the 'Update to SysML' command allows to pass the results on to the visual model afterwards.

Finally the solution is written to the original Excel file by the 'write to' Excel functionality.

CONCLUSIONS AND FURTHER RESEARCH

The purpose of the project is to model the in-plant logistics processes of parts to an automotive assembly line and create a tool for analyzing the costs of the materials supply. Structure and Parametric Diagrams in SysML were used for modeling and ParaMagic was used as a solver.

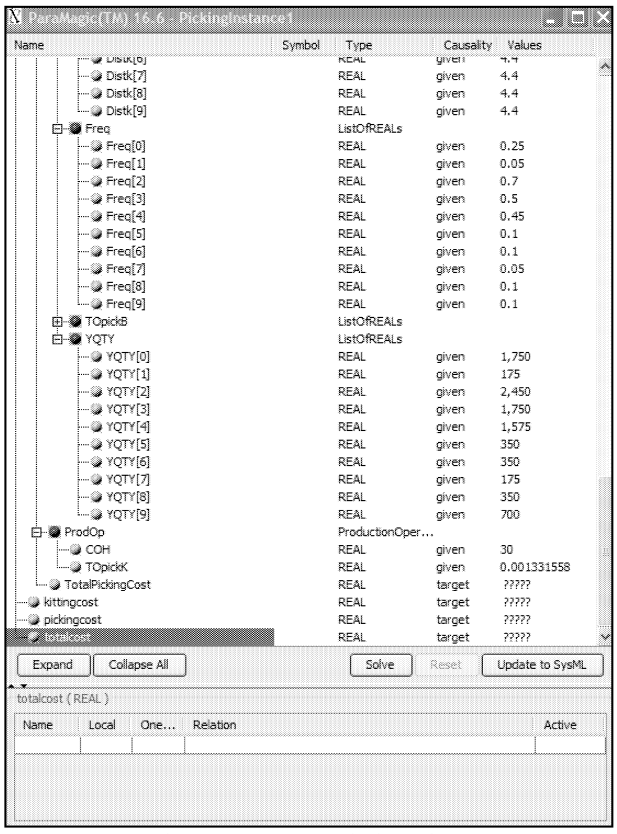


Figure 9: ParaMagic interface

Modeling in SysML has two main benefits. First, the model creates a common basis for understanding and a domain specific language is created. The structure of the model is clear and is graphically represented in a comprehensible way. Secondly, the model does not only describe the structure, which could equally be done by use of flow charts or other visual tools, but the model can be used for analysis purposes as well. This avoids duplication of efforts for building the descriptive model and then another computational model. For analysis purposes SysML supports a broad range of integration and interoperation with specific solvers, thus it enables tight integration between description and analysis. In this project ParaMagic is used to interoperate with Mathematica in order to solve the model.

The work on this project attempts to lay the groundwork for the implementation of a complete model for the problem of determining the optimal configuration of kitted and bulk fed parts to an assembly line. It is, in effect the starting point towards facilitating the use of SysML to completely specify and quantify the problem.

REFERENCES

Bozer, Y.A. and L.F. McGinnis. 1992. "Kitting versus line stocking." *International Journal of Production Economics*, 28, 1-19.

Friedenthal, S.; A. Moore; and R. Steiner. 2008. *A Practical Guide to SysML. The Systems Modeling Language.* Elsevier Inc.

Peak, R.S. R.M. Burkhart, S.A. Friedenthal, M.W. Wilson, M. Bajaj, and I. Kim. 2007. "Simulation-Based Design Using SysML – Part 1: A Parametrics Primer." *INCOSE Intl. Symposium, San Diego.*

SUPPLY CHAIN SIMULATION

A PRACTICAL APPROACH TO PERFORMANCE IMPROVEMENT AND OPTIMISATION IN SUPPLY CHAIN MANAGEMENT

Walid Smew
Paul Young
John Geraghty
Enterprise Process Research Centre
School of Mechanical & Manufacturing Engineering
Dublin City University
Dublin 9 Ireland
E-mail: john.geraghty@dcu.ie

KEYWORDS
Supply chain management; Kanban; Modelling; Discrete event simulation; Design of experiment; Optimisation

ABSTRACT

This paper presents a simulation study of a serial supply chain employing the Kanban production control strategy and examines the trade-off between the conflicting objectives of maximising customer service level and minimising average Work-In-Process (WIP). A simulation model was developed in ExtendSim V6 for a four-node centralised serial Supply Chain (SC) adopting Kanban to process a single product. The model was used to explore the impact of essential input factors e.g. the number of Kanban cards at each node, the maximum production capacity of a node, and the standard deviation of demand on customer service level and the average WIP using Response Surface Methodology (RSM). Design-Expert V7 was used to construct the experimental design matrices, analyse the experimental data, and conduct a metamodel-based multi-objective SC optimisation. This paper provides a discussion of the experimental results and implications for supply chain managers and offers insights on future research opportunities in this field.

INTRODUCTION

Supply chain management focuses on the integration of suppliers, manufacturers, warehouses, and stores. The challenge in Supply Chain (SC) integration is to co-ordinate activities across the entire SC so that the enterprise can improve performance, reduce system inventory levels and potential inventory cost, increase customer service level, better utilise resources, and effectively respond to changes in the market. Designing and implementing a globally optimal SC is difficult because of its dynamics and the conflicting objectives employed by the different facilities and partners in the SC. However, sharing information and implementing an effective SC control strategy are the keys to successfully achieve these goals (Simchi-Levi et al. 2003).

Supply chains are often categorized as push-based, pull-based or push-pull supply chains. In push-based supply chains, such as material requirements planning (MRP) systems, production and distribution decisions are based on long term demand forecast and products are pushed through the channel, from the production side upstream to retailers downstream and hence control throughput and observe WIP from time to time. This characteristic may enable the system to reduce delivery lead time since many semi-finished or finished products are available but also, it will lead to the inability to meet changing demand patterns and to the bullwhip effect (Simchi-Levi et al. 2003; Takahashi and Nakamura 2004; Ghrayeb et al. 2009).

In pull-based supply chains, such as Kanban systems, production and distribution decisions are based on true customer demand rather than forecasted demand and SC members do not hold any excess inventory and only respond to specific orders (Simchi-Levi et al. 2003). Such systems usually have significant reductions in inventory levels and costs, and better response to market changes. However, this system may not work well in multi-product environments and in environments with large demand variations resulting in significant backorders, longer delivery lead times, and higher late penalty costs (Krishnamurthy et al. 2004). In push-pull supply chains, some stages of the SC, typically the initial stages are operated in a push-based manner while the remaining stages are operated in pull-based manner. This hybrid control often compromises the conflicting performance characteristics of the push and the pull environments so that a better system performance can be anticipated (Hodgson and Wang 1991). For advantages and disadvantages of the push and pull systems and other details see also the literature (Krajewski et al. 1987; Spearman et al. 1990; Deleersnyder et al. 1992; Spearman and Zazanis 1992; Hopp and Roof 1998; Sendil Kumar and Panneerselvam 2007; Lavoie et al. 2010). Of all the existing production control strategies, only the Kanban Control Strategy (KCS) will be considered in this paper.

LITERATURE REVIEW

KCS is the best known pull production control strategy that was first developed by a Japanese automobile manufacturer, TOYOTA, during the 1970s and is often considered to be closely associated with the philosophy of Just-In-Time (JIT) manufacturing (Kimura and Terada 1981; Monden 1981; Schonberger 1982). KCS has been the subject of numerous studies (Huang and Kusiak 1996; Akturk and Erhun 1999). A recent literature review on the various Kanban systems and alternatives proposed by several researchers, classified all the previous studies and presented suggestions for future research (Lage Junior and Godinho Filho 2010).

In KCS (see Figure 1) Kanbans are used to control the release of parts to each production stage. The Kanban is attached to the part when production starts and travels

downstream with the part to the stage's immediate successor. When the immediate successor begins production on the part, the Kanban is detached and sent back upstream to the production stage in order to authorise the production of a replacement part. Only a demand event can remove a part from the finished-items inventory points. The advantage of this mechanism is that the number of parts in every stage is limited by the number of Kanbans of that stage. Its disadvantage is that the system may not respond quickly enough to changes in the demand especially in upstream stages of longer lines. Further, KCS requires inventories of semi-finished products to be maintained at each production stage. Therefore, in multi-product environments the amount of semi-finished inventory maintained in the line could be prohibitively large (Liberopoulos and Dallery 2000).

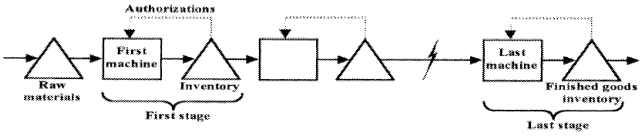


Figure 1: Kanban Control Strategy (Gaury et al. 2001)

Generic Kanban System (GKS) is a KCS applicable to dynamic, multi-product, and non-repetitive manufacturing environments (Chang and Yih 1994). GKS operates by providing a fixed number of Kanbans at each workstation that can be acquired by any product. A product/job can only enter the system if it acquires Kanbans from each of the workstations in the system. In comparison with CONWIP control strategy, GKS was favourable and shown to be more flexible. GKS can be shown to reduce to segmented CONWIP with one workstation in each segment (Gstettner and Kuhn 1996). This, however, is not completely true, as there is an overall cap on WIP caused by the requirement for jobs to acquire Kanbans from all the workstations before the job can enter the system. Generalized Kanban (Buzacott 1989) and the Extended Kanban (Dallery and Liberopoulos 2000) depend on two parameters per stage, the number of Kanbans and the base stock of parts in inventory. The Generalised and Extended Kanban control strategies are both based on the integration of KCS and the Basestock Control Strategy (BSCS). Combining the merits of BSCS and KCS led to many benefits as the Basestock mechanism reacts faster to demand and the Kanban mechanism achieves better coordination and limits WIP.

The problem of determining optimally the number of raw material orders, Kanbans, finished goods shipments to the buyers, and the batch size for each shipment for a multi-stage KCS with linear demand has been addressed in the literature (Sarker and Balan 1999). A cost function based on the costs incurred due to these parameters was developed. The optimal number of raw material orders that minimizes the total cost was obtained and used to find the optimal number of Kanbans, finished goods shipments, and the batch sizes for shipments. In KCS the number of cards in use is fixed and with the fluctuation of demand this may lead to either large WIP or backorders. Adaptive KCS (number of cards is dynamically adjusted based on current inventory and backorders levels) was shown to outperform KCS, even under stable conditions, and is easy to implement (Tardif and Maaseidvaag 2001).

KCS has been compared to minimal blocking KCS, BSCS, CONWIP, and hybrid Kanban-CONWIP in tandem production lines (Bonvik et al. 1997) and the Hybrid Kanban-CONWIP strategy was shown to decrease inventories by 10% to 20% over KCS while maintaining the same service levels. Another comparison study of pull control mechanisms for unreliable tandem transfer lines producing a single product observed that the hybrid mechanism always outperforms CONWIP and Kanban when storage space and inventory costs are considered explicitly (Lavoie et al. 2010). However, hybrid was equivalent to CONWIP and both outperform KCS when storage space costs are not considered explicitly but aggregated with inventory costs in terms of holding costs.

MODELING A KANBAN SUPPLY CHAIN

The centralised KCS-SC in this paper is defined as a production–distribution system, in which each firm is considered as a "work centre" being a part of a "global line" of supply and also in which a virtual centre of control governs the SC and manages the information and parts flow and the inventories along the chain. The proposed SC consists of four nodes in series (see Figure 2).

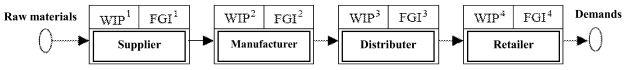


Figure 2: Supply Chain Consisting of Four Nodes in Series

Notations and Definitions

i. Material flow variables

P_t^i: Pipeline (WIP) in node *i* in period *t*, where *i=1, 2,..., n*
Y_t^i: Finished goods inventory (FGI) of node *i* in period *t*
S_t^i: Shipments from node *i* to node *i+1* in period *t*
O_t^i: Output from the pipeline of node *i* in period *t*
I_t^i: Input to the pipeline of node *i* in period *t*

ii. Information flow variables

D_t: Incoming orders to the SC at final node in period *t*
OP_t: Orders placed by the SC to node *1* in period *t*
B_t: Backlog of incoming orders in the SC in period *t*
TY_t^i: Available total FGI in node *i*, in period *t*

iii. Model parameters

L^i: Cycle (lead) time for a unit in the pipeline to arrive to the FGI in node *i*
MLP^i: Maximum number of items to be processed in node *i*
K^i: Total number of Kanban cards of node *i*

Model Formulation

i. Shipments

Incoming orders to the final node (D_t), will be immediately shipped to customers but inventory constraints may affect these shipments. Therefore, S_t^n is given by:

$$S_t^n = \min[TY_t^n, D_t] \quad (1)$$

Otherwise, S_t^i depends on TY_t^i, the maximum number of parts to be processed in node *i+1* during its lead

time (MLP^{i+1}/L^{i+1}), and the available number of Kanban cards in node $i+1$:

$$S_t^i = \begin{cases} \min\left[\begin{matrix} TY_t^i, \left(\frac{MLP^{i+1}}{L^{i+1}}\right), \\ \left(K^{i+1} - [P_{t-1}^{i+1} + Y_{t-1}^{i+1}]\right) \end{matrix}\right] & i = 1, \dots, n-2 \\ \min\left[\begin{matrix} TY_t^{n-1}, \left(\frac{MLP^n}{L^n}\right), \\ \left(K^n - [P_{t-1}^n + Y_{t-1}^n - S_t^n]\right) \end{matrix}\right] & i = n-1 \end{cases} \quad (2)$$

ii. Backlogs

$$B_t = \max[(B_{t-1} + D_t - O_t^n) * (BL), 0] \quad (3)$$

Where: $BL = \begin{cases} 0 & if\ no\ backloging \\ 1 & otherwise \end{cases}$

iii. Materials flow, WIP, and inventory

Assuming that the initial conditions are known, P_t^i and Y_t^i are given by:

$$P_t^i = P_{t-1}^i + I_t^i - O_t^i \quad (4)$$

$$Y_t^i = Y_{t-1}^i + O_t^i - S_t^i \quad (5)$$

TY_t^i can be calculated as follows:

$$TY_t^i = \begin{cases} Y_{t-1}^i + O_t^i & i = 1\ to\ n-1 \\ \max[(Y_{t-1}^n + O_t^n - B_{t-1}), 0] & i = n \end{cases} \quad (6)$$

Where:

$$O_t^i = I_{t-L^i}^i \quad (7)$$

$$I_t^i = \begin{cases} OP_t & for\ i = 1 \\ S_t^{i-1} & for\ i \neq 1 \end{cases} \quad (8)$$

iv. Orders

OP_t depends on the maximum number of parts to be processed in the first node during its lead time (MLP^1/L^1), and the available number of Kanban cards in node 1 and is given by:

$$OP_t = \max\left[\min\left(\frac{MLP^1}{L^1}, [K^1 - (P_{t-1}^1 + Y_{t-1}^1)]\right), 0\right] \quad (9)$$

Performance Measures

The performance measures for the SC that will be considered in this work will be the Average Inventory in the system and the Average Service Level (or fill rate) achieved by the system after *n* periods of time. Inventory level is a key decision associated with SC performance since maintaining the right level of inventory at the right place and at the right time helps reducing costs and improves service levels.

SIMULATION MODEL DEVELOPMENT

Simulation modelling provides a virtual environment that enables "what-if" analyses to test several strategies and scenarios and permits the comparison of various operational alternatives leading to better future decision. The modelling phase starts after the supply chain is analysed and the key components and performance measures are identified (Jain et al. 2001). The conceptual, mathematical model described earlier has been implemented in a periodic review simulation model of a centralised serial SC adopting KCS using ExtendSim. The Activity block in the ExtendSim library was modified, using the provided programming language Mod-L, to trigger the calculation of the mathematical model each time an entity entered the block. The 'entity' is a control command to instruct the model to perform the calculations for the next period. This single block effectively models the control structure of the entire SC. Information, such as the number of nodes in the SC, capacities of nodes, number of Kanbans assigned to nodes etc., is stored in global arrays which are accessed by the Activity block. This modelling approach provides a flexible, parametric and time efficient model.

META-MODEL DEVELOPMENT

In most RSM problems, the true form of the functional relationship between the response and the independent variables is unknown, but can be reasonably approximated by a second-order polynomial model over a relatively small region of the independent variables space as follows:

$$y = \beta_0 + \sum_{i=1}^{k} \beta_i\ x_i + \sum_{i=1}^{k} \beta_{ii}\ x_{ii}^2 + \sum_{\substack{i,j=1 \\ i \neq j}} \beta_{ij}\ x_i x_j + \varepsilon \quad (10)$$

Where β_0 is the intercept, $\beta_i\, x_i$ are the linear terms, $\beta_{ii}\, x_{ii}^2$ are the quadratic terms, and $\beta_{ij}\, x_i x_j$ are the interaction terms.

Fitting and analysing this second-order polynomial model with data from a simulation model can be greatly facilitated by careful consideration of the RSM experimental design. The model of Equation (10) contains $1 + 2k + k(k-1)/2$ parameters (where k is the number of variables), so the selected experimental design must have at least this number of distinct design points and at least three levels for each selected independent variable. Box and Wilson Central composite designs (CCD) and Box and Behnken three-level designs (BBD) are the most popular class of second-order designs (Myers and Montgomery 2002; Ferreira et al. 2007). These designs are efficient while not involving large number of design points. BBD is a class of rotatable or nearly rotatable second-order designs based on three-level fractional factorial design for $k \geq 3$ and consist of a total number of experiments that can be defined by $2k(k-1) + n_c$ where the multiple center runs, n_c, are required in order to have a sufficient number of degrees of freedom to estimate the experimental error and to manage the distribution and stability of the scaled prediction variance at different design regions (the property of rotatability).

In order to limit the working range of the input factors, different meta-models were developed at different levels of demand standard deviation (Smew et al. 2010). Therefore, five input factors were identified to be varied for each meta-model, (as in Table 1). The working range of the Kanban cards of node 4 depends on the selected level of demand standard deviation, the service level to be achieved, and a response maximum to minimum ratio of less than 3. The individual targeted levels of demand standard deviation considered were 1, 4.5, and 8 items to investigate KCS-SC under low, medium, and high demand variability situations.

Table 1: KCS-SC Design Summary at Demand SD = 8

Factor	*Name*	*Units*	*Low*	*High*	
A	Node Capacity	Items	8	24	
B	Node1Kanbans	Cards	12	16	
C	Node2Kanbans	Cards	12	16	
D	Node3Kanbans	Cards	12	20	
E	Node4Kanbans	Cards	28	112	
Response	*Name*	*Units*	*Min*	*Max*	*Ratio*
Y_1	Service Level	%	82.50	98.11	1.19
Y_2	Average WIP	Items	54.53	133.55	2.45

A BBD was selected where each variable has three levels; low, middle, and high coded to the usual (-1, 0, 1) notation. The total number of experimental runs of all scenarios according to this design is 46; $[2k(k-1)=40]$ augmented with 6 replicated centre points n_c coded as (0, 0, 0). Table 1 provides a summary of the experimental design for SD =8. For all conducted experiments: (i) simulation run-length was 11000 periods which includes a warm-up time of 1000 periods to remove the bias of the initial conditions (ii) the average output of 50 replications was recorded for each response variable, (iii) the lead time was fixed, the same for all nodes, and had a value of 2 periods, (iv) all nodes had the same node capacity value which was varied according the specified working range, and (v) customer demand was modelled by a Lognormal Distribution with a mean of 4 items per period.

To estimate the different meta-models terms, the polynomial model (Equation 10) was fitted to the data using the method of least squares and the step-wise regression procedure which excludes all the non-significant terms in the meta-models at a level of significance of 5% ($\alpha = 0.05$). Although the simulation model is stochastic in nature, only a small variation between replications was observed in the simulation outputs, which makes the results appear to be almost deterministic. Therefore, the conventional *lack-of-fit* test will not be considered to validate the fitted meta-models. Despite this, much of the standard analysis of variance tools remain relevant. ANOVAs (e.g. Table 2) revealed that all models are quadratic, significant, and fit the data adequately at a level of significance of 5%.

Table 2: ANOVA for SL Reduced Model at SD=8

Source	*Sum of Squares*	*df*	*Mean Square*	*F Value*	*p-value Prob > F*	
Model	1036.43	13	79.73	400.27	< 0.0001	***significant***
A	16.77	1	16.77	84.18	< 0.0001	
B	2.37	1	2.37	11.92	0.0014	
C	3.48	1	3.48	17.46	0.0002	
D	10.17	1	10.17	51.08	< 0.0001	
E	750.26	1	750.26	3766.75	< 0.0001	
AD	2.37	1	2.37	11.92	0.0014	
AE	1.89	1	1.89	9.48	0.0038	
DE	0.89	1	0.89	4.48	0.0409	
A^2	7.89	1	7.89	39.59	< 0.0001	
B^2	2.03	1	2.03	10.20	0.0028	
C^2	2.16	1	2.16	10.84	0.0022	
D^2	3.48	1	3.48	17.45	0.0002	
E^2	127.65	1	127.65	640.88	< 0.0001	
Residual	7.57	38	0.20			
Total	1043.99	51				
R^2	0.9928		***Adj R^2***	0.9903		
Pred R^2	0.9853		***PRESS***	15.3738		

R^2, adjusted R^2, and Predicted R^2 show that a high percentage of the variability in the original and new observations are explained by the fitted models and that they are in a logical agreement, indicating that all models are adequate. The differences between PRESS and the ordinary residual sum of squares in all models are reasonable indicating that the fitted models are capable of making good predictions for new scenarios. In addition, examination of the service level and average WIP trade-off curves of the developed meta-models and the simulation model (Figure 3) further evidences the accuracy of the meta-models.

OPTIMISATION

Multiple response optimisation consists of finding a set of operating conditions that optimises all responses or that keeps them in a desired range simultaneously. Rather than optimising the KCS-SC performance measures to find a single set of optimum operating conditions, service level and average WIP trade-off curves that could be used by managers to assess the performance of the SC were constructed. The Desirability Approach (Derringer and Suich 1980) was applied to the metamodels and the optimum input factors combinations that minimize the average WIP for targeted service levels were obtained.

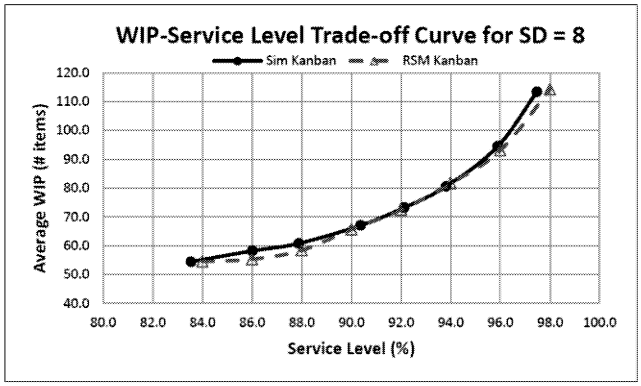


Figure 3: KCS-SC Trade-off Curve for Demand SD=8

To build the trade-off curves for the KCS-SC, the service level (at several possible targets) and the average WIP were optimised simultaneously for each selected demand SD. The two-sided desirability function for each targeted service level was defined as in the following example:

$$d_1 = \begin{cases} 0 & service\ level < 83.5 \\ (0,1) & 83.5 \le service\ level \le 84 \\ (0,1) & 84 \le service\ level \le 84.5 \\ 0 & service\ level > 84.5 \end{cases} \quad (11)$$

For the average WIP meta-models the goal is always to minimise them and their one-sided desirability function was defined as in the following example:

$$d_2 = \begin{cases} 1 & average\ WIP < 54.53 \\ (0,1) & 54.53 \le average\ WIP \le 133.55 \\ 0 & average\ WIP > 133.55 \end{cases} \quad (12)$$

The optimal solutions from the desirability approach along with actual simulation outputs for SD=8 are illustrated in the trade-off curve shown in Figure 3. Similar analysis has been conducted for the other values of SD.

CONCLUSIONS AND FUTURE RESEARCH

The research presented here is preliminary findings from work aimed at firstly understanding the influence of customer demand variability in a KCS serial supply-chain and secondly at the potential for utilising meta-models to

optimise such systems to address the trade-off between inventory and service level. The influence of demand variability on the performance of the supply-chain could be mitigated by optimising the number of Kanban cards in the system by applying the Desirability Approach to the meta-models. The combined use of Simulation, RSM and Optimisation for such problems is potentially a valuable tool for decision makers. In the near future it is planned to investigate other control strategies and compare their performances to KCS. Other structures, such as a Tiered-SC, and other sources of variability will be analysed such as lead-times and capacities at nodes in the SC.

ACKNOWLEDGEMENTS

The Libyan Government is gratefully acknowledged for the financial support provided.

REFERENCES

Akturk, M.S. and F. Erhun. 1999. "An Overview of Design and Operational Issues of Kanban Systems." *International Journal of Production Research,* Vol. 37, No. 17, 3859-3881.

Bonvik, A.M.; C.E. Couch; and S.B. Gershwin. 1997. "A Comparison of Production-Line Control Mechanisms." *International Journal of Production Research,* Vol. 35, No. 3, 789-804.

Buzacott, J.A. 1989. "Queueing Models of Kanban and MRP Controlled Production Systems." *Engineering Costs and Production Economics,* Vol. 17, No. 1-4, 3-20.

Chang, T.M. and Y. Yih. 1994. "Generic Kanban Systems for Dynamic Environments." *International Journal of Production Research,* Vol. 32, No. 4, 889-902.

Dallery, Y. and G. Liberopoulos. 2000. "Extended Kanban Control System: Combining Kanban and Base Stock." *IIE Transactions,* Vol. 32, No. 4, 369-386.

Deleersnyder, J.L.; T.J. Hodgson; R.E. King; P.J. O'Grady; and A. Savva. 1992. "Integrating Kanban Type Pull Systems and MRP Type Push Systems: Insights from a Markovian Model." *IIE transactions,* Vol. 24, No. 3, 43-56.

Derringer, G. and R. Suich. 1980. "Simultaneous Optimization of several Response Variables." *Journal of quality technology,* Vol. 12, No. 4, 214-219.

Ferreira, S.L.C.; R.E. Bruns; H.S. Ferreira; G.D. Matos; J.M. David; G.C. Brandão; E.G.P. da Silva; L.A. Portugal; P.S. dos Reis; A.S. Souza; and W.N.L. dos Santos. 2007. "Box-Behnken Design: An Alternative for the Optimization of Analytical Methods." *Analytica Chimica Acta,* Vol. 597, No. 2, 179-186.

Gaury, E.G.A.; J.P.C. Kleijnen; and H. Pierreval. 2001. "A Methodology to Customize Pull Control Systems." *The Journal of the Operational Research Society,* Vol. 52, No. 7, 789-799.

Ghrayeb, O.; N. Phojanamongkolkij; and B.A. Tan. 2009. "A Hybrid push/pull System in Assemble-to-Order Manufacturing Environment." *Journal of Intelligent Manufacturing,* Vol. 20, No. 4, 379-387.

Gstettner, S. and H. Kuhn. 1996. "Analysis of Production Control Systems Kanban and CONWIP." *International Journal of Production Research,* Vol. 34, No. 11, 3253-3273.

Hodgson, T.J. and D. Wang. 1991. "Optimal Hybrid push/pull Control Strategies for Parallel Multistage System: Part I." *International Journal of Production Research,* Vol. 29, No. 6, 1279-1287.

Hopp, W.J. and M.L. Roof. 1998. "Setting WIP Levels with Statistical Throughput Control (STC) in CONWIP Production Lines." *International Journal of Production Research,* Vol. 36, No. 4, 867-882.

Huang, C.C. and A. Kusiak. 1996. "Overview of Kanban Systems." *International Journal of Computer Integrated Manufacturing,* Vol. 9, No. 3, 169-189.

Jain, S.; R.W. Workman; L.M. Collins; E.C. Ervin; and R. Accenture. 2001. "Development of a high-level supply chain simulation model". In *Proceedings of the 2001 Winter Simulation Conference* (Arlington, VA, Dec 9-12). IEEE 1129-1137.

Kimura, O. and H. Terada. 1981. "Design and Analysis of Pull System, a Method of Multi-Stage Production Control." *International Journal of Production Research,* Vol. 19, No. 3, 241-253.

Krajewski, L.J.; B.E. King; L.P. Ritzman; and D.S. Wong. 1987. "Kanban, MRP, and Shaping the Manufacturing Environment." *Management Science,* Vol. 33, No. 1, 39-57.

Krishnamurthy, A.; R. Suri; and M. Vernon. 2004. "Re-Examining the Performance of MRP and Kanban Material Control Strategies for Multi-Product Flexible Manufacturing Systems." *International Journal of Flexible Manufacturing Systems,* Vol. 16, No. 2, 123-150.

Lage Junior, M. and M. Godinho Filho. 2010. "Variations of the Kanban System: Literature Review and Classification." *International Journal of Production Economics,* Vol. 125, No. 1, 13-21.

Lavoie, P.; A. Gharbi; and J.-. Kenné. 2010. "A Comparative Study of Pull Control Mechanisms for Unreliable Homogenous Transfer Lines." *International Journal of Production Economics,* Vol. 124, No. 1, 241-251.

Liberopoulos, G. and Y. Dallery. 2000. "A Unified Framework for Pull Control Mechanisms in multi-stage Manufacturing Systems." *Annals of Operations Research,* Vol. 93, No. 1, 325-355.

Monden, Y. 1981. "How Toyota Shortened Supply Lot Production Time, Waiting Time and Conveyance Time." *Industrial Engineering,* Vol. 13, No. 9, 22-30.

Myers, R.H. and D.C. Montgomery. 2002. *Response Surface Methodology: Process and Product Optimization using Designed Experiments.* John Wiley & Sons Inc, New York.

Sarker, B.R. and C.V. Balan. 1999. "Operations Planning for a Multi-Stage Kanban System." *European Journal of Operational Research,* Vol. 112, No. 2, 284-303.

Schonberger, R. 1982. *Japanese Manufacturing Techniques: Nine Hidden Lessons in Simplicity.* The Free Press, New York.

Sendil Kumar, C. and R. Panneerselvam. 2007. "Literature Review of JIT-KANBAN System." *The International Journal of Advanced Manufacturing Technology,* Vol. 32, No. 3, 393-408.

Simchi-Levi, D.; P. Kaminsky and E. Simchi-Levi. 2003. *Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies.* McGraw-Hill, New York.

Smew, W.; P. Young; and J. Geraghty. 2010. "Exploring the Deployment of CONWIP in Serial Supply Chains Using Meta-Modeling". In *Proceedings of the 27th International Manufacturing Conference* (Galway, Ireland Sep. 1-3). GMIT, 13-24.

Spearman, M.L.; D.L. Woodruff; and W.J. Hopp. 1990. "CONWIP: A Pull Alternative to Kanban." *International Journal of Production Research,* Vol. 28, No. 5, 879-894.

Spearman, M.L. and M.A. Zazanis. 1992. "Push and Pull Production Systems: Issues and Comparisons." *Operations research,* Vol. 40, No. 3, 521-532.

Takahashi, K. and N. Nakamura. 2004. "Push, Pull, Or Hybrid Control in Supply Chain Management." *International Journal of Computer Integrated Manufacturing,* Vol. 17, No. 2, 126-140.

Tardif, V. and L. Maaseidvaag. 2001. "An Adaptive Approach to Controlling Kanban Systems." *European Journal of Operational Research,* Vol. 132, No. 2, 411-424.

Vendor Managed Inventory in the inbound Supply Chain in the Soft-Drink Industry

Eric R.W. Haardt [1)], Jaap A. Ottjes [2)], Bas J.H. van Delft [1)] and Gabriel Lodewijks [2)]

[1)] Vrumona BV
P.O. Box 1
3980 CA Bunnik
The Netherlands
e-mail: bas.vandelft@vrumona.nl

[2)] Delft University of Technology
Faculty of Mechanical, Maritime and Materials Engineering
Mekelweg 2, 2628 CD, Delft
e-mail: j.a.ottjes@tudelft.nl

KEYWORDS

Discrete Simulation; Vendor Managed Inventory; Soft-drink Manufacturer; Inbound Supply Chain

ABSTRACT

The ever growing competition in the soft-drink market led to manufacturers introducing the concept of Vendor Managed Inventory (VMI) with their suppliers. The goal of this concept is to reduce costs by letting the supplier manage the replenishment process of the customer. By sharing demand forecast and stock level information, the customer enables the supplier to plan its productions as effectively and efficiently as possible. The supplier is able to further improve the efficiency of production when he is able to hold a certain amount of stock for the customer, which in turn guarantees purchase. The maximum quantity, that the customer conforms itself to, is dictated by the maximum supply chain stock. For VMI it is important to analyse the required value for this constraint, in order to provide insight in the agreement that should be made between supplier and customer. The analysis is performed using a discrete simulation model of the VMI process of the inbound supply chain. Several scenarios with varying input parameters, such as the demand forecast accuracy, are tested.

INTRODUCTION

The process of VMI in the inbound supply chain shifts the control over the replenishment process from the customer to the supplier. This shift can be deduced from Figure 1 and Figure 2.

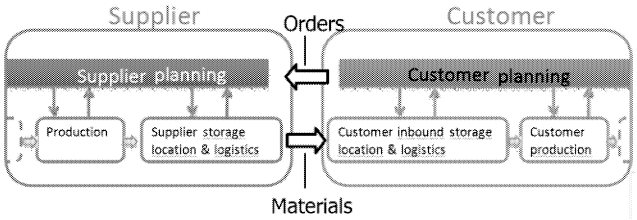


Figure 1: Traditional supply chain structure

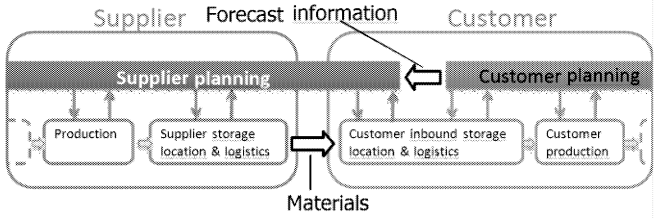


Figure 2: VMI supply chain structure

The figures show that under VMI, the supplier manages the material replenishment of the customer's storage location. To calculate the delivery size and timing, he uses forecast and inventory level information. He is then able to anticipate the requirements of the customer and optimise his own production planning accordingly. This process will therefore reduce production costs at the supplier, reduce planning control load at the customer (Kumar and Muthu, 2003), (Sari, 2008), and should lead to lower inventory levels at its storage location (Carey et al. 2005), (Mroz, 2001) at a higher supplier service level. The last two benefits emerge from the fact that with help of the forecast information, the supply chain becomes more transparent for the supplier.

Simulation of VMI has been used on multiple occasions to analyse the performance of the process and the possible benefits, (Southard et al. 2005). The focus of these simulations has been primarily on a vendor – retailer supply chain, investigating cost savings by reducing stock keeping costs and increasing

service level. In most cases it was found that the downstream member benefits the most, especially when VMI was imposed on the supplier. In the latter case often stringent criteria for maximum and minimum supply chain stock restricted the optimisation of the latter's process.
This research focuses on analysing the performance of VMI by setting wider targets for the VMI constraints. The important discrepancy with other VMI agreements is that in this case the customer will guarantee purchase for products up to a predetermined maximum supply chain stock. This implies that the supplier can maximise its production run size up to the optimal balance between unit production costs and unit stock keeping costs. Since the risk of not selling excess stock is taken away by the customer, the suppliers' incentive to carry out this optimisation is increased.
The aim of this paper is therefore to investigate the performance of VMI under the conditions stated above and to analyse the minimal values for the supply chain constraints to adhere to the maximum possible service level.

VENDOR MANAGED INVENTORY

Under VMI, the planning process of replenishments shifts from the customer to the supplier. Effectively this implies that the planning of productions and deliveries are now integrated at the supplier. The structure of the replenishment process is depicted in Figure 3.

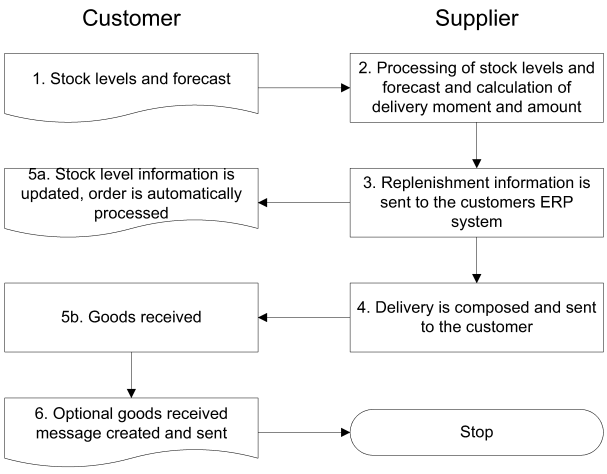


Figure 3: VMI replenishment process

Studying this VMI process shows that once the supplier has received or retrieved the stock level and forecast information of the customer, the process continues until the required materials are received at the customer's storage location. This implies that no intermediate checks or confirmations by the customer are necessary, although optionally these messages can be sent.
The initialisation and execution interval of the VMI process is dependent on the product and supply chain characteristics. Typically it is on a daily or weekly basis. In the latter case the interval between forecast updates is the leading factor. This case is the subject of the present research. This choice is based on two factors: First of all the soft-drink company that initiated this investigation works with weekly updates of the forecast; secondly replenishment planning is carried out on a weekly basis, except for a few materials of which the investigation lies outside the scope of this research.
The VMI process makes use of demand forecast and stock level information. All calculations are based on this data and it is therefore imperative that this information is correct and clear.

MODELLING

The discrete simulation model was developed in the TOMAS for Delphi package (Veeke and Ottjes, 2000). This package adds process-interactive based simulation properties, which facilitates the analysis of VMI. The model accepts both historical as well as stochastic input data, which is why solution by simulation was chosen. The historical data is based on actual demand, demand forecast and material information of the soft-drink manufacturer. This provided interesting insight in how the replenishment process would have performed if VMI had been used. The stochastic data was used to perform analysis of outliers, analyse the robustness of the VMI process in the model and determine the values for the VMI constraints.
Next to applying VMI, the simulation model is also capable of running the replenishment process in the traditional way, of which the structure is presented in Figure 1. The VMI and traditional model can then be compared to assess the performance of both. This is deliberately done to rule out the human factor in planning, since a human planner is capable of making adjustments outside the normal restrictions of the replenishment process. These are for instance temporarily shortened lead times, last minute changes to the delivery size or even rescheduling of its own production planning when a material is unavailable.

The simulation model assumes identical production lead times for both the traditional and VMI process. Under VMI however, the delivery lead time is

shortened to one week, since the supplier is able to deliver from stock.

Both the processes in the model have been validated with help of the historical input data, and the information from the manufacturers' enterprise resource planning (ERP) system. The validation results for the traditional model are depicted in *Table 1*.

Table 1: Traditional process validation results

Value	Model difference
Average stock level	-39% (± 3)
Deliveries	1.0% (± 0.2)
Service level	-5.0% (± 0.3)

It can be seen from the results that the simulation model accurately represents the number of deliveries per year. The stock level and service level however show discrepancies. The model significantly underestimates the stock level and this underestimation even leads to out-of-stock situations which in turn influence the service level. The underestimation can be explained by the following:

- The stock level needs to be corrected for safety stock, which was not taken into account in the model. This would reduce the difference to a still significant -25%. The discrepancy is therefore still not justified;
- Out-of-stock situations in the model lead to a negative stock. This assumption is made since the historical demand data is leading and rescheduling the requirements is thus not possible. This negative stock leads to a lower average stock level, but also to a lower service level.

Since the safety stock factor and the factor of negative stock are part of both the traditional and VMI simulation model, comparison between both is possible.

The results of the validation of the VMI model are presented in *Table 2*. The validation focused on assessing the number of productions in the actual situation (for which a pilot in the real production environment was initiated) and the model. Results show that the number of productions at the supplier is represented by the model rather accurately. Inaccuracies do occur when very stringent constraints are used, but this is due to the fact that in reality not all materials are required every week.

Table 2: VMI validation results (SCS=Supply Chain Stock)

VMI constraints [weeks]		Supplier productions per year		
SCS_{min}	SCS_{max}	Expected	Model	Actual
2	3	52	46	...
2	7	7	8	...
2	13	5	5	5

With the information presented in this section it is concluded that the model is suitable for comparing between processes, although the stock level is underestimated due to the assumption that in out-of-stock situations, the stock level can become negative.

EXPERIMENTS AND RESULTS

In order to assess the performance of VMI and determine the minimal required constraints, multiple experiments were carried out. The following input parameters were varied:

- Production size – this value represents the production requirement at the customer. For each day of the year, 10 different samples were taken. An exponential distribution was used here to simulate erratic demand behaviour;
- Production seasonality – the seasonality determines the possibility of a demand occurring at a specific day. As with the production size, 10 different samples were taken for each day of the year;
- Forecast accuracy – based on the production size the forecast accuracy was sampled 10 times for the 13 weeks of every forecast (in total 53 in a year). The average forecast accuracy was assumed to decrease with increasing week number.
- Minimum supply chain stock – for each sample of the above three parameters the minimum supply chain stock is varied from 1 up to 13 weeks. Dependent of course on the maximum supply chain stock, which it should not exceed;
- Maximum supply chain stock – same as minimum supply chain stock.

On closer inspection of the above one will see the extent of the data involved in these simulations. This is one of the strengths of the simulation model. It is capable of handling a very large amount of data and process the information quickly. In fact the input parameters mentioned above imply a total of more than

4500 simulation runs. Which resulted in 8100 data points, with a total simulation run time of approximately 35 minutes. Running an analysis for the company's 1400 specific materials even led to a much larger value: 85000 simulation runs, amounting to more than 1.5 million data points.
The results of the experiments with varying input data were divided into two separate analyses. One for the performance of VMI and the other for the determination of VMI constraints.

The results of the performance analysis of VMI are presented in *Table 3*.

Table 3: Performance analysis results

	Traditional	VMI	Difference
Average Supply chain stock	5440	19074	251%
Average productions	22	8	-64%
Average deliveries	22	27	20%
Average SL	98%	99%	1%

From the results it becomes clear that the total supply chain stock under the proposed form of VMI will increase significantly. This is due to the fact that the number of production runs at the supplier is reduced. Although it should be noted that this is not a desirable situation for every material, for most of the materials that the soft drink manufacturer uses, it provides a significant cost advantage. On top of that the service level is increased marginally. The number of deliveries however may increase under VMI. Cost calculations with the actual data of the manufacturer however showed that the influence of this factor is small. In fact an overall cost saving of approximately 17% was possible for several suppliers when moving to VMI. The previously mentioned pilot study confirmed this value, with a cost saving of more than 18%.

The results of the analysis of VMI constraints are presented in *Table 4*. These results can be read as follows: For instance if a customer would agree with its supplier to a minimum supply chain stock of 1 week (row 1) and a maximum supply chain stock of 7 weeks (column 7), the theoretical service level will be 98% under VMI. This implies in this analysis that for 98% of the deliveries no corrective action is required. Increasing the maximum supply chain stock will further increase this number to 100%.

Table 4: VMI constraints analysis results

Service level behaviour under VMI

Maximum supply chain stock (in weeks) →

Minimum supply chain stock (in weeks) ↓	1	2	3	4	5	6	7	8	9	10	11	12	13
1	52%	70%	84%	85%	94%	94%	98%	100%	100%	100%	100%	100%	100%
2	-	70%	84%	85%	94%	94%	97%	100%	100%	100%	100%	100%	100%
3	-	-	84%	85%	94%	94%	97%	100%	100%	100%	100%	100%	100%
4	-	-	-	85%	94%	94%	97%	100%	100%	100%	100%	100%	100%
5	-	-	-	-	95%	94%	98%	100%	100%	100%	100%	100%	100%
6	-	-	-	-	-	98%	100%	100%	100%	100%	100%	100%	100%
7	-	-	-	-	-	-	100%	100%	100%	100%	100%	100%	100%
8	-	-	-	-	-	-	-	100%	100%	100%	100%	100%	100%
9	-	-	-	-	-	-	-	-	100%	100%	100%	100%	100%
10	-	-	-	-	-	-	-	-	-	100%	100%	100%	100%
11	-	-	-	-	-	-	-	-	-	-	100%	100%	100%
12	-	-	-	-	-	-	-	-	-	-	-	100%	100%
13	-	-	-	-	-	-	-	-	-	-	-	-	100%

The results in Table 4 show that, for a maximum service level, a supply chain with a minimum of one week and maximum of eight weeks can be adhered to. A higher maximum supply chain stock reduces the probability of the supplier being unable to deliver, since low stock levels occur less frequently in a year. This leads to less out-of-stock situations.

With this in mind it is interesting to notice that increasing the minimum supply chain stock has less effect on the service level and only when that level reaches values of seven weeks and above the maximum service level can be attained. This is the result of the erratic demand behaviour of the material in the stochastic model. It can thus occur that the demand in a week increases to more than seven weeks. The minimum supply chain stock is then no longer sufficient.
On further close inspection of Table 4, one can see that in one case, minimum 1, maximum 7 weeks, the theoretical service level is actually higher than the case for a minimum of 2 weeks at the same maximum. The cause for this is identical to the one mentioned above, here the erratic demand behaviour is somewhat more unfavourable for the 2-7 weeks agreement.

CONCLUSIONS AND RECOMMENDATIONS

The investigation of the implementation of VMI with help of a simulation model leads to the following conclusions:

- The largest cost reduction is likely to occur in the production process at the supplier as was also concluded in the VMI pilot that was carried out;
- The service level generally increases for most articles since VMI is more robust to demand changes and forecast accuracy fluctuations because of the following reasons:

    - The delivery lead time under VMI is one week, which is shorter than usual for most articles. The delivery size is consequently determined on the forecast one week advance, instead of the less inaccurate forecasts for longer periods. This leads to a more accurate match between delivery size and actual demand;
    - Since the number of productions of the supplier decreases under VMI, the number of moments that only the minimal supply chain stock is present is also reduced. This in turn decreases the probability that the supply chain stock is insufficient to adhere to the customer's requirements;
- The stock level at the customer is likely to reduce, that of the supplier will in most occasions increase;
    - The latter should not be seen as an issue since the increase of stock at the supplier facilitates the freedom of production planning of the supplier and for most materials this will lead to a cost reduction;
- Under VMI the number of deliveries may increase;
    - This is due to the fact that the delivery size will better match the actual demand and situations of overstocking will occur less frequently. It however implies that the stock at the customer has to be replenished more often;
    - Suppliers that deliver from remote locations are therefore less suitable for VMI, although this issue can of course be covered in an agreement.

The simulation model that has been used for this analysis is fairly simple to facilitate fast calculation. For future research is recommended that the model is further developed to analyse specific cases in more detail. The model currently did not take perishable goods into account, which is an issue for several materials. If such an extension were to be implemented, the model can be used for other businesses as well. Finally it is advised that more detailed cost calculations are carried out for specific situations, in order to assess the overall benefit of implementing VMI at a company.

REFERENCES

Carey, Catherine, Kinane, Pat en Praznik, Debra, 2005. *Vendor Managed Inventory and Collaborative Forecasting*.. sl : Uniform Code council, 2005. Connect Conference.

Kumar, Phani en Kumar, Muthu, 2003. *Vendor Managed Inventory in retail industry.* sl : Tata consultancy services..

Mroz, Martin Frederick 2001. *Knowledge in business processes: The VMI Case*. Vancouver, Canada : University of British Columbia,

Sari, Kazim 2008, *On the benefits of CPFR and VMI: A comparative simulation study*. 113, Istanbul : sn, 2008, International journal of production economics, pp. 575-586.

Southard, Peter B. and Swenseth, Scott R. St. Paul 2008. *Evaluating vendor-managed inventory(VMI) in non-traditional environments using simulation*.: University of St. Thomas, 2008, Int. J. Production Economics, pp. 275-287.

Veeke, H. P.M. and Ottjes, J. A. 2000. *Tomas: Tool for Object-Oriented Modelling and Simulation*. Proc. of the Business and Industry Simulation Symposium [SCS], pp. 76-81 Washington D.C.

FLUID FLOW SIMULATION

CANDU LIQUID INJECTION SHUTDOWN SYSTEM HYDRAULIC MODELING

Ilie Prisecaru and Daniel Dupleac
Power Plant Engineering Faculty
Politehnica University of Bucharest
313 Splaiul Independentei, 060042, sector 6, Bucharest, Romania
E-mail: prisec@cne.pub.ro, danieldu@cne.pub.ro

Niță Iulian
Center for Engineering and
Technology for Nuclear Objective
Bucharest, Magurele
nitai@router.citon.ro

KEYWORDS
Nuclear engineering, Model development, Model evaluation, Lumped parameter, Continuous simulation

ABSTRACT

The paper presents the mathematical model and hydraulic analysis of the CANDU Liquid Injection Shutdown System (LISS). The mathematical model for the LISS equipments was developed starting from the conservation law of mass, momentum and energy. The mathematical model was coded in ACSL simulation language. The main purpose of simulation and hydraulic analysis of the LISS is to asses the system performance and possible improve the efficiency of the system. The main results and conclusions of the preliminary results are presented.

INTRODUCTION

In all nuclear reactors, there are special safety systems which are specifically designed to mitigate the consequences of a serious process failure. These systems perform no function in the normal operation of the plant. In CANDU reactors, there are two separate and independently reactor shutdown system, named: a. Shutdown System Number 1 (SDS1) and b. Shutdown System Number 2 (SDS2). The LISS is part of the reactor SDS2. The injection system must quickly shut down the reactor by injecting a neutron absorbing liquid (gadolinium nitrate dissolved in heavy water) directly into the moderator in the reactor core.

A schematic of the LISS is presented in Figure 1 (AECB, 1993). The neutron absorbing gadolinium nitrate solution (called "poison") is stored in six identical pressure vessels (poison tanks) located in an accessible part of the reactor building. Each poison tank feeds one injection line which passes through the calandria vault to an injection nozzle passing horizontally through the reactor core. A single helium supply tank contains high pressure helium to force the liquid poison from the poison tanks into the core. An array of six quick opening valves in the line from the helium supply tank to the poison tanks isolate the poison tanks from the high pressure helium. Upon opening of these valves, the helium pressurizes the poison tanks and injects the liquid poison into the reactor. A floating polyethylene ball in each poison tank seals on a seat at the bottom of the tank when the tank is empty of liquid, preventing passage of the helium to the calandria.

There is a liquid-to-liquid interface between the poison solution and the moderator as shown in Figure 1. Motion of the interface is caused by the poison very slowly migrating from an area of high concentration to an area of low concentration. Also, physical motion of the liquid back and forth in the line causes mixing of the poison solution with the moderator.

Upon injection signal receipt, the quick opening valves of those channels open. The pressurized helium in the helium supply tank flows through the valve array into the helium header, pressurizing this. The pressurized helium in the helium header forces the liquid in the poison tanks down the injection lines and into the calandria. The balls in the poison tanks are forced down and are sealed on the bottom seat when the tank is empty, preventing further flow. Time from the opening of the quick opening valves to the seating of the poison tank balls is approximately 1 second.

For the design basis initiating events the LISS must have sufficient speed and negative reactivity depth to reduce the reactor power to levels consistent with available cooling (AECB, 1993). The poison injection time from the tank into the calandria is an important safety parameter. An accurate estimation is required to validate the system design.

The paper underlines the LISS model development and the preliminary obtained results.

MODEL DESCRIPTION

The mass balance equation for the helium tank is written as:

$$\frac{dM_{He}}{dt} = -W_{He} \tag{1}$$

Helium is treated as a perfect gas, thus (Wark, 1983):

$$\rho_{He} = \frac{p_{He}}{R_{He} T_{He}} \tag{2}$$

and

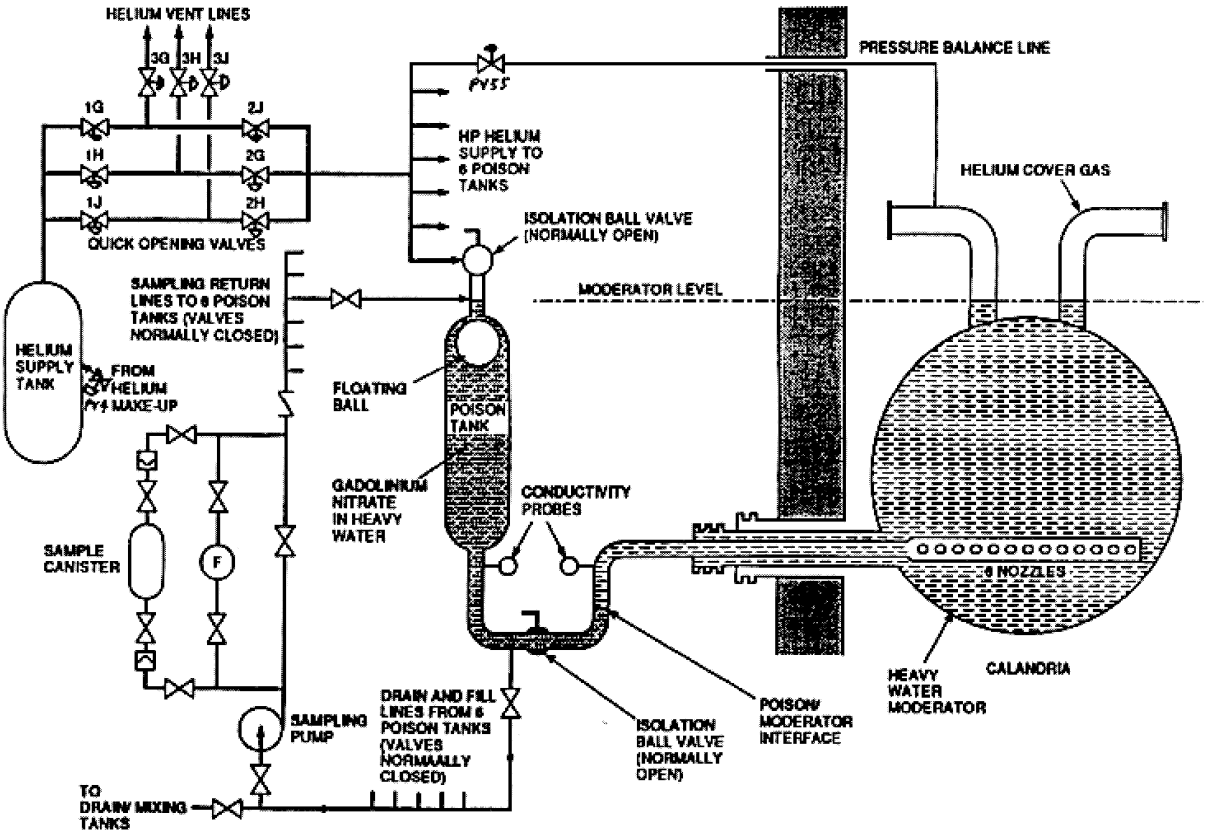


Figure 1: CANDU-6 Liquid Injection Shutdown System

$$M_{He} = \rho_{He} V_T \tag{3}$$

The energy balance equation for the helium tank is written as:

$$\frac{d(u_{He}\rho_{He})}{dt} = \frac{Q_{He} - W_{He}h_{He}}{V_T} \tag{4}$$

The helium expansion process inside the tank is considered adiabatic, thus $Q_{He}=0$. We have the following relationships between the thermodynamic variables in equation (4) (Wark, 1983):

$$u = h - pv \tag{5}$$

and

$$h = c_p T \tag{6}$$

Thus, after some algebras manipulations we find the following expression for the total helium internal energy:

$$u_{he}\rho_{He} = p_{He}\left(\frac{c_{pHe}}{R_{He}} - 1\right) \tag{7}$$

and the equation (4) becomes:

$$\frac{dp_{He}}{dt} = -\frac{W_{He}c_{pHe}T_{He}}{V_T\left(\frac{c_{pHe}}{R_{He}} - 1\right)} \tag{8}$$

The array of six quick opening valves and the line from the helium supply tank to the poison tanks are lumped in one equivalent pipe characterized by an equivalent conductance, *C*. A simplified form of momentum balance equation is used for this part of the system:

$$W_{He} = C\sqrt{\rho_{he}\Delta p_{He}} \tag{9}$$

However, a simplified model can be employed for the calculation of helium pressure variation. In this model, the helium pressure can be calculated either from isothermal expansion process or isentropic process. In this case, the helium pressure is given by (Wark, 1983):

$$p_{He} = p_{He0}\left(\frac{V_{He0}}{V_{He}}\right)^{\gamma} \tag{10}$$

The helium expand through the valve array, the helium header and then through the poison tank. The

pressurization of the valve array and the helium header is considered an instantaneous process, thus the initial helium volume account for these volumes together with the corresponding decrease of initial helium pressure. The increase of helium volume when the gas penetrates to the poison tank is calculated taking into account that the liquid poison is incompressible:

$$V_{Hept} + V_{ppt} = V_{pt} = const \tag{11}$$

and thus:

$$\frac{dV_{Hept}}{dt} = -\frac{dV_{ppt}}{dt} = A_{pt} v_p \tag{12}$$

The balance equation written for the poison tank take into account that during the injection phase, a region of gas, helium, and a region of liquid, gadolinium nitrate in heavy water, will exist.

For the gas region, the mass balance equation is:

$$\frac{dM_g}{dt} = W_{He} \tag{13}$$

The energy balance equation may be written as:

$$M_g \frac{dh_g}{dt} = V_g \frac{dp_g}{dt} + Q_g \tag{14}$$

The injection process is adiabatic, thus $Q_g=0$. With this assumption and (6), the equation (14) becomes:

$$\frac{dT_g}{dt} = \frac{V_g}{M_g c_p} \frac{dp_g}{dt} \tag{15}$$

The mathematical equation for an ideal gas undergoing an adiabatic process is (Wark, 1983):

$$pV^{\gamma} = const \tag{16}$$

From equation (16) we have the relation between gas volume and its pressure as:

$$V_g = \frac{const}{p_g^{1/\gamma}} \tag{17}$$

Combining mass and balance equations for gas region and after some algebras manipulations we find:

$$\frac{dV_g}{dt} = \frac{W_{He} R_g T_g}{p_g} + \frac{1}{\gamma} \frac{V_g}{p_g} \frac{dp_g}{dt} \tag{18}$$

The total volume of the poison tank is constant, thus;

$$V_g + V_l = V_p = const \tag{19}$$

and

$$\frac{dV_l}{dt} = -\frac{dV_g}{dt} \tag{20}$$

Replacing (20) in equation (18), the equation describing the liquid poison volume dynamics is expressed by the relation:

$$\frac{dV_l}{dt} = -\frac{W_{He} R_g T_g}{p_g} - \frac{1}{\gamma} \frac{V_g}{p_g} \frac{dp_g}{dt} \tag{21}$$

The mass balance equation for the liquid poison region is written as:

$$\frac{dM_l}{dt} = -W_p \tag{22}$$

and the energy balance equation for the liquid poison region is written as:

$$\frac{d(M_l h_l)}{dt} = V_l \frac{dp_l}{dt} - W_l h_l + Q_p \tag{23}$$

Considering the injection process as adiabatic, thus $Q_p=0$, and combining the mass and balance equations for liquid poison region and after some algebras manipulations we find the following equations:

$$a_{11} \frac{dh_l}{dt} + a_{12} \frac{dp_l}{dt} = b_1 \tag{24}$$

$$a_{21} \frac{dh_l}{dt} + a_{22} \frac{dp_l}{dt} = b_2 \tag{25}$$

this must be solved together. In equations (24) and (25) the coefficients are:

$$a_{11} = V_l \frac{\partial \rho_l}{\partial h} \tag{26}$$

$$a_{12} = V_l \frac{\partial \rho_l}{\partial p} - \frac{\rho_l (V_T - V_l)}{p_l \gamma} \tag{27}$$

$$a_{21} = \rho_l + h_l \frac{\partial \rho_l}{\partial h} \tag{28}$$

$$a_{22} = h_l \frac{\partial \rho_l}{\partial p} - 1 \tag{29}$$

$$b_1 = W_{He} \frac{\rho_l T_g R_{He}}{p_l} - W_l \tag{30}$$

$$b_2 = -\frac{W_l h_l}{V_l} \tag{31}$$

Solving the equation (24) and (25), we obtain the following relation for liquid poison pressure:

$$\frac{dp_l}{dt} = \frac{W_{He} \frac{p_{He}}{p_l} \frac{\rho_l}{\rho_{He}} - W_l}{V_l \left(\frac{\partial \rho_l}{\partial p} + \frac{1}{\rho_l} \frac{\partial \rho_l}{\partial h} \right) - \frac{\rho_l (V_T - V_l)}{p_{He} \gamma}} \tag{32}$$

The initial portion of system filled with gadolinium nitrate is considered as a moving control volume of fluid with same velocity as the velocity of fluid (Singh, 2009). The conservation of momentum, for the combined poison tank and injection line momentum equation is (Toderas 1993):

$$V\rho \frac{\partial v}{\partial t} - \frac{1}{2} A_{in} \rho_{in} v_{in}^{\ 2} + \frac{1}{2} A_e \rho_e v_e^{\ 2} = - A_e p_e + A_{in} p_{in} - Fv - LA \sin(\theta) \rho g \tag{33}$$

The term F represents the wall drag coefficient and could also contain the local form losses (Toderas 1993):

$$F = \frac{1}{2} \left(f \left(\frac{L}{d} \right) + \zeta \right) A \rho |v| \tag{34}$$

From mass balance equation taking into account the incompressibility of liquid poison, we have (Singh, 2009):

$$v_e = \frac{A}{A_e} v_{in} \tag{35}$$

Thus, the momentum flux terms become:

$$\frac{1}{2} A_{in} \rho_{in} v_{in}^{\ 2} - \frac{1}{2} A_e \rho_e v_e^{\ 2} = \frac{1}{2} \rho A \left(1 - \left(\frac{A}{A_o} \right)^2 \right) v^2 \tag{36}$$

where v represent the fluid velocity in pipe (liquid poison velocity), and A_o is the outlet surface area, i.e. area total area of holes from which poison goes out.

The other term of equation (33) is evaluated as follow. The gravitational pressure term is evaluated as elevation pressure differential between poison injection line and poison liquid surface. The inlet pressure is considered the helium pressure, and the outlet pressure is the moderator pressure which is considered constant.

The equation (33) gives the variation of velocity of liquid poison during injection and thus the flow rate of liquid poison injected. This equation can be also used for tracking the liquid poison front movement.

PRELIMINARY RESULTS

The mathematical models were coded in ACSL simulation languages (MGA, 1993). The ACSL coded program was used for the validation of the mathematical models. Afterward, the models will be implemented in the Modular Modeling System (MMS) code using the CompGen tools (Framatome Tehnologies, 1998a, 1998b).

In order to asses the system performance, we evaluate the helium gas pressure fall versus time and the quantity of poison injected versus time. The model will further be improved to allow the analysis of jet length versus time characteristics of liquid poison injected into moderator (Nawathe, 1991).

The main data used in the analysis is presented in Table I (AECB, 1993; CITON, 2005; KAERI, 2005).

TABLE I: Main data used in LISS analysis

Parameter	Value
Helium tank volume	1.13 m^3
Number of Poison Tanks/Injection Nozzles	6
Poison tank volume	0.079 m^3
Nozzle length	6.952 m
Nozzle hole diameter	0.0032 m
Helium pressure	8.3 MPa(g)
Poison density	1127 kg/m^3

The helium gas pressure fall calculated using the simplified model is shown in Figure 1, both for isothermal and isentropic expansion process. The actual gas expansion line will be between the two limit cases. The isentropic assumption gives conservative results as predict lower helium pressure as the gas expand. The helium pressure obtained this way was used to evaluate the

quantity of poison injected into moderator, as shown in figure 2.

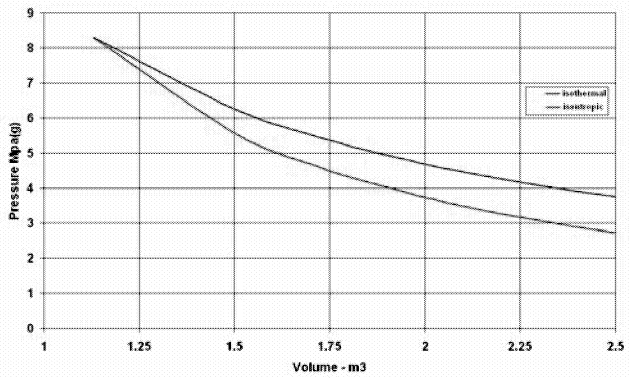


Figure 1: Helium gas pressure fall for isothermal and isentropic expansion process

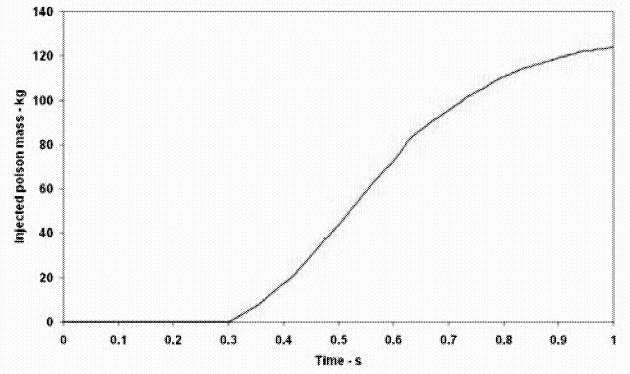


Figure 2: Quantity of poison injected into moderator (6 nozzle)

CONCLUSIONS

In this paper the mathematical model for the hydraulic analysis of liquid poison injection system was developed starting from the conservation law of mass, momentum and energy. The main aspects of the mathematical model are highlighted. The mathematical models have been coded in ACSL. The preliminary analysis results for helium pressure fall during gas expansion and the total mass of poison injected into moderator are presented.

However, the model is still under development. The further development will regard improvements to the present model, further assessment of the model and full analysis of the liquid injection system performance including jet propagation into calandria vessel.

REFERENCES

Atomic Energy Control Board (AECB), 1993. *Fundamentals of Power Reactors – Module Two: Nuclear Systems*. Training Manual.

Center for Engineering and Technology for Nuclear Objective (CITON). 2005 *Liquid Injection Shutdown Units Design Manual*

Korea Atomic Energy Research Institute (2005) - *Development of a 3D CFD Model for Velocity and Concentration Field of a Jet Flow in CANDU-6 Moderator Tank induced by High Velocity Injection of Liquid Neutron Poison* - KAERI/TR-2989/2005.

Framatome Tehnologies. 1998a. *MMS Theory Manual Release 5.1*.

Framatome Tehnologies. 1998b. *CompGen Users Manual*.

Nawathe, S., et al.,, "*Development of Liquid Poison Injection System (SDS-2) for 500 MW(e) PHWRs*" Bhabha Atomic Research Centre, 1991

Singh, R.K., et al., *Phenomena of reverse momentum pulse and fluid leak after poison injection in the hut down system of PHWR*, Nuclear Engineering and Design, Volume 239, Issue 10, October 2009, Pages 1760-1767

Todreas, N.E.; and M.S. Kazimi 1993, *Nuclear Systems, Thermal hydraulic Fundamentals*,Taylor &Francis.

Wark, K. 1983, *Thermodynamics*, Mc-Graw Hill, New York.

NOTATION

A – flow area, m^2
c_p – heat capacity, J/kg ^oC
d – diameter, m
f – friction coefficient
h – enthalpy, J/kg
L – length, m
M – mass, kg
p – pressure, Pa
R – gas constant, J/kg K
T – temperature, ^oC
u – internal energy, J/kg
V – volume, m^3
v – velocity, m/s
W – flow rate, kg/s
ρ – density, kg/m^3
ξ – local energy losses coefficient
γ - ratio of specific heats (isentropic coefficient)

Subscript

g – gas
He – helium
l – liquid
p – poison
pt – poison tank
pmp – pump

PROBABILISTIC MODELS FOR DISSOLUTION OF ETHYLCELLULOSE COATED MICROSPHERES

Marija Bezbradica, Ana Barat, Heather J. Ruskin, and Martin Crane
Centre for Scientific Computing and Complex Systems Modelling, School of Computing
Dublin City University, Ireland
email: mbezbradica@computing.dcu.ie

KEYWORDS
Discrete Simulation, Monte Carlo, Cellular Automata, Microspheres, Drug Dissolution

ABSTRACT

In the last few decades, a number of probabilistic models for drug delivery have been developed. Of particular interest are those that model controlled release systems to provide targeted dose delivery. Controlled release is achieved by using polymers with different dissolution characteristics. We present here a model based on Monte Carlo and Cellular Automata approaches, for simulating drug release from coated microspheres in the gastro-intestinal tract. Controlled release is obtained using ethylcellulose as the coating polymer. Modelling features, such as the drug and coating dissolution are non-trivial, since material is non-homogenously dispersed and the dissolution exhibits complex behaviour. Important underlying mechanisms of the process, such as erosion, are described here.

INTRODUCTION

Drug delivery systems (DDS) are pharmaceutical systems designed for transporting drugs into the body. The computational modelling of DDS is a constantly developing field, with the potential to become an integral part of pharmaceutical research, as some modern drug formulations are very complex and the influence of system composition and variables on the release profiles is not fully understood. *In-silico* modelling of DDS can, therefore, be of benefit in reducing the cost of experiments, (involving large amount of *in vitro* testing), as well as length of time needed to introduce the drug to the market. Controlled drug delivery systems are a type of DDS, the primary objective of which is to deliver drug at the desired rate to a targeted site in the body. Control is maintained by using polymers of different structures. Consequently, polymer dissolution is one of the most important problems to solve in achieving this type of release.

Here, we present a probabilistic model based on Monte Carlo (MC) and Cellular Automata (CA) approaches, for simulating controlled release from coated microspheres. The work is rooted in an ongoing industrial collaboration and aims to address the release problem for the drug (cyclosporine) in targeting the gastro-intestinal (GI) tract. As one of a number of suitable coating materials, we model the use of ethylcellulose (EC), a generally non-invasive polymer with good film-forming capabilities. It is both an inert and hydrophobic polymer but, when maintained under high humidity, can absorb water in large amounts (Geraghty 2004). In general, properties of ethylcellulose make it ideal for use in matrix agents, for prolonged release, or as a coating material.

THEORY

The first step in modelling the dissolution is to determine the *in vitro* dissolution rate under various external conditions, as measured in the dissolution test apparatus. Standard test methods for measuring those rates are outlined in the pharmacopoeias (European Pharmacopoeia, United States Pharmacopoeia). Here, we used the paddle apparatus II, (USP II), which allows us to mimic metabolic conditions, such as change of temperature, type of solution (pH and dynamic flux of the solution), by varying speed of the paddle, and monitor experimentally their influence on the drug dissolution rate.

Theoretical models will vary according to the type and complexity of the system, and can be used to explore experimental *in vitro* data. Theories are generally divided into three broad groups: mechanistic, empirical and probabilistic (Siepmann and Siepmann 2008). The first two groups take the top-down approach, start from known parameters and use specific deterministic release rate equations for each individual problem, (i.e. to represent physical laws and empirical information). The most important properties must be either known from the manufacturing process or be possible to calculate, which is not straightforward given the complex nature of the formulations. Conversely, for complex detail modelling, bottom-up theories, (incorporating stochastic assumptions), have the potential to yield better predictions while requiring less initial information. These simulate release curves by observing the probabilistic behaviour of individual particles and have the advantage of providing a simplified representation of the system,

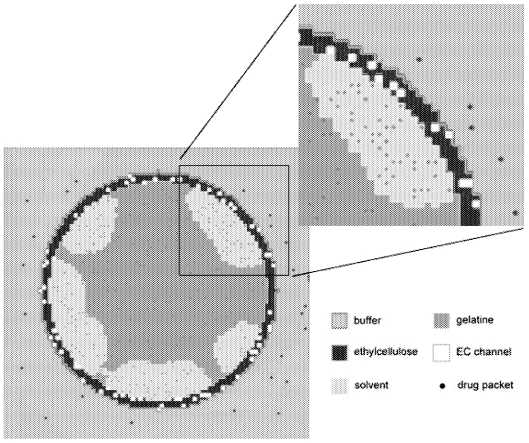


Figure 1: Simplified Internal Morphology of one 3D Sphere Simulating Drug Dissolution through Coating Layer (Ethylcellulose). Enlarged: Part of the Sphere with Definition of Model Cell Types

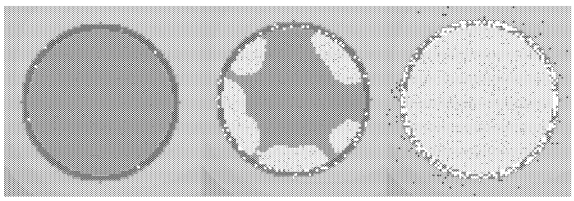


Figure 2: Simulation Stages: Initial Stage (left), During Release, with Gelatine (centre), After all Gelatine is Dissolved (right)

(Barat et al. 2008b).

Monte Carlo methods have numerous applications, ranging from fluid dynamics, space and traffic modelling and statistical physics, to financial analysis (Li et al. 2007, Sopasakis 2004, Tezuka et al. 2005). In one of the first direct MC models of DDS, microstructural changes in bioerodible polymers were simulated (Göpferich and Langer 1993). MC is also often used as a framework for Cellular Automata movement. In a CA system, a discrete grid of individual cells is defined, where each cell can have one of a number of defined states. The behaviour of each cell is governed by a set of rules that describe the local state transitions over discrete time steps, (Zygourakis 1990).

Drug Dissolution Phenomena

Important phenomena in polymeric drug delivery systems include: (i) diffusion: the motion of molecules from a region of higher to one of lower concentration i.e. flux. In the modelling of DDS, diffusion is often described by Fick's laws, (Crank 1975). In one dimension, Fick's first

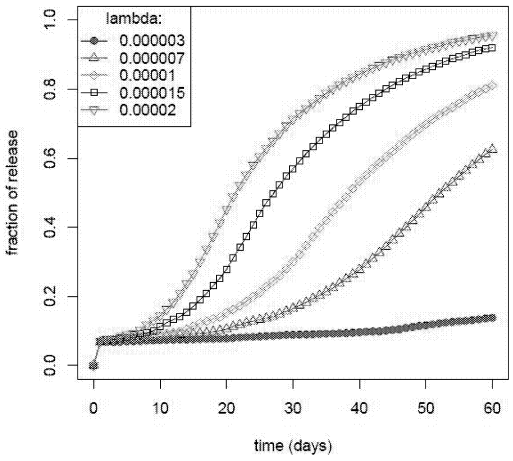


Figure 3: Release Profiles as a Function of the Degradation Rate Lambda (λ)

law can be represented as:

$$J = -D\frac{\delta c}{\delta x} \tag{1}$$

where J is the diffusion flux, D is the diffusion coefficient and c is the concentration; (ii) degradation: the process whereby chain scission occurs causing polymer chains to become oligomers and monomers, facilitating easy release. Degradation is the first step of erosion (Göpferich 1996); (iii) erosion: the loss of material from a polymer due to degradation. It determines the release rate of the drug (Siepmann and Göpferich 2001). When a polymer erodes it leaves space for the drug to be released from the compact, or for water penetration. Two types of erosion are defined (Langer and Peppas 1983): Surface erosion is a homogeneous process and represents the stage during which the size of the compact decreases while the shape remains constant; the compact loses material only from its surface; Bulk erosion is a heterogeneous process with material being lost from the whole compact, although the size remains constant.

MODEL FEATURES

In a previous project (Barat et al. 2008a), (Barat et al. 2008b), microspheres filled with polylactic-co-glycolic acid (PLGA) polymer were modelled. In these, drug is homogeneously dispersed throughout the sphere and contained within a polymer. Understanding the phenomena occurring in polymeric spheres was achieved through dimensional (3D) stochastic Cellular Automata and agent-based modelling. The rationale for introducing the agent-basis is simplification of the interaction of system elements using specifically designed entities (agents), which have simple properties and characteristics. The influence of a number of parameters defining release curves was investigated, including porosity,

drug loading, sphere size and type of neighbourhood, (Von Neumann or Moore). Some of the earlier results are illustrated (Figure 3), which show how the sphere porosity growth dynamics affect the release rate.

Current Work

The modelled device now consists of a drug, non-homogenously dispersed in a coated sphere, where the coating predominantly controls the rate of drug release. The aim of the model is to determine how different properties of the device, such as coating thickness or size of the sphere, affect the release rates. Monte Carlo and Cellular Automata methods are used to describe the system in terms of a 3D grid of cells, with erosion and diffusion as the main release mechanisms. This discrete approach is preferred over a continuous one, (used in mechanistic theories), due to straightforward mimicking of the real system which is discrete by nature. Each cell can have one of several possible states, described in the (Table 1). The state transitions are influenced by the states of all the cells in the Von Neumann neighbourhood for a 3D matrix (i.e. 26 neighbouring cells).
In its initial state, the drug is taken to be randomly dispersed in the form of "packets" inside a gelatinous sphere, coated by a polymer layer. Diffusion of the drug inside the sphere is represented as a random walk of drug packets, influenced by concentration differences between neighbouring cells. The highest probability for movement will be in the direction of the largest concentration gradient, based on the Fick's first law.
Behaviour of coating layer cells is based on work, (Göpferich and Langer 1993), where scission of the polymer chains and formation of pores follows the Erlang distribution:

$$e(t) = \lambda t e^{\lambda t}, t \geq 0 \quad (2)$$

The rate of the pore formation is characterised by λ which defines the lifetime, t, of an individual EC cell:

$$t = \frac{1}{\lambda} ln(U) \quad (3)$$

where U takes randomly distributed values between 0 and 1. When the cell lifetime is reached, the cell is considered to be eroded, and forms a channel through which the drug packet can diffuse. This occurs at a rate slower than the diffusion rate inside the coating, (solvent cells), reflecting the influence of permeability of the membrane (Laaksonen et al. 2009). The release rate is measured by counting the number of packets that reach the buffer zone, which is assumed to have perfect sink conditions.

RESULTS AND EXPLANATIONS

The simulation was developed in C++ with OpenGL used for graphical representation. The matrix size was

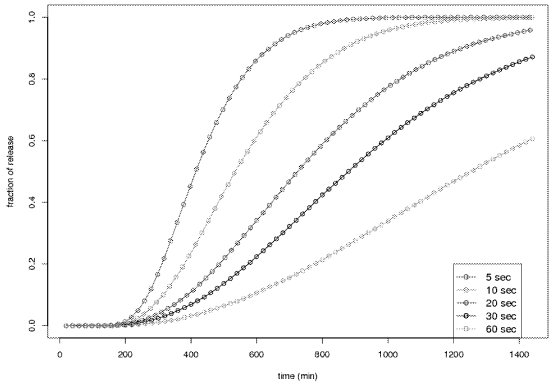


Figure 4: Release Profiles as a Function of the Simulation Time Interval

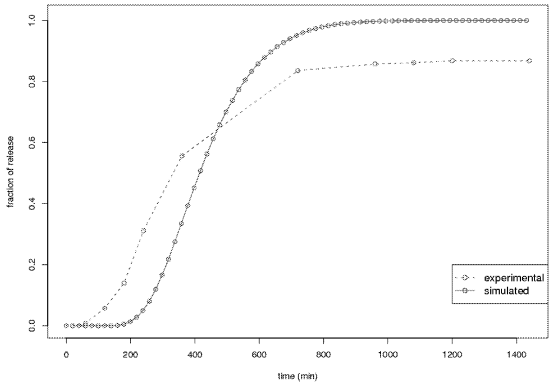


Figure 5: Simulated Release Profiles Against Experimental Release Profiles. Experimental Data Provided by Sigmoid Pharma Ltd.

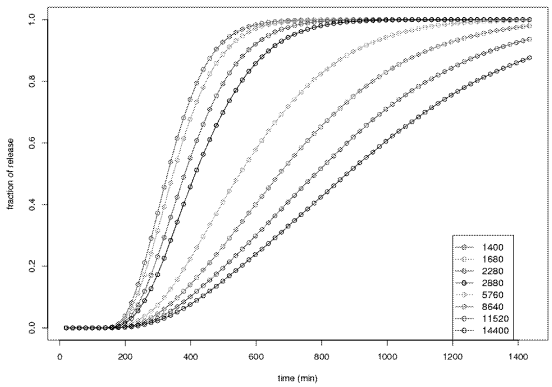


Figure 6: Release Profiles as a Function of the Coating Degradation Rate Lambda

Cell type	Behaviour description
Buffer	Drug is released when it reaches buffer zone. Cell type acts as a perfect sink.
Ethyl-cellulose	Drug-free coating layer. Assigned lifetime based on the λ (degradation rate) parameter. Forms an EC channel cell upon complete erosion.
EC-channel	Gelatine and drug can diffuse through EC channel cells.
Gelatine	Fixed lifetime. During diffusion through solvent cells, facilitates movement of drug "packets".
Solvent	Gelatine erodes into the solvent. Drug can diffuse through the latter.
Drug packet	Drug, initially dispersed in gelatine cells. Each cell can hold a maximum (saturation) amount of drug "packets".

Table 1: Set of Cell Types and Rules of Behaviour.

200 x 200 x 200, with cell size being 10μm. The averaging of the values for 24 repeated runs showed a negligible variation in results for a given set of same parameters.

Model parameters, the effects of which on drug release rates were studied in this work were: the size of the microsphere, the erosion rates of gelatine and ethylcellulose and the effect of coating thickness. The initial values were set according to the available *in vitro* data and unless otherwise stated, the erosion rates of gelatine and ethylcellulose are 90 minutes and 2 days, respectively. The sphere was taken to be 1.43 mm in diameter, and to contain 5% of EC coating. Drug loading is kept constant for all simulations at 10.8% of the mass.

The first step in modelling was to determine the most appropriate time interval in which cell states should be updated. This essential parameter directly determines the diffusion and has order of magnitude in seconds, (from Fick's first law and dimensional analysis). Simulations were then performed for intervals between 5 seconds and 1 minute, (Figure 4). By comparing results against experimental data, (Figure 5), a time interval of 5 seconds was chosen for all subsequent simulations.

The effect of porosity in the coating layer was investigated by varying lifetime of EC cells, (i.e. varying λ), to obtain different release behaviours for different lifetimes of EC chains, (Figure 6). The slower release rate of the drug is due to decreased porosity, i.e. slower channel formation occurring in the coating.

Weight gain of the coating thickness is also one of the primary factors influencing drug formulation and performance. Here, we vary the coating levels from 4% - 8%, (Figure 7). The produced release curves qualitatively reproduce reduction in diffusion rate, but the impact of adjusting the weight gain is somewhat smaller than experienced *in vitro*. This is expected, to some extent, as perfect sink conditions are modelled in this case. In

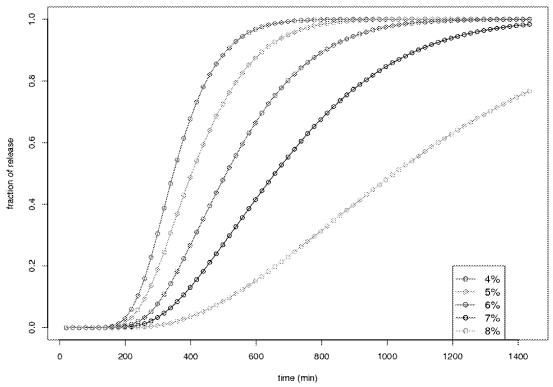


Figure 7: Release Profiles as a Function of Coating Weight Gain

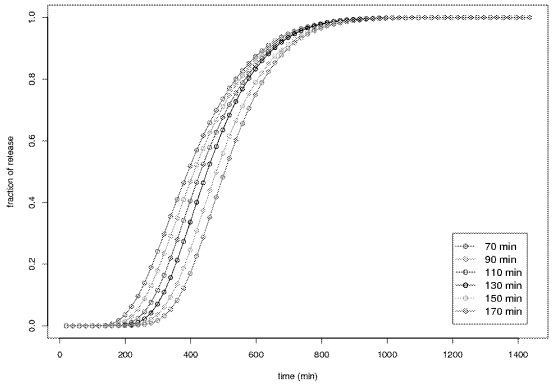


Figure 8: Release Profiles as a Function of Gelatine Lifetime

reality, concentration of ethylcellulose in the boundary layer inhibits movement of drug packets and some local saturation occurs. This phenomenon must be taken into account in future model extensions.

As can be seen, (Figure 8), the lifetime of the gelatine carrier can be considered to be negligible in terms of influence on final release rate. However, gelatine is an important controller of the drug release in the initial stages, as it influences the "burst effect" by accelerating drug transfer through EC channels. The size distribution of microspheres determines the actual mean and variance of release rate, (Figure 9). This feature suggests refinement of initial parameters is necessary, (e.g. coating thickness or drug loading), as these are directly dependent on the size of the sphere. Increasing microsphere size mostly slows down drug release due both to the capability of the device to hold more drug and the fact that larger spheres have heavier and thus less permeable coatings.

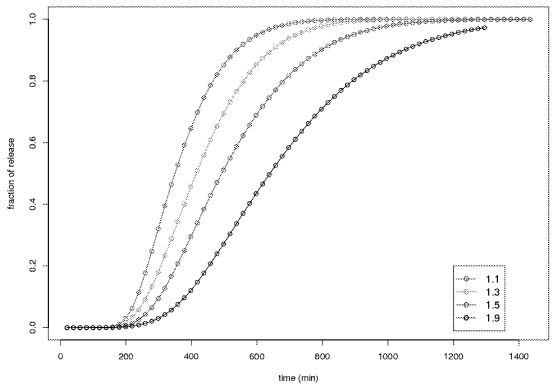


Figure 9: Release Profiles as a Function of Sphere Diameter

FUTURE WORK

Although simulated results do not reflect quantitative *in vitro* data precisely, (Figure 5), results obtained are promising. These show that the model predicts qualitatively similar behaviour compared to that seen for experimental release curves. Results also show that the behaviour of EC cannot be modelled using erosion only, but that other phenomena, such as swelling, which influence volume increase due to the increased hydration of the broken polymer chains, have to be taken into account (Atyabi et al. 2004). A future focus will thus be incorporation, into the model, of the swelling effect, caused by adding additives to the coating structure.

Additionally, the model will be augmented to include the effect of different polymer coatings and different media surrounding the drug (biphasic release) in order to simulate different stages of the GI tract environment. The deduction of unknown parameters from the experimental release curves using reverse engineering and inverse Monte Carlo methods is a long-term aim. Nevertheless, the approach to date enables comparison of a set of simulated and experimental release curves allowing us to determine key parameters and their values for this novel formulation and to reproduce qualitative behaviour. This enables us to compare a large set of simulated release curves with real ones in order to estimate best fit parameter values.

ACKNOWLEDGEMENTS

Authors acknowledge the financial support from the Irish Research Council for Science, Engineering and Technology (IRCSET), grant number P07669.

REFERENCES

Atyabi F.; Vahabzadeh R.; and Dinarvand R., 2004. *Preparation of Ethylcellulose Coated Gelatin Microspheres as a Multiparticulate Colonic Delivery System for 5-Aminosalicilic Acid. Iranian Journal of Pharmaceutical Research*, 2, 81–86.

Barat A.; Crane M.; and Ruskin H., 2008a. *Quantitative multi-agent models for simulating protein release from PLGA bioerodible nano- and microspheres. Journal of Pharmaceutical and Biomedical Analysis*, 48, 361 – 368.

Barat A.; Ruskin H.; and Crane M., 2008b. *3D Multi-agent models for protein release from PLGA spherical particles with complex inner morphologies. Theory in Biosciences*, 127, –.

Crank J., 1975. *Mathematics of Diffusion.* Clarendon: Oxford.

Geraghty M., 2004. *Investigation of ibuprofen release from ethylcellulose matrix compacts.* Ph.D. thesis, University of Dublin, Trinity College.

Göpferich A., 1996. *Mechanisms of polymer degradation and erosion. Biomaterials*, 17, 103 – 114.

Göpferich A. and Langer R., 1993. *Modeling of Polymer Erosion. Macromolecules*, 26, 4105–4112.

Laaksonen T.; Laaksonen H.; Hirvonen J.; and Murtomäki L., 2009. *Cellular automata model for drug release from binary matrix and reservoir polymeric devices. Biomaterials*, 30, 1978 – 1987.

Langer R. and Peppas N., 1983. *Chemical and Physical Structure of Polymers as Carriers for Controlled Release of Bioactive Agents: A Review. Journal of Macromolecular Science*, 23, 61 – 126.

Li D.; Hohne D.; Bortz D.; Bull J.; and Younger J., 2007. *Modeling bacterial clearance from the bloodstream using computational fluid dynamics and Monte Carlo simulation. Journal of Critical Care*, 22, 344 – 344.

Siepmann J. and Göpferich A., 2001. *Mathematical modeling of bioerodible, polymeric drug delivery systems. Advanced Drug Delivery Reviews*, 48, 229 – 247.

Siepmann J. and Siepmann F., 2008. *Mathematical modeling of drug delivery. International Journal of Pharmaceutics*, 364, no. 2, 328 – 343.

Sopasakis A., 2004. *Stochastic noise approach to traffic flow modeling. Physica A: Statistical Mechanics and its Applications*, 342, 741 – 754.

Tezuka S.; Murata H.; Tanaka S.; and Yumae S., 2005. *Monte Carlo grid for financial risk management. Future Generation Computer Systems*, 21, 811 – 821.

Zygourakis K., 1990. *Development and temporal evolution of erosion fronts in bioerodible controlled release devices. Chemical Engineering Science*, 45, 2359 – 2366.

WEB REFERENCES

European Pharmacopoeia, http://www.edqm.eu
United States Pharmacopoeia, http://www.usp.org

IMPACT OF GASEOUS AND PARTICULATE MATTER EMISSION FOR FLUID CATALYTIC CRACKING UNIT

Wael Yateem
Chemical Engineering Department
Loughborough University
Loughborough LE11 3TU

Vahid Nassehi
Chemical Engineering Department
Loughborough University
Loughborough LE11 3TU
Email: v.nassehi@lboro.ac.uk

Abdul R. Khan
Department of Environment
Technology and Management
College for Women
Kuwait University
Kuwait

Bahareh Kaveh-Baghbaderani
Chemical Engineering Department
Loughborough University
Loughborough LE11 3TU

KEYWORDS
Dispersion model, Aermod, emissions, FCC, pollutants exeedance

ABSTRACT

Fluid catalytic cracking unit is a major part of petroleum refineries as it treats heavy fractions from various process units to produce light ends (valuable products). FCC unit feedstock consists of heavy hydrocarbon with high sulphur contents and the catalyst used is zeolite impregnated with rare earth metals i.e. Lanthanum and Cerium. Catalytic cracking reaction takes place at an elevated temperature in fluidized bed reactors generating sulphur-contaminated coke on the catalyst with large quantity of attrited catalyst fines. In the regenerator, coke is completely burnt producing SO_2, PM emissions are mainly due to high attrition of cold makeup catalyst charge and operating conditions, vapour velocity particle velocity, particle collision and particle degradation. This study is dedicated to the quantitative analysis of the impact of harmful emissions resulting from FCC units on the environment.

INTRODUCTION

Fluid catalytic cracking (FCC) of heavy ends into high value liquid fuels is commonly carried out in the oil refining industry. In this process the heavy feedstock containing sulphur as a major contaminant is cracked to light products. Sulphur is redistributed in the liquid and gaseous products and coke on the catalyst. In the regenerator coke with sulphur contamination is completely burnt and flue gas containing SO_2 is discharged with catalyst fines produced, mainly due to high attrition of cold makeup catalyst charge and operating conditions i.e. vapour velocity, particle velocity, particle collision and particle degradation (Abdul Wahab et al., 2002).
In the present work, a comprehensive emission inventories from FCC unit in an oil refinery have been prepared. These inventories are calculated based on complete combustion of sulphur and coke impregnated on the catalyst in the regenerator. Mainly for SO_2 and Particulate matter (PM) emission rates are calculated accurately using material balances for a yearlong period considering seasonal variations in the operation of the process unit, Yateem et al., (2010). PM emission inventory is used in dispersion model to assess its impact on the immediate surroundings of the refinery.
The most advanced dispersion model Aermod (Caputo *et al.*, 2003; Isakov *et al.,* 2007; Kesarkar *et al.,* 2007) has been selected for prediction ground level concentration of PM based on comprehensive year long emission inventory of FCC unit.
Aermod is a dispersion model that uses Gaussian distribution for the stable conditions and non-Gaussian probabilities density function for the unstable conditions. Aermet (Aermod pre-processor) provides planetary boundary layer parameters over a high altitude to yield accurate predicted concentration values for a given meteorological conditions. It can accommodate large meteorological data (multiple years). Aermap (Aermod pre-processor) generates regular receptors over a given terrain for the evaluation of pollutants ground level concentrations. The meteorological data for year 2008 are obtained and are used in pre-processor Aermet to generate planetary boundary layers parameters. These generated data are used in Aermod for actual emission rates to predict ground level concentrations of PM and study the influence of prevailing meteorological conditions at this particular site.

MODEL APPLICATION

1. *Input Data*

Aermod dispersion model implementation requires the following items of data:
1. Source information: including pollutant emission rate (g/s), location coordinates in Universal Transverse Mercator (UTM) (m), base elevation from the sea level (m), stack height (m), exit stack inner diameter (m), exit stack gas velocity (m/s), and exit stack gas temperature (^oK).
2. Meteorological information for the region of interest: includes anemometer height (m), wind speed (m/s), wind

direction (flow vector from which the wind is blowing) (in degrees clockwise from the north), ambient air temperature (^oC), stability class at the hour of measurement (dimensionless) and hourly mixing height (m).
3. Receptor information: This can be specified or generated by the program to predict the pollutants' concentrations at the selected receptors.
The entire required source input data are obtained from FCC unit in the refinery. A stack of 80 m height, an inner diameter of 2.3 m, with an average exit gas velocity of 20 m/s and exit gas temperature of 550 ^oK are fed into the model. Monthly emission variation is considered with total SO_2 emission rate of 6089.2 g/s and total PM emission rate of 302 g/s as presented in detail (Yateem *et al.* 2010).

2. *Area of Study*

The area of study in this work covers portion of Ahmadi governorate in the state of Kuwait. Fahaheel area is adjacent to the petroleum refinery has one of the Kuwait EPA air quality monitoring station located at a polyclinic. Both areas Fahaheel and Ahmadi are surrounded by arid desert in the west side and bordered by the Persian Gulf from the east.
Two different types of receptor coordinates are used as input to the Aermod model to predict the ground level concentration of SO_2 and PM, these are:
1. Discrete Cartesian receptors specified at the sensitive areas viz., a school, a shopping area and EPA monitoring stations in Fahaheel. A hospital and petroleum services companies' offices are selected in Ahmadi.
2. Uniform Cartesian Grid receptors covering the entire area of study, where the FCC stack (emissions source) is located almost in the centre of the mesh grid.
The receptors selected are based on the actual sites in a UTM location coordinate of the area of interest map. Table 1 shows the selected discrete receptors information.
The uniform grid receptors of a total 1764 (42 x 42) were divided into (Δx = 300 m x Δy = 250 m) to cover about 12 x 10 km area of study. The optimum selection of the mesh size is based on the computational accuracy and time.

Table 1 the selected discrete receptors information Coordinates are related to the centre of wind rose

ID Number	Discrete receptor identity	X-coordinate	Y-coordinate
1	Fahaheel Polyclinic	219854.25	3219765.79
2	Petroleum Services Offices in Ahmadi	216666.87	3220105.63
3	School in Fahaheel	220300.00	3219820.85
4	Ahmadi Hospital	213458.86	3221523.64
5	Shopping area in Fahaheel	219274.32	3219554.21

RESULTS AND DISCUSSION

A yearlong comprehensive metrological data are processed by Aermet to generate boundary layer parameters and to pass all meteorological observations to Aermod.
Figure 1 shows wind direction and magnitude for a period of year 2008. It is observed that most of the time; the prevailing wind direction is from North West. There is strong influence from the neighbouring Persian Gulf as the refinery is located at the coast, resulting into strong sea breeze blowing from East direction. Wind class frequency distribution for the entire year confirming 2 % calm conditions, while 39.8 % is between 3.6 - 5.7 m/s. the highest wind class 8.8-11.1 m/s is less than 1%.

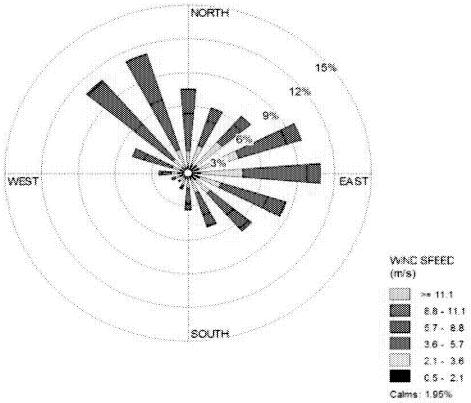


Fig. 1 wind rose for a period of year 2008

A model run is performed for actual monthly emission variation with total SO_2 emission rate of 6089.2 g/s and PM emission rate of 302 g/s. Monthly emission factors for SO_2 is tabulated in Table 2 and Monthly emission factors for PM is tabulated in Table 3. A discrete receptor is selected at Kuwait Environmental Public Authority monitoring station located at polyclinic in Fahaheel area. Concentrations of SO_2, NO_x, H_2S, O_3, CO, CO_2, methane, non-methane hydrocarbon, Benzene, Toluene, Xylenes, ethylbenzene, total suspended particulates and meteorological parameters are continuously recorded on hourly basis.

Table 2 SO_2 monthly emission factors

January	February	March	April	May	June
0.077	0.083	0.096	0.1	0.077	0.088
July	**August**	**September**	**October**	**November**	**December**
0.067	0.067	0.088	0.077	0.1	0.75

Table 3 PM monthly emission factors

January	February	March	April	May	June
0.093	0.097	0.091	0.079	0.079	0.083
July	**August**	**September**	**October**	**November**	**December**
0.064	0.063	0.085	0.079	0.079	0.1

Hourly predicted ground level concentrations at specified discrete receptor showed large scatter due to variation in meteorological conditions and the recorded values influenced by the contribution of various emission sources has made the comparison impracticable. Therefore, daily average measured concentrations of SO_2 were compared with the daily-predicted concentrations to validate the model output.
Figure 2 shows the plot between the measured top 20 daily average values versus the daily predicted top 20 values at the discrete receptor, Kuwait-EPA monitoring station.

The slope is equal to 0.72, reflecting high measured values compared to predicted values, depicting the contribution of other emission sources. The correlation coefficient is equal to 0.91 reflecting an acceptable validation of the model output with measured average daily SO_2 concentrations.

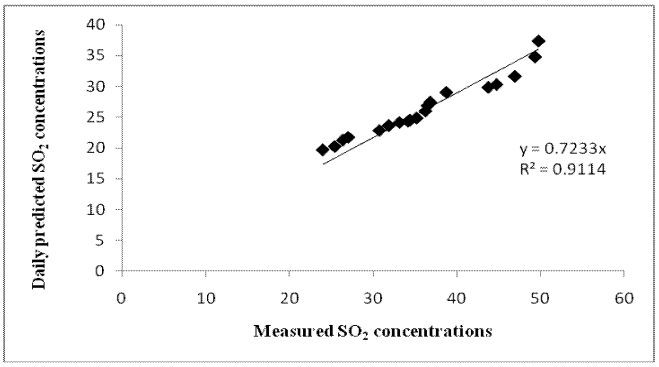


Fig. 2 Daily predicted SO_2 concentrations vs. measured SO_2 concentrations

The predicted hourly average ground level concentrations of SO_2 are compared with Kuwait-EPA Ambient Air Quality Standards (AAQS) at all of the selected receptors.
The maximum allowable level for the hourly average concentration of SO_2, specified by Kuwait-EPA, is 444 $\mu g/m^3$. Fig. 3 shows the isopleths of the predicted hourly average ground level concentration of SO_2 calculated at the selected uniform grid receptors.

Fig. 3 Isopleths plot of the predicted hourly average ground level concentration of SO_2

The isopleths indicate the predicted spatial variations of the ground level concentrations of SO_2. The maximum predicted hourly average ground level concentration of SO_2 in the vicinity of the refinery exceeded by as much as 300 $\mu g/m^3$. The highest predicted concentration is equal to 769 $\mu g/m^3$, observed on the 8th of March 2008 at 8:00 hour and about 1.713 km in the NW direction from the FCC stack, and not far from the Fahaheel and Ahmadi areas at the receptor coordinates of X = 218557.94, Y = 3219169. This high value of the predicted SO_2 concentration is expected due to the elevated SO_2 emission rate, which resulted from the high sulphur content in the FCC feedstock and other operational conditions and the prevailing meteorological conditions (temperature, humidity, wind speed, wind direction, stability class and planetary boundary layer).
A thorough inspection on fig. 3 indicates that predicted concentrations of SO_2 exceed the allowable hourly limit at 5.3 % of the study area from North West and South West direction from the stack.
Similarly, the predicted daily average ground level concentration of SO_2 is compared with Kuwait EPA ambient air quality standards at all receptors. The allowable level for the daily average concentration of SO_2 is 157 $\mu g/m^3$. Fig. 4 shows the isopleths of the predicted daily average ground level concentration of SO_2 computed at the selected uniform grid receptors.

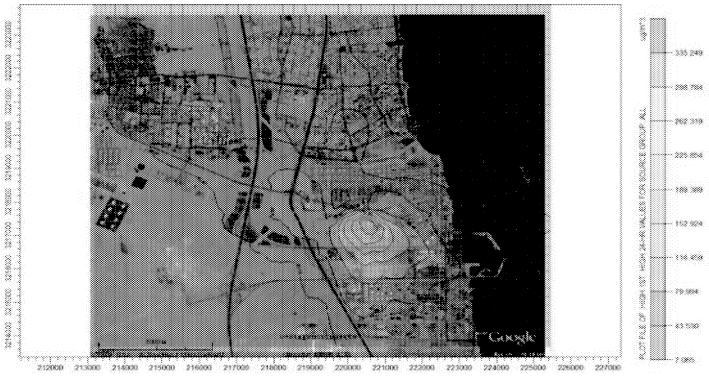


Fig. 4 Isopleths plot of the predicted daily average ground level concentration of SO_2

The isopleths indicate the daily predicted spatial variations of the ground level concentrations of SO_2 in the area of study. The highest daily predicted concentration is equal to 335$\mu g/m^3$, observed on the 9th of November 2008 and about 0.75 km in the SE direction from the stack, at a receptor coordinates of X = 220357.94, Y = 3217419 affecting the neighbouring Shuaiba industrial area, Kuwait main industrial complex. This high value of the daily predicted SO_2 concentration is exceeded the allowable level by 157 $\mu g/m^3$ and obviously influenced by the prevailing meteorological conditions, especially the predominant North West wind and other meteorological factors.
Discrete receptor 2, is located at Petroleum services offices, has shown the highest SO_2 hourly concentration equal to 544$\mu g/m^3$ on 27th February at 8:00 hours. The hourly concentration level rise beyond acceptable peak is occurred four times at this location throughout the study period. The highest daily concentration at the same receptor is equal to 39$\mu g/m^3$ on 8th March.
Discrete receptor 3 shows the highest SO_2 hourly concentration equal to 279$\mu g/m^3$ on 2nd March at 4:00 hours. This concentration is below the Kuwait EPA hourly standards. The daily highest concentration is equal to 57$\mu g/m^3$ on 2nd March. Discrete receptor 4, is located at Ahmadi hospital, has shown the highest SO_2 hourly ground level concentration equal to 288$\mu g/m^3$ on 27th February at 8:00 hours. This value is also below the specified hourly limit set by Kuwait EPA. The daily predicted concentration is equal to 23$\mu g/m^3$ on 30th April. Discrete receptor 5, is located at shopping area, has shown the highest SO_2 hourly ground level concentration is equal to 336$\mu g/m^3$ on 23rd October at 8:00 hours. The daily predicted concentration is equal to 45$\mu g/m^3$ on 22nd April. Both hourly and daily predicted values are below Kuwait EPA hourly and daily ambient air quality standards.

1. *Model Sensitivity*

To observe the computational model sensitivity, another run is performed using two finer meshes consisting of 21 x 21 uniform receptor points, the first covering hourly highest ground level concentration area, the second covering daily highest predicted ground level concentration area. The output accuracy has improved for both pollutants due to application of interpolation using small values of Δx = 150 m, Δy = 110 m for the first mesh and Δx = 100 m, Δy = 100m for the second mesh. There is 0.65% increase in the hourly highest ground level concentration and 2.8% increase in the daily highest ground level concentration, which are insignificant.

2. *Parametric Study*

FCC stack sensitivity analysis is performed on 3 scenarios (stack height, SO_2 emission rate and stack diameter). In scenario 1, analysis for stack heights 50 m, 80 m, 120 m, 160 m and 200 m is conducted while keeping the emission rate, exit flue gas velocity, exit temperature and stack diameter constant. The influence of stack height is shown in fig. 5. It is obvious from the figure that the highest predicted hourly and daily ground level concentrations of SO_2 are reduced substantially as stack height is increased. The reduction in the highest computed hourly ground level concentration of SO_2 is almost 50% when stack height is doubled. The decrease in evaluated hourly SO_2 concentration as a function of stack height is given as an exponential expression $C(\mu g/m^3) = 1600.7e^{-9.071x10^{-3}h}$ and r^2 is 0.999, where h is the stack height (m). The hourly gradient dC/dh $= 14.52e^{-9.071x10^{-3}h}$ becomes insignificant at higher stack elevations. The highest daily predicted ground level concentration as a function of stack height is given as $C(\mu g/m^3) = 1409.8e^{-1.732x10^{-2}h}$ and r^2 is 0.984. The daily highest predicted concentration gradient is $dC/dh = 24.42e^{-1.732x10^{-2}h}$. The locations of hourly highest predicted concentrations of SO_2 from the stack, as a function of stack height is shown in figure 7 and related as $D(km) = 0.597e^{1.16x10^{-2}h}$ and r^2 is 0.9.

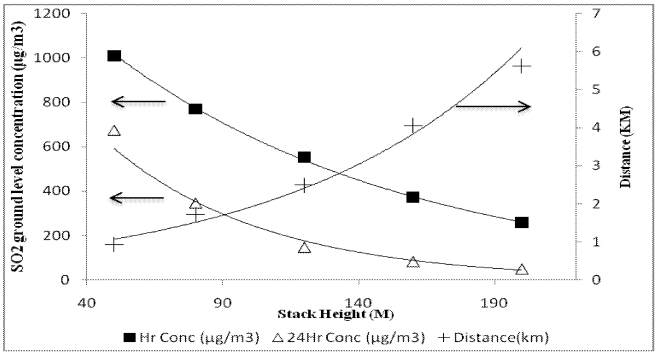


Fig. 5 Stack height vs. hourly and daily predicted ground level concentrations of SO_2

In scenario 2, SO_2 emission rate effect from FCC stack is tested at stack height of 80 m for different total monthly emission rates of 3000 g/s, 4000 g/s, 5000 g/s, 6000 g/s, 7000 g/s and 8000 g/s, taking into consideration the monthly emission variations (by using emission factors, table 2) and fixing other stack parameters i.e. exit temperature, exit flue gas velocity and stack diameter.

It is noticed from fig. 8 that the highest predicted hourly and daily ground level concentrations of SO_2 is substantially decreased as SO_2 emission rate is reduced. At 50% reduction in the emission rate, the highest hourly and daily ground level concentrations decreased by 50%.

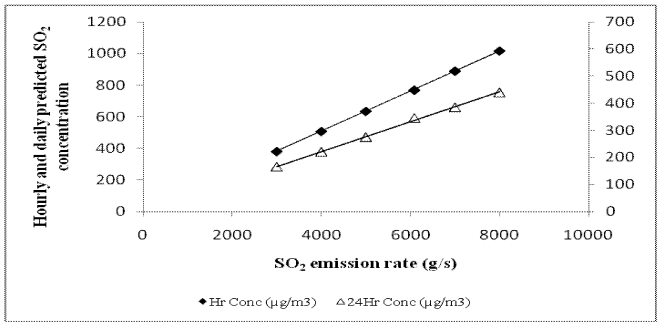


Fig. 6 SO_2 emission rate vs. hourly and daily predicted SO_2 ground level concentrations

In scenario 3, FCC stack diameter effect is examined at stack height of 80 m for different diameters of 1.5 m, 2.3 m, 3 m and 4 m. The exit flue gas velocity is also changed as directly related to the square of the diameter for a fixed exit flue gas flow rate. It is observed that the dispersion and rise of the plume are not affected by diameter variation and the predicted ground level concentration of SO_2 remained almost unaltered. The hourly and daily predicted concentrations of SO_2 are almost identical for all the cases.

Kulkarni et al., (2009) have reported that Lanthanum and Lanthanides are used as markers for particulate matters pollution as $PM_{2.5}$ in petroleum refineries, mainly from FCC units. US EPA daily $PM_{2.5}$ standard is 35µg/m^3. In the present work, the application of Aermod to predict ground level concentration of PM is considered as $PM_{2.5}$ for rare earth elements i.e. Lanthanum and Cerium. $PM_{2.5}$ is inhalable and has adverse impact on public health causing cardiovascular diseases. Kuwait EPA has no standard for $PM_{2.5}$ and has only specified daily and yearly standard for PM_{10}. Figure 5 shows the isopleths of the predicted hourly average ground level concentration of PM calculated at the selected uniform grid receptors.

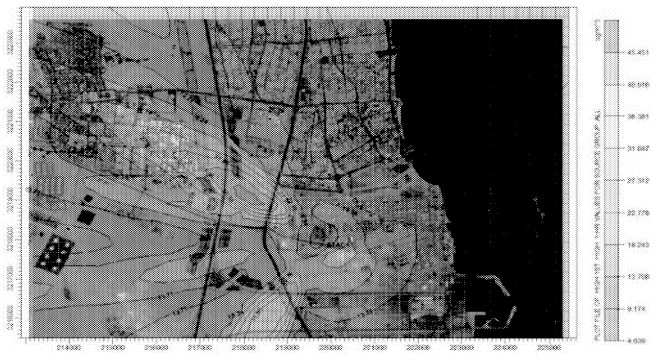


Fig. 7 Isopleths plot of the predicted hourly average ground level concentration of PM

The isopleths indicate the hourly predicted spatial variations of the ground level concentrations of PM. The maximum hourly predicted average ground level

concentration of PM is equal to 45μg/m^3, observed on the 27th of February 2008 at 8:00 hour and about 1.56 km in the NW direction from the FCC stack, and at receptor coordinates of X = 218557.94, Y = 3218919.

Similarly, the predicted daily average ground level concentration of PM is compared with US EPA ambient air quality standards for $PM_{2.5}$ at all receptors. Figure 6 shows the isopleths of the predicted daily average ground level concentration of PM computed at the selected uniform grid receptors.

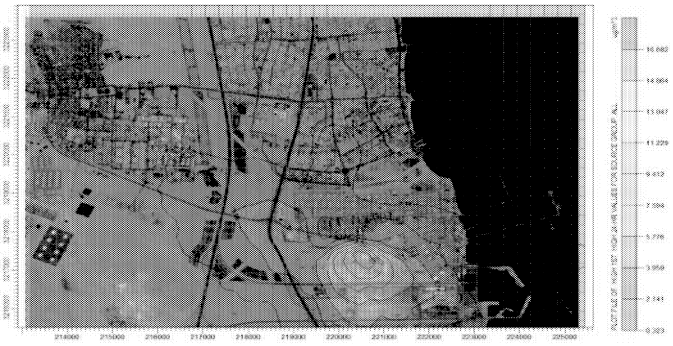


Fig. 8 Isopleths plot of the predicted daily average ground level concentration of PM

The isopleths indicate the daily average predicted spatial variations of the ground level concentrations of PM in the area of study. The highest daily predicted concentration is equal to 16μg/m^3, observed on the 29th of December 2008 and about 0.75 km in the SE direction from the stack, at a receptor coordinates of X = 220657.94, Y = 3217419 due to the influence of the prevailing meteorological conditions, especially the predominant North West wind and other meteorological factors.

To observe the computational model sensitivity, another scenario run is performed adding two finer meshes consisting of 21 x 21 uniform receptor points, the first one covering hourly highest ground level concentration area, the other one covering daily highest predicted ground level concentration area. The output accuracy has improved for both pollutants due to application of interpolation using small values of Δx = 150 m, Δy = 110 m for the first mesh and Δx = 100 m, Δy = 100m for the second mesh. There is 0.65% increase in the hourly highest ground level concentration and 2.8% increase in the daily highest ground level concentration, which are insignificant.

CONCLUSIONS

FCC unit in a refinery is a major contributor to SO_2 and PM emissions. These gases have adverse impact on the immediate neighbourhood of refineries. In this study a complete emission inventory for a year long period have been prepared for SO_2 and PM. A model run performed for actual monthly emission variation with total SO_2 emission rate of 6089.2 g/s and PM emission rate of 302 g/s, taking into consideration monthly emission factors for both SO_2 and PM.

The daily predicted ground level concentrations of SO_2 are compared with Kuwait EPA monitoring station daily measured SO_2 concentrations at the same discrete receptor and showed acceptable validation of the model output.

The highest hourly predicted concentration of SO_2 is equal to 769 μg/m^3. It is observed on the 8th of March 2008 at 8:00 hour, due to elevated SO_2 emission rate in this month and the prevailing meteorological conditions, especially sea breeze effect in the early morning hours. The highest daily predicted concentration is equal to 335μg/m^3. It is observed on the 9th of November 2008, and obviously influenced by the predominant North West wind and high SO_2 emission rate in the month of November.

The maximum hourly predicted average ground level concentration of PM is equal to 45μg/m^3. It is observed on the 27th of February 2008 at 8:00 hour. The highest daily predicted concentration is equal to 16μg/m^3, observed on the 29th of December 2008.

The stack sensitivity is explored by changing stack height, total emission rate and stack diameter independently. It is observed that the higher stack facilitated good dispersion, thus lowering the ground level average concentration of the pollutant up to 50% when the stack height doubled.

It is notice that the highest predicted hourly and daily ground level concentrations of SO_2 are substantially decreased as SO_2 emission rate is reduced. At 50% reduction in the emission rate, the highest hourly and daily ground level concentrations decreased by almost 48%.

The influence of stack diameter inherently changed the exit flue gas velocity due to invariable flue gas flow-rate. The plume rise and dispersion are related to the exit flue gas velocity, which decreased with the increase of stack diameter because of proportionality to the square of diameter. For a fixed load there is no noticeable change in the average hourly and daily predicted ground level concentrations of SO_2. The study results presented in this paper provide, for the first time, a comprehensive quantitative analysis of the impact of a typical FCC unit on its surrounding environment.

References

Abdul Wahab S. A., Al-Alawi S.M., El-Zawahri A. (2002), "Patterns of SO_2 emissions: a refinery case study" Environmental Modeling and Software 17 563-570

Caputo M., Gimenez M., SchlampM. (2003), "Inter-comparison of atmospheric dispersion models" Atmospheric Environment 37, 2435–2449

Isakov V., Venkatram A, Touma S. J, Koracin.D,Otte L. T. (2007) "Evaluating the use of outputs from comprehensive meteorological models in air quality modeling applications" Atmospheric Environment 41, 1689–1705

Kesarkar A. P., Dalvi M., Kaginalkar A., Ojha A. (2007) "Coupling of the Weather Research and Forecasting Model with AERMOD for pollutant dispersion modeling. A case study for PM_{10} dispersion over Pune, India" Atmospheric Environment 41, 1976–1988

Kulkarni P., Chellam S., Fraser M. P. (2009) "Tracking Petroleum Refinery Emission Events Using Lanthanum and Lanthanides as Elemental Markers for $PM_{2.5}$" Environmental Science and Technology, **43** (8), 2990–2991, 2009.

Yateem W., Nassehi V., Khan A. R., (2010) "Inventories of SO_2 and PM emissions from Fluid Catalytic Cracking (FCC) units in petroleum refineries", Water, Air & Soil pollution (to be published).

ENVIRONMENTAL SIMULATION

PREDICTIVE CONTROL FOR THERMAL COMFORT OPTIMIZATION AND ENERGY SAVING

Mariusz Nowak
Andrzej Urbaniak

Institute of Computing Science
Poznan University of Technology
Piotrowo 2, 60-965 Poznan, Poland
E-mail: {Mariusz.Nowak|Andrzej.Urbaniak}@put.poznan.pl

KEYWORDS: thermal comfort, intelligent control, model predictive control, energy saving, HVAC system

ABSTRACT

In the paper there are presented the problems of optimal control for indoor thermal comfort in buildings equipped with HVAC (Heating, Ventilation and Air Conditioning) systems. It is important that the design of such control systems allows to minimize the energy consumption. There are discussed the results of air-conditioning system simulation and there is given an example of simulation model of a room and mathematical model of thermal comfort. Moreover, in the paper there is presented the hierarchical structure of climate comfort control system. Fuzzy control algorithms are used in direct control layer (Nowak and Urbaniak 2007), while for supervisory control layer there are presented model predictive control algorithms. The tuning of predictive controllers is discussed in the paper, too.

INTRODUCTION

Nowadays, an important issue in the residential sector is to reduce and to optimize the consumption of energy. Very important task is the implementation of a monitoring system that shows the electrical energy consumption of air conditioning systems in a building. At the same time, thermal comfort has a great impact on the productivity and satisfaction of the indoor building's occupants. The interactions between people and thermal environment are of a very complex nature and have been the subject of many studies (ASHRAE 2003, Fanger 1973, Fanger 1982, Jones 2001).

In usual, simple HVAC systems for thermal comfort in intelligent buildings are based either on a single temperature control loop or temperature and humidity control loops together. However, as far as thermal comfort optimization is concerned, other parameters should also be considered in order to provide thermal satisfaction to inhabitants. Therefore, there is presented the application of fuzzy logic and predictive control for thermal comfort condition.

The control algorithms of such complex systems are very sophisticated ones. There is a new approach presented in the paper, in which the hierarchical structure of control is utilized. In previous articles (see Nowak and Urbaniak 2005, Nowak and Urbaniak 2007, Nowak 2008) the detailed development of fuzzy controllers were described. However, in this paper main focus is put on supervisory control layer, in which the predictive control algorithms are implemented.

As mentioned above, the crucial part of energy consumption in a building is connected with heating and cooling devices. For this reason, the first step would be the preparation of mathematical model of a room. Then, the next step shows the realisation of a thermal comfort model of this room. The thermal comfort model can be used for the study of HVAC systems. The model is also useful for the studies of control strategies (fuzzy, model predictive control) as well as for finding the solutions for reducing the electrical energy consumption and for maintaining acceptable indoor air conditions related to thermal comfort. Thus, in the paper there are suggested control strategies for reducing energy consumption and maintaining these acceptable indoor air conditions related to thermal comfort.

THERMAL MODEL OF A ROOM AND THERMAL COMFORT MODEL

In this paper there is presented the simulation model of a room in a building prepared in Matlab/Simulink software (Fig. 1). The model includes physical parameters and parameters of the construction of walls, floor and roof. The mathematical simulation model of a room requires the definitions on the room geometry, specifies thermal properties of the room materials, thermal resistance of the room, heater characteristics (temperature of hot air, flow-rate), air-conditioning characteristics and initial room temperature. The room's model is presented in Matlab/Simulink as a subsystem that calculates room's temperature variations. It takes into consideration heat flow from a heater, cold flow from an air-conditioning system and heat losses to the environment. Heat losses and temperature time derivative are expressed by special equation (Nowak and Urbaniak 2005).

Thermal comfort in buildings is a concept that is difficult to define. As thermal satisfaction depends on several parameters, research works on thermal comfort have been conducted and some comfort indices have been proposed over the last several years. A large number of thermal comfort indices have been established for indoor climate analysis and HVAC control systems' design; among them a quite disseminated one is the PMV (Predicted Mean Vote). The PMV index can be written as a function of four environmental variables: air temperature, humidity, air velocity and mean radiant temperature; and two individual

parameters: clothing insulation and human activity. The other index is PPD (Predicted Percentage of Dissatisfied), obtained from the PMV one. PPD index provides information on thermal discomfort by predicting the percentage of satisfied or dissatisfied people. In addition, to predict the percentage of dissatisfied people due to a draught, European law EN ISO 7730 introduces the DR (Draught Rating) index (EN ISO 7730:1994). In together, these three indexes can be used to control thermal comfort in a room.

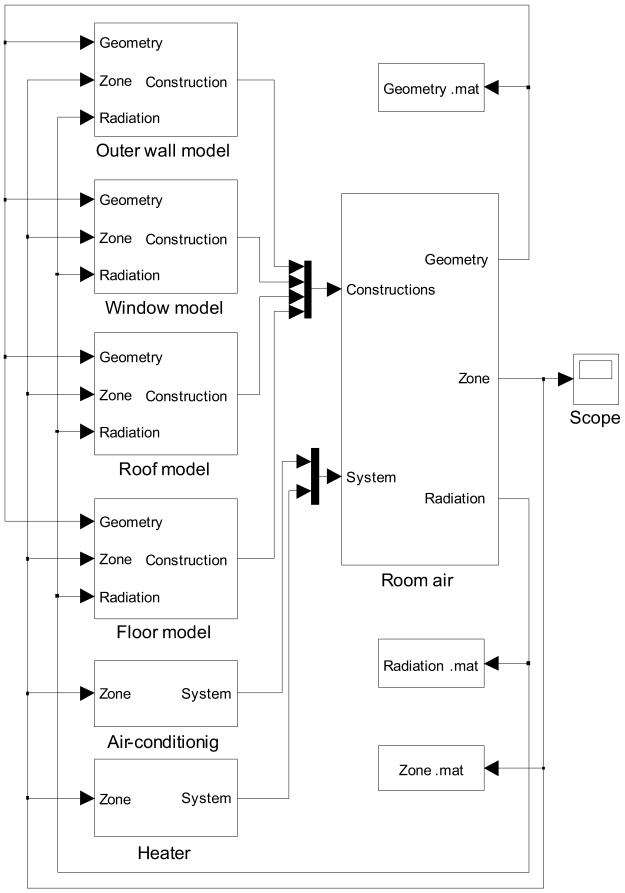


Figure 1: Model of a room in Matlab/Simulink

HIERARCHICAL STRUCTURE OF INTELLIGENT THERMAL CONTROL

An approach for control of thermal comfort has been presented by Fanger (Fanger 1973, Fanger 1982). The main goal of this control is to obtain such values of microclimate parameters that allow to achieve the demand level of general PMV index. In usual, this demand is formulated respect to optimum energy consumption. These two general goals: PMV achievement (in an expected period) and minimum energy consumption are the conflicting ones. Thus, there is an option to formulate the optimization problem with two conflicting criteria and to search eventually for the compromise solution.

For control processes it is necessary to formulate the set of partial goals which very often later guarantees security controlled processes. Fulfilling these partial goals is a condition for searching the best compromise solution. Taking into account the remarks mentioned above there is a natural suggestion towards using the control approach with hierarchical structure of control system. In this approach have been defined four layers of hierarchy: direct control layer, supervisory control layer, optimization layer and planning layer (Fig. 2). First layer named direct control is responsible for security controlled process. Control algorithms defined in this level must be safe and simple. The direct access to controlled process is only achievable from this layer, excluding all higher layers. Here the classic PID and fuzzy logic algorithms are used quite often.

In the second layer fixed points for direct control algorithms are calculated together with other suggested algorithms – the predictive ones (for example: MPC). The algorithms realized in this layer need the process models which have been already controlled.

The achievement of climate comfort defined by three general indexes: PMV, PPD and DR is realized in an optimization layer. The results obtained in this layer allow to calculate the important fixed points for supervisory control level.

The highest layer gives the possibility to find the compromise solution respect to general conflicting criteria: climate comfort maximization vs. energy consumption minimization.

The proposed approach allows better definition of the control goals, helps simplifying design process and gives the possibility of sequential preparation of the project – in a step by step way.

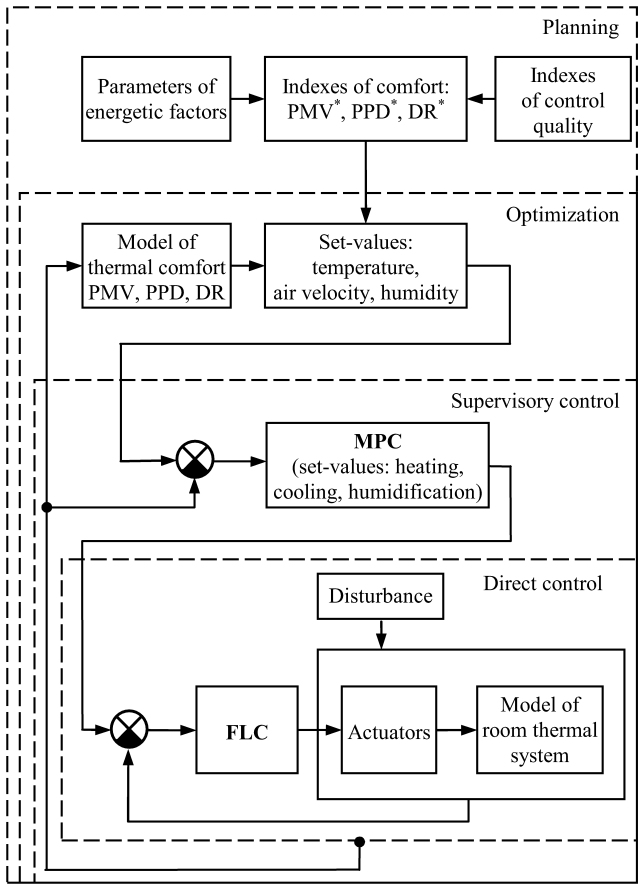


Figure 2: Hierarchical structure of control of thermal comfort

ALGORITHMS OF CLIMATIC COMFORT CONTROL

In the direct control layer fuzzy logic controllers have been used. The analysis of their implementation was presented in previous papers (Nowak and Urbaniak 2007, Nowak 2008).

Now let us pay attention to the supervisory control layer with predictive control algorithms. The methodology is related to controllers that present almost the same structure and characteristics as in the classic solution. Thus, MPC algorithms are defined by the process model related to control purposes. The design of the control system is characterized by four main steps (Freire at al. 2008, Nowak 2008):

- process modelling: The data from input (manipulated) and output (controlled) signals are used to predict the process behaviour (output prediction) in a future horizon, defined as a prediction horizon (N_y).
- cost function definition: The system closed-loop performance during the prediction horizon is specified. It is defined by using the output prediction, the reference signal and the control effort.
- cost function optimization: The cost function is optimized as a function of the set of future control signals (within control horizon - *Nu*) to be applied to the process during the prediction horizon. In this step, constraints for the manipulated and controlled variables can be added in order to deal with the system operation constraints, *e.g.*, limits on the actuators of the HVAC system.
- redefined strategy horizon: Only the first control signal computed from the cost function optimization is applied to the real process and, in the next step time, the whole algorithm is repeated.

PMV values can be calculated by using a mathematical model of thermal comfort. A control strategy is described where a building's occupants' thermal comfort sensation is given by PMV calculations. The PMV predictions computed by using the nonlinear model are included in the cost function. The closer to zero of the PMV value, the better thermal sensation is achieved. Therefore, the control rule is given by the following general optimization problem (1):

$$\min J(i) = \sum_{i=1}^{N_y} \left(x_{PMV}(i) - y_{PMV}(i) \right)^2 + \lambda \sum_{i=0}^{N_u - 1} \Delta u(i)^2 \qquad (1)$$

where: $x_{PMV}(i)$ – i-th PMV index reference, $y_{PMV}(i)$ – i-th PMV index measured output, $\Delta u(i)$ – i-th manipulated variable change, λ – weighting coefficient penalizing relative big changes in Δu, N_y – prediction horizon, N_u – control horizon.

In this paper there is described the consideration of the constraint over the control signal, imposed by the HVAC device.

There has been undertaken experimental research using predictive algorithms DMC (Dynamic Matrix Control) and GPC (Generalized Predictive Control), to be described in the next sections.

Dynamic Matrix Control algorithm

In the DMC algorithm control object's dynamics is modelled by discrete step response of an object $\{0,s_1,s_2,s_3,...,s_n...\}$ inducted by unit step function Δu, where $s_1,s_2,s_3,...$are step responses' values obtained during successive sampling moments. The step response function has been obtained by using software simulation method. On the base of the already known discrete step response there is possibility to model the object's discrete response for any discrete control signal. It is because any discrete signal can be treated as a sum of step signals which start together with the consecutive sampling moments with different amplitudes. Basing on a superposition rule it can described (Tatjewski 2002):

$$\begin{aligned} y(1) &= y(0) + s_1 \Delta u(0) \\ y(2) &= y(0) + s_2 \Delta u(0) + s_1 \Delta u(1) \\ y(3) &= y(0) + s_3 \Delta u(0) + s_2 \Delta u(1) + s_1 \Delta u(2) \\ &\dots \end{aligned} \qquad (2)$$

Hence, for any k = 1,2,3,… the following formula is obtained:

$$y(k) = y(0) + \sum_{j=1}^{k} s_j \Delta u(k-j) \qquad (3)$$

Value of the control signal for k+p sampling moment, calculated from the formula (3), can be formed as:

$$y(k+p) = y(0) + \sum_{j=1}^{k+p} s_j \Delta u(k+p-j) \qquad (4)$$

The formula (4) is necessary for calculating prediction equations in the DMC algorithm. In general, in the DMC algorithm there is unchangeable value of disturbance signal assumed (Tatjewski 2002). For simplification reason, in the presented experimental results the disturbance signal is omitted. The predictive value of control signal in a current sampling moment k on a sampling moment k+p is denoted as y(k+p|k); and increment value of control signal for sampling moment k on future moments is denoted as $\Delta u(k+p|k)$, respectively. Thus, the value of control signal y(k+p|k) is obtained from the formula (5):

$$\begin{aligned} y(k+p \mid k) &= y(0) + \sum_{j=1}^{p} s_j \Delta u(k+p-j \mid k) + \\ &+ \sum_{j=p+1}^{k+p} s_j \Delta u(k+p-j) = \\ &= \Delta y(k+p \mid k) + y^0(k+p \mid k) \end{aligned} \qquad (5)$$

where $\Delta y(k+p|k)$ is enforced component of predicted control signal' trajectory. This component depends on increment values of control signals in current sampling moment k and future ones. However, $y^0(k+p|k)$ is a free component depending only on previous values of control signal's increment. In the analytic version of an algorithm the vectors x(k), $y^0(k)$, $\Delta y(k)$, $\Delta u(k)$ are defined; thus the talked over control criterion function has a following shape (6):

$$J(k) = \left\| [x(k) - y^0(k)] - \Delta y(k) \right\|^2 + \lambda \left\| \Delta u(k) \right\|^2 \qquad (6)$$

The vectors $y^0(k)$, $\Delta y(k)$ and y(k) are calculated on the base of process' model in the shape of finite step response. In the experiments there has been assumed one-dimensional control object, thus the free component of predicted trajectory is formed as (7):

$$y^0(k) = y(k) + M^P \Delta u^P(k) \quad (7)$$

where M^P is the proper dynamic matrix and its elements are differences' values between relevant values of s coefficients. The number of columns and rows in the matrix depends on the given prediction horizon. The high index p denotes that the matrix describes predicted control signals basing on previous increments of control signals. The Δu^P is a proper vector of control signal increment in consecutive sampling moments. What is more, the enforced component has a following form:

$$\Delta y(k) = M \Delta u(k) \quad (8)$$

where M is the dynamic matrix with its elements, which are values of respective s coefficients. The number of columns and rows in the matrix is determined by initial moment of calculating the criterion function and by prediction horizon value.

The simulation research has been conducted for different values of coefficient and different length of prediction and control horizons. The chosen diagrams with the results are presented in section 5.

Generalized Predictive Control algorithm

The object's model in the GPC algorithm is presented in a form of a discrete differential equation (Tatjewski 2002). Unlike the DMC algorithm, there is used the reference trajectory in the criterion function instead of the reference values' trajectory. The formula (9) describes the dependences between these signals:

$$x^{ref}(k+p \mid k) = \gamma \cdot x^{ref}(k+p-1 \mid k) + (1-\gamma) \cdot x(k+p \mid k) \quad (9)$$

where x^{ref} is the reference trajectory, x denotes the trajectory of reference values and $0 \le \gamma \le 1$. The value of γ parameter determines draw velocity of reference trajectory x^{ref} to x trajectory. In the simulation experiments there have been examined the influence of γ coefficient value's increase on drawing up the reference trajectory to reference values trajectory; and also, how γ coefficient value's increase effects the control signal.

While creating the GPC algorithm, a control object model in the form of differential equation, has been used:

$$A(z^{-1}) \cdot y(k) = B(z^{-1}) \cdot u(k-1) + \frac{v(k)}{\Delta} \quad (10)$$

Where $A(z^{-1})$ and $B(z^{-1})$ are the polynomials of variable z^{-1}:

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + ... + a_{n_A} z^{-n_A}$$
$$B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2} + ... + b_{n_B} z^{-n_B} \quad (11)$$

In formula (11) z^{-1} denotes an operator of entity delay, v(k) is a vector of white noise with zero average value and $\Delta = 1 - z^{-1}$ denotes an operator of backward difference.

SIMULATION RESULTS

The mathematical model of thermal comfort has been implemented in Matlab/Simulink together with the use of intelligent algorithms in simulation investigations: fuzzy algorithms in the direct control layer, predicted algorithms in the supervisory control layer. The main task of the proposed hierarchical control system is to provide thermal comfort and minimize energy consumption. Simulation studies have been conducted by taking into account two conflicting criteria: optimization of the PMV index value, and energy saving.

The problem of optimizing the PMV index values was formulated as a minimization of objective function given by the equation (1). The problem of energy saving was analyzed by assuming that the PMV index value must be included in the limit: -0,5 < PMV < +0,5.

In the paper there are shown the results of computer simulations, in which the length of the predictions' horizon has been changed. These results have been obtained using DMC (Dynamic Matrix Control) and GPC (Generalized Predictive Control) algorithms.

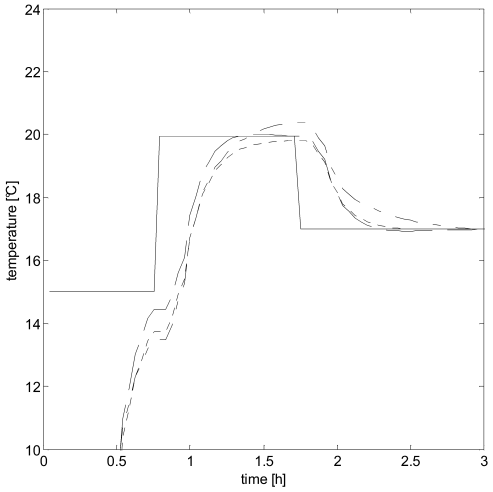


Figure 3: Changes of room temperature with the changes the prediction horizon (DMC algorithm).

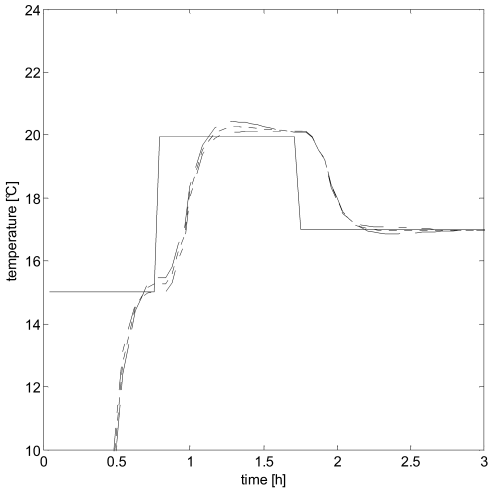


Figure 4: Changes of room temperature with the changes the control horizon (DMC algorithm).

There have been presented the curves of temperature change in a room using DMC algorithm for different prediction and control horizons in Fig. 3 and Fig. 4, respectively. The continuous line describes reference temperature value; the dashed and point lines describe control values.

The increase of prediction and control horizon caused decrease of error between reference value and control signal. On the Fig. 5 there is compared the operation of the two analysed algorithms with optimal values of prediction and control horizons (continuous line – reference value, dashed line – control value for DMC algorithm, pointed line – control value for GPC algorithm). On the Fig. 6 there is shown the curve of PMV index change during 24 hours for an office-room (continuous line – for the priority demand to aspire the PMV index value to zero; dashed line – with additional restriction of energy consuming).

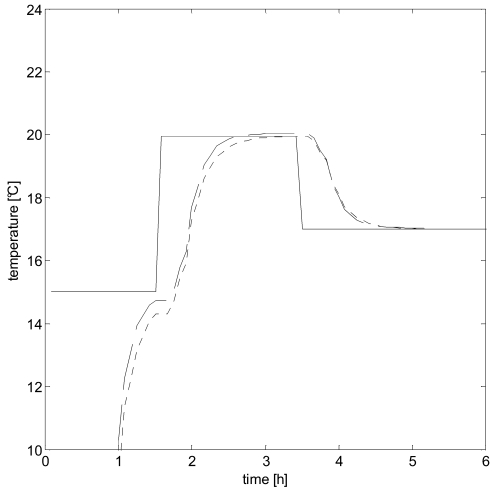


Fig 5. Comparison of optimal DMC and optimal GPC algorithms.

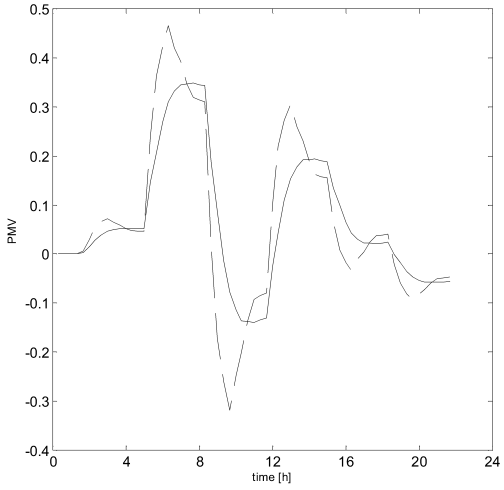


Fig 6. Changes PMV index.

CONCLUSIONS

In the paper there are compared several approaches for thermal comfort optimization with a use of simulation tools. The classic approach to the control was expanded in the research concerning minimization of power consumption with maintaining thermal comfort index level within the specified limits. The results of computer simulations presented in the paper shows that the compromise between two conflicting goals is possible thanks to hierarchical structure of control allowing the implementation of intelligent control algorithms.

The experiments have shown very small differences in effectiveness between DMC and GPC algorithms applied as tools assuring microclimate comfort in controlled objects.

REFERENCES

EN ISO 7730:1994, "Moderate thermal environments – Determination of the PMV and PPD indices and specification of the conditions for thermal comfort"

ASHRAE 2003, "Handbook of HVAC Applications", American Society of Heating, Refrigerating and Air Conditioning Engineers, Atlanta, USA, 2003

Fanger, P.O., 1973, "Thermal Comfort", McGraw-Hill, New York, USA 1973

Fanger, P. O., 1982, "Thermal Comfort", Krieger Publ. House, Malanbar 1982

Freire, R. Z., Oliveira, G. H. C. and Mendes N., 2008, "Predictive controllers for thermal comfort optimization and energy saving". In B. Todorovic editors, *Energy and Buildings* 40, pages 1353-1365, Elsevier 2008

Jones, W. P., 2001, "Air Conditioning", Wydawnictwo Arkady, Warszawa 2001 (in polish)

Nowak, M. and Urbaniak, A., 2005, "Computer modelling and simulation of the influence of physical parameters on thermal comfort". In Manuel Feliz Teixeira, J., Carvalho-Brito, A. E., editors, *Proc. of The 2005 European Simulation and Modelling Conference EUROSIS-ETI*, pages 445-447, Porto-Portugal, 2005

Nowak, M., Urbaniak A., 2007, "Simulation of integrated control system for climate control", In Sklenar J., Tanguy A., Bertelle C and Fortino G. editors, *Proc. Of The 2007 European Simulation and Modelling Conference EUROSIS-ETI*, pages 606-610, St.Julians-Malta, 2007

Nowak M., 2008, "Operating cost optimization of air conditioning systems in the buildings using an intelligent algorithms", In Kapron H. editors, *Rynek Energii*, nr 3(76) June 2008, pages 48-53, Lublin-Poland, 2008, (in polish)

Tatjewski P., 2002, "Advanced Process Control – structures and algorithms", Akademicka Oficyna Wydawnicza EXIT, Warszawa 2002 (in polish).

BIOGRAPHIES

Mariusz NOWAK was born in Poland and went to the Poznan University of Technology (control engineering). He obtained the PhD degree in 2007. From September 2007 he is an Assistant Professor at the Institute of Computing Science of the Poznan University of Technology. He research interest include: computer simulation, intelligent control system, computer control systems for environmental engineering, intelligent building systems, comfort climate control.

Andrzej URBANIAK was born in Poland and went to the Poznan University of Technology (control engineering) and Poznan University of A. Mickiewicz (mathematics). He obtained the PhD degree in 1979. From 1990 he is a professor of Institute of Computing Science. He is author or co-author of 5 books and over 200 papers concerning the computer control systems and application of computer science in environmental engineering.

SIMULATION AS A TOOL FOR THE EVALUATION OF FOREST MANAGEMENT TREATMENTS

Ulla Ahonen-Jonnarth and Jan Odelstad
Department of Industrial Development, IT and Land Management
Faculty of Engineering and Sustainable Development
University of Gävle
Sweden
E-mail: uah@hig.se, jod@hig.se

KEYWORDS
On-line algorithm, forest cleaning.

ABSTRACT
Cleaning of young forest stands is a multicriteria problem with conflicting goals. This kind of forest management treatment is performed by human beings but it is possible that this work may be performed by artificial agents in the future. The artificial agents need detailed information about how to clean a forest stand and/or what are the goals for cleaning. One problem in development of cleaning rules for artificial cleaning agents is that explicit knowledge about good cleaning results is not detailed. In this paper we present a tool for developing and testing rules and judging evaluation functions for cleaning. We illustrate this tool by presenting examples of some ways to clean forest stands in a computer environment and we present how the cleaning results can be evaluated. In order to obtain material for experiments it is also possible to simulate forest stands using this tool.

INTRODUCTION

Cleaning, thinning and final harvesting are forest management treatments in commercial forest management. Cleaning is the first forest management treatment in a growing young forest. Cleaning is important because without performing it young trees grow too close to each other often resulting in decreased growth of the trees and possibly in a lower quality of the full-grown trees. There are differences between forest stands with regard to, for example, species composition, age of trees and soil conditions. The goals for cleaning depend on the final goals for the forest stand – what kind of end products the forest owner plans to gain after the final harvesting of the forest stand. It is thus necessary that a cleaning agent is able to perform cleaning in different ways, depending on the goals for the actual forest stand.

Sometimes it is difficult to find people who want to work as cleaners. A possible scenario is that in the future cleaning is performed by autonomous artificial agents. Another task more close at hand is the construction of an educational tool that can be used for introducing seasonal cleaners to their work. In both cases explicit instructions are needed about how cleaning should be done. In order to develop such instructions for an artificial agent or for an educational simulator it is important to define how a good cleaning result ought to look like. This knowledge is partly tacit knowledge that human beings obtain during working periods as cleaners. In this paper, we discuss artificial cleaning agents that work alone and thus constitute one-agent-systems. According to the classification in Russell & Norvig (2003, Ch. 2) the task environment for the cleaning agents is partially observable, deterministic, sequential and discrete. We presume in this paper that the cleaning agent works in an "on-line" fashion. The cleaning agent does not have complete information about the whole forest stand when it starts its work. It observes one small area at time and takes cleaning actions, if needed, by removing some of the young trees in this small area. Then it saves information of the cleaned areas and goes further.

A difficult and important question in artificial cleaning of young forest stands is what characterizes a good cleaning result. Attempts have been made in order to extract knowledge from professional cleaners (Vestlund 2004). This information has then been used by Vestlund et al. (2006) for the construction of cleaning rules for an artificial cleaner in a computer environment. For evaluation of cleaning results we have, in an earlier paper, presented a preliminary evaluation function for cleaning results (Ahonen-Jonnarth & Odelstad 2006). This evaluation function aggregates aspects for this multicriteria problem to a summary measure. This and other evaluation functions can be judged and further developed in the forest simulator presented in this paper. In the following sections we present the forest simulator that is a tool for developing cleaning rules and judging evaluation functions. Examples of cleaning and evaluation of cleaning results are also presented.

FOREST SIMULATOR

The forest simulator we have constructed is a tool for developing and testing forest management treatments in a computer environment. Using this tool, it is possible to test and develop both different ways of cleaning and different evaluation functions for forest management treatments. In the forest simulator a forest stand is represented as a grid where there are trees in some positions. Each tree is represented by several aspect values, for example values for position, diameter at breast height, sort and if the tree is damaged or not. Cleaning activity in the forest simulator is deterministic. Input is a forest stand before cleaning and output is a forest stand after cleaning. In both cases the information about the

stand (size) and the trees (individual variable values) are entered in a database. When a simulation of a cleaning of a forest stand is going to be performed in the forest simulator, a forest stand is first divided into smaller parts, known as cleaning units (squares or rectangles). A cleaning agent moves from one cleaning unit to another but the way it can move is regulated. In a cleaning unit the agent decides which trees to remove and which to save. The decision can be based on different principles. Both individual aspects and global stand aspects can be used as a ground for the decision. One possible direction for a cleaning agent is shown in figure 1. There are eight possible directions a cleaning agent can clean a forest stand. The cleaning agent can only move forward; it can not go back to a cleaning unit it already has visited. When the agent is on a cleaning unit it can remove trees on this actual cleaning unit. Because the agent can not go back to a cleaning unit it can not change its cleaning result by removing more trees on a cleaning unit. The cleaning agent does not in general have information about the whole forest stand that is going to be cleaned. The agent first enters the first cleaning unit and then acquires information about the trees and their qualities on this unit. When the cleaning agent moves to the next cleaning unit, it has knowledge about how the earlier visited cleaning units have been cleaned. The agent's knowledge about the forest site is increasing during the cleaning process.

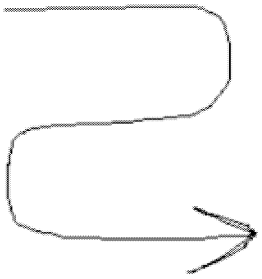


Figure 1: One possible direction for performing cleaning in a forest stand.

The forest simulator contains data about actual forest stands collected by forest researches. However, in order to test cleaning algorithms and evaluation functions, numerous forest stands to clean are needed. In order to fulfill this need it is possible to simulate new forest stands in the forest simulator. In that way it is possible to produce several replicates for different experiments. A replicate of a forest stand is a representation of a stand of the same kind but not identical as the forest stand that has been the model forest stand for simulation. The forest stand types differ from each other and so does the optimal way to clean different forest types. The forest stand replicates that have been simulated are of the same type as the forest stand that has been used as a model forest stand during the forest simulation. Probabilities that are used during simulation for different aspect values of trees (for example sort of tree and diameter) are based on the actual frequencies observed in the model forest stand. Coordinates for trees are chosen randomly but probability that a new tree is situated near an already placed tree is changed compared to the original probability. The amount of trees in a simulated stand is randomly chosen from an interval. A user can change default values for simulation parameters and thus affect variation in different aspects in the simulated forest stands.

RULES FOR FOREST CLEANING

For the agents that have been implemented so far, the agent architecture is goal based according to the classification in Russell & Norvig (2003, Ch. 2). In the construction of the agents, utility based ideas about good cleaning have been used even if it is not defined in detail what a good cleaning result is. One approach has been to implicitly include utility in the model and balance it with trade-off questions between different aspects because the goals for cleaning are often in conflict with each other. This approach is quite complicated. In order to decrease the complexity, another approach based on simple rules was used. The aim with the second approach is to find out if there are simple rules or rule combinations that give good or at least acceptable cleaning results but are easier for a human user to understand and regulate.

In the following, different cleaning approaches that have been implemented in the forest simulator are presented. These are 1) Ap algorithm, 2) variations of cleaning that are based on simple rules and 3) a corridor cleaning that represents one kind of geometric cleaning.

Ap – a pine forest algorithm

The algorithm Ap is based on the principles of Vestlund et al (2006) (aimed at cleaning pine forests) where the ideal amount of trees is 2500 per hectare and the cleaning unit has a size of 4 m * 4 m. The basic idea behind this algorithm is that an artificial cleaner ought to give similar results as a human cleaner (but there may be differences in how the results are obtained). The trees are classified as suitable or not suitable according to the numerical values the trees have in interesting aspects. Finally, the class 'suitable' is ordered according to the diameter of the trees and sufficient number of trees is saved in order of size by starting from the tree that has the largest diameter (the trees that are not saved are removed). None of the trees is allowed to have another tree growing closer to it than a threshold value. In this algorithm the mean diameter (at the breast height) of the uncleaned forest stand is used as one of the parameters regulating cleaning and in that way this algorithm is not strictly on-line. However, the calculated mean diameter of the uncleaned forest stand could be replaced by an approximation of the mean diameter of the forest stand in a real on-line situation. Below is a detailed description of this algorithm as it is implemented in the forest simulator:

1. When the cleaning agent arrives to a new cleaning unit it removes all trees that are too close to any tree on the cleaning units that have already been visited.

2. The cleaning agent classifies the trees into two classes: (1) those that are non-damaged and have a diameter that is between 0.6 * mean diameter and 1.6 * mean diameter, and (2) those that do not fulfill

these conditions. The trees in group (1) are labeled as suitable. The other trees are so far not suitable.

3. The group (2) (not suitable trees so far) is investigated further. First these trees are ordered according to their diameter. The trees are checked on the basis of the following aspects in order of size by starting from the tree that has the largest diameter. If a tree passes one of the following threshold values it is labelled as suitable, and it is moved to the group (1).
    a. A tree is labeled suitable if it is not damaged and if there are on average less than 4 trees per cleaning unit
    b. A tree is labeled suitable if it is a non-damaged broad-leave tree and the proportion of broad-leave trees is less than 10%
    c. A tree is labeled suitable if it is damaged and there are on average less than 3 trees per cleaning unit

4. All trees that have been labeled suitable are ordered according to their mean diameter. All trees that are too close to the tree with the largest diameter are removed. Then all trees are removed that are too close to the tree with the second largest diameter. All the trees are checked and processed in the similar way. Four trees (or less, if there are less than 4 trees left) with the largest diameters are saved which means that they are not going to be removed during the cleaning process. If there are more than 4 trees left then the thus far unsaved trees (that are labeled suitable) are checked in order of size by starting from the tree that has the largest diameter with regard to the following rules.
    a. A tree is saved if it is a conifer tree and there are on average less than 5,5 trees per cleaning unit
    b. A tree is saved if it is a birch and the proportion of birches is less than 10%
    c. A tree is saved if it is a broad-leave tree other than birch

    All trees that are not saved are removed.

This algorithm can be used for other types of forest stands than pine forest stands if the parameter values are changed to suit the specific forest type and the forest owners' goals for cleaning. The parameter values regulate for example how many trees there should be per hectare, proportion of birches and what diameter interval should be used for classification in the step 2. However, this algorithm is complicated and it may be difficult for a human being to follow what happens with different combinations of parameter values. If we have an artificial agent that uses variations of the Ap algorithm, a human user should be able to choose the parameter values of the algorithm. It is easy to understand and give parameter values that represent single goals but it is more difficult to handle parameter values that regulate goal conflicts. It would be interesting to investigate how human beings act when they choose parameter values for this kind of complicated system.

For example one thing that can be difficult for a human user to handle is the dependence of the pine forest algorithm on mean diameter of the uncleaned forest stand. The mean diameter and two parameter values regulating a good interval around the mean diameter have a large impact on the cleaning results by affecting which trees are labeled as suitable in step 2. As an example of this, a simulated forest stand (figure 2) was first treated by removing all trees that grow too close to some other tree: if two trees were closer than 0.5 meters to another tree the tree with a smaller diameter was removed (see figure 3). Then both forest stands were cleaned by using the pine forest algorithm. Differences in cleaning results are shown in figures 4 & 5. The differences mainly depend on different mean diameters in these stands and the mean diameter is one of the important parameters in Ap algorithm regulating which of the trees are classified suitable in the step 2. Both of the cleaned stands have 36 trees but four of the trees in both stands are lacking in the other stand.

The algorithm Ap can be used as a starting point for further development of cleaning algorithms. It is also of interest as a reference for comparisons with cleaning algorithms that are based on different principles than the algorithm Ap.

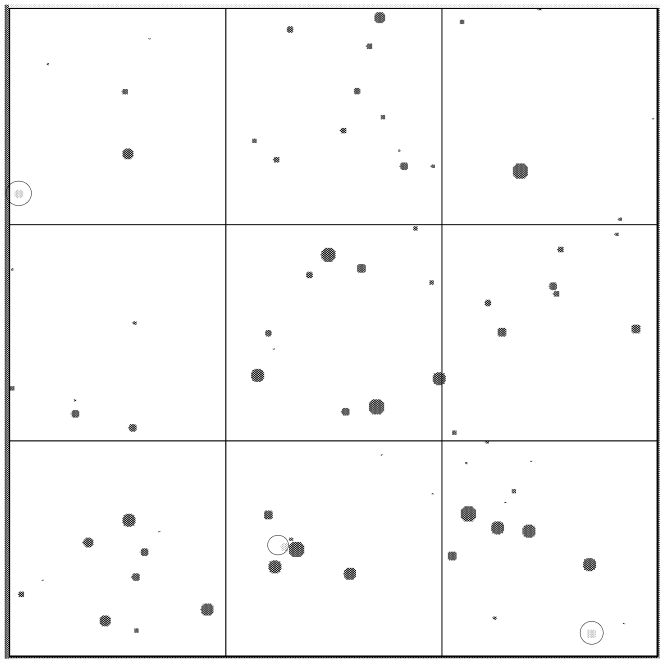


Figure 2: A simulated pine forest stand (FS1) including 79 trees. The larger the size of a dot, the larger the diameter of the tree that the dot represents. Dark dots represent conifer trees (mainly pines) and the light dots represent birch trees (marked by circles).

Forest cleaning based on simple rules

One aspect that is important for many forest owners and forest managers is that cleaning is performed fast and that it is not too expensive. Some forest managers even think what

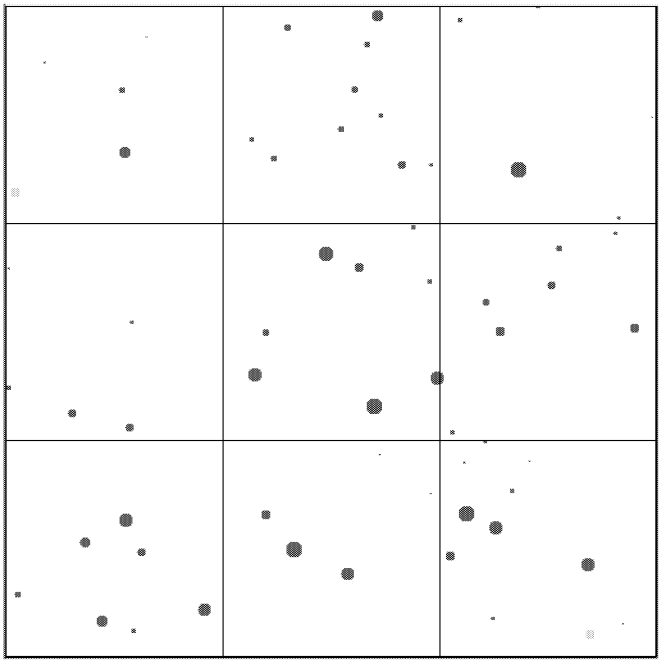


Figure 3. The simulated forest stand presented in figure 2 was treated by removing all trees that are closer than 0.5 meter from each other. When two trees were too close to each other, the tree that had smaller diameter was removed. The resulting forest stand (FS2) includes 67 trees.

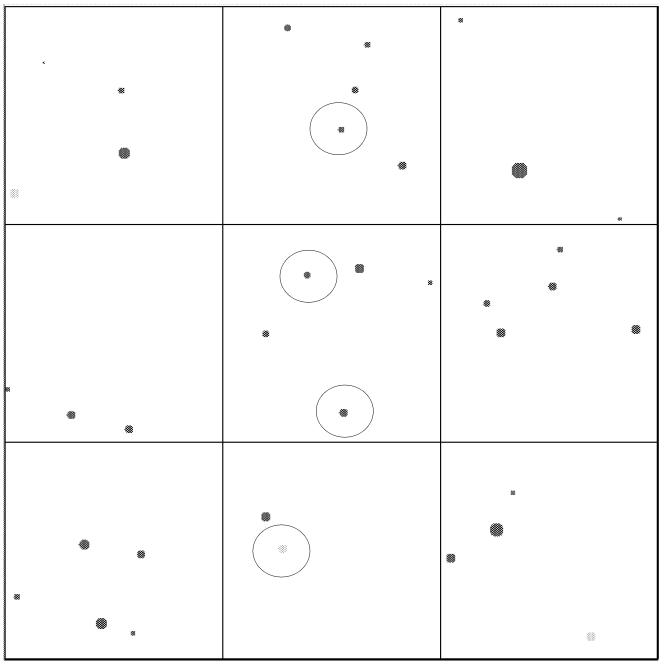


Figure 4. The simulated forest stand in figure 2 (FS1) was cleaned by the Ap algorithm. Trees missing in figure 5 are marked by circles.

is most important is that cleaning is performed and it does not matter how it is done. However, this attitude hides the fact that each human cleaner has already some idea about what is an acceptable way to clean a forest stand. For an autonomous cleaning agent it is different. It must obtain

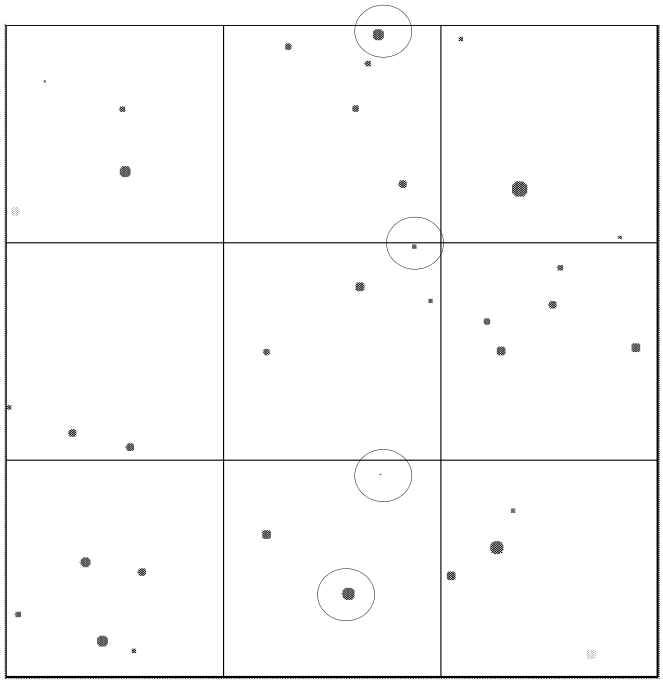


Figure 5. After removal of trees growing too close to each other in FS1 the resulting forest stand (FS2) (see figures 2 & 3) was cleaned by the Ap algorithm.

detailed information from the beginning about how it should work or/and what it should achieve as a cleaning result.

One approach to finding acceptable algorithms for artificial cleaning is to test forest cleaning based on simple rules. By using evaluation functions in the forest simulator it is thus possible to compare to what degree good cleaning results can be obtained with simple rules or rule combinations compared to other ways to clean. The idea behind the rule systems is to investigate if it is possible to obtain acceptable results using these simple systems that are easy for a human user to understand and regulate. The following three rule systems have been implemented in the forest simulator.

Simple rule system 1. Distance between trees is the only aspect that is considered. The threshold distance is chosen in a way that leads to the desired amount of trees in the cleaned forest stand. If two trees are too close to each other, the tree that has the smaller diameter is removed.
Rule system 1: If two trees are closer than *x* meters from each other, remove the tree that has the smaller diameter.

Simple rule system 2. This rule system is similar to rule system 1 except that the aspect damage is taken into consideration, in addition to the distance between trees. When two trees are too close to each other and one of them is damaged, the damaged tree is removed.
Rule system 2: If two trees are closer than *x* meters from each other and one of them is damaged, remove the damaged tree. Otherwise remove the tree that has the smaller diameter.

Simple rule system 3. This rule system includes everything in rule system 2. In addition, the desired proportion of birches is taken into consideration.

Rule system 3: If two trees are closer than x meters from each other and one of them is damaged, remove the damaged tree. If one of the trees is birch and the other one is not birch (and both are either non-damaged or damaged), save the birch if the proportion of birches is less than y % among the trees that have been saved so far. Otherwise remove the tree that has the smaller diameter.

The value of x depends on the desired amount of trees per hectare and the value of y depends on the desired proportion of birches in the cleaned forest stand.

Several variations of cleaning based on rules are interesting to test in future research. The cleaning agents could apply the rules on trees in order of size by starting from the tree that has the largest diameter. Another alternative is to check the trees in the order they are located in the cleaning unit, that is, depending on their coordinates. This would mimic the way a human being walks through a cleaning unit.

Corridor cleaning

Corridor cleaning is a type of so called geometric cleaning that has been suggested as one way to lower cleaning costs (see for example Bergström 2009). Corridor cleaning is performed using a specific forest vehicle. First a vehicle driver produces thick corridors such that there is room for the vehicle. Then the vehicle arms make smaller cleaning corridors, for example in three directions radiating from one place. A constructed (unnatural) test forest stand including 1600 trees in figure 6 was cleaned using one variant of corridor cleaning (figure 7) in order to illustrate how corridor cleaning may work.

EVALUATION FUNCTIONS FOR CLEANING RESULTS

There are several aspects that ought to be taken into consideration during cleaning and for evaluation of cleaning results even if it is difficult to state in detail what good cleaning results look like. The following goals can be seen as general goals for cleaning of young forests (see also Vestlund 2004, Ahonen-Jonnarh & Odelstad 2006).

A. Increase the mean diameter of trees.
B. Increase the proportion of undamaged trees.
C. Obtain a certain proportion of birches
D. Obtain as even as possible spatial distribution of trees
E. Obtain a certain amount of stems per area (a certain density)

It is often impossible to reach the best value for all these goals and trade-offs must be made. One way to solve this is to construct an evaluation function that aggregates the values for different aspects in a desired way. This evaluation function could then be used for testing and developing cleaning algorithms in a computer environment. If an evaluation function is complicated it may be difficult to judge if it evaluates the results in a correct way. The evaluation function could be investigated for example by checking how different elements and weight values in the evaluation function affect the total evaluation value for a large amount of cleaned forest stands. This is possible to do in the forest simulator. It could also be interesting to apply forest growth models in evaluation of the cleaning results and in that way try to predict how the cleaned forest stand will develop over several years.

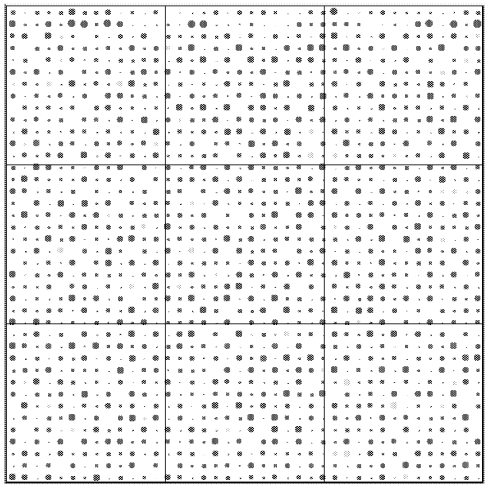


Figure 6. A constructed (unnatural) forest stand for test purposes, including 1600 trees.

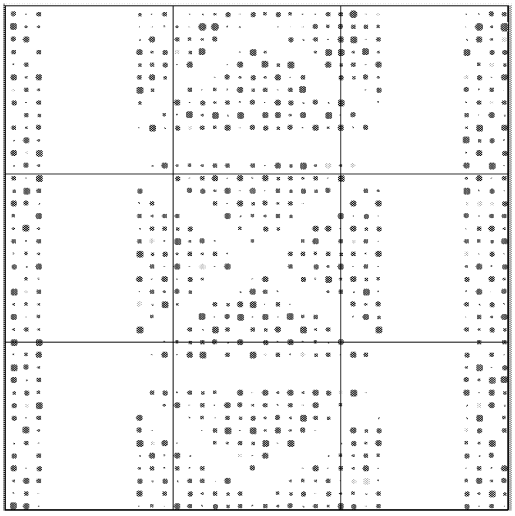


Figure 7. The constructed forest stand in figure 6 cleaned by a cleaning agent using one variant of the corridor cleaning.

A modified evaluation function (see Ahonen-Jonnarth & Odelstad 2006, inspired by Vestlund et al. (2006)) has been implemented in the forest simulator. The evaluation function includes six elements that are aggregated to a summary measure. This evaluation function is preliminary and needs to be developed further. In the forest simulator it is also possible to investigate how single values affect the total evaluation value, which may help development of this and other evaluation functions.

The evaluation function in the forest simulator evaluates and aggregates the following elements:

E1. Tree diameter for each tree is multiplied by 0.1 and the sum of these products is added to the total evaluation value.

E2. Value for each tree that depends on the eventual damage of the tree:
0.3 if tree is not damaged
-0.3 if tree is damaged
The sum of these values is added to the total evaluation value.

E3. Proximity aspect generates -0.5 for each tree that is closer than 1 meter to some other tree. The sum of these penalty values is added to the total evaluation value.

E4. Tree diameter for each tree gives the following values:
1 if the tree diameter is between 0.6 * mean diameter and 1.4 * mean diameter
-1 if the tree diameter is between 0.4 * mean diameter and 0.6 * mean diameter or if the tree diameter is between 1.4 * mean diameter and 1.6 * mean diameter
-2 if the tree diameter is lower than 0.4 * mean diameter or if the tree diameter is higher than 1.6 * mean diameter
The sum of these values is added to the total evaluation value.

E5. Proportion of birches gives the following values:
5 if the proportion of birches is between 8% and 12%
-5 if the proportion of birches is higher than 12%
-5 if the proportion of birches is lower than 8%
The sum of these values is added to the total evaluation value.

E6. Amount of trees per hectare gives the following values:
5 if the amount of trees is between 2400 and 3000
-5 if the amount of trees is higher than 3000
-5 if the amount of trees is lower than 2400
The sum of these values is added to the total evaluation value.

EXAMPLES

As examples the simulated forest stand shown in figure 2 was cleaned by agents using different ways to clean. The original forest stand included 79 trees. Four of the trees were damaged and 3 of them (4%) were birches. Cleaning using the pine forest algorithm is presented in figure 4. Figures 8 – 10 show cleaning results after cleaning by the rule systems 1 – 3. The differences between cleaning results when the rule systems were applied are few because most of the trees were non-damaged and amount of birches was low in the uncleaned forest stand and it was those two aspects that differ between the rule systems. Cleaning using the corridor cleaning agent differs significantly from the other cleaning results (see figure 11).

Summary of some aspect values in the cleaned forest stands is presented in table 1. The evaluation values for single elements and the total evaluation value are shown in table 2. Comparisons for a large amount of cleanings could be used to judge and develop evaluation functions. For example it can be clearly seen in tables 1 and 2 that a sharp difference in the evaluation value which depends on the proportion of birches generates a large difference even if the difference in the proportions of birches in different cleaned stands is small. As can be seen in tables 1 and 2, the evaluation values for elements 5 (proportion of birches) and 6 (amount of trees) may differ greatly even if the absolute differences between the aspect values are not large. This can be clearly seen when single elements are presented but in the future the calculations for these element values can be modified in order to avoid this kind of undesired effects.

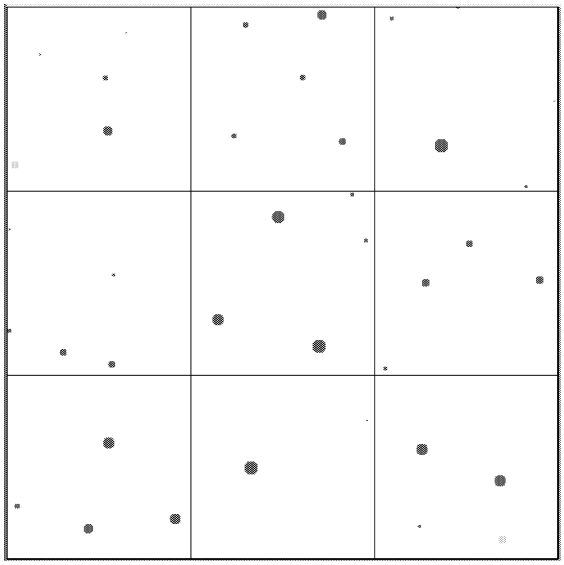


Figure 8. Forest cleaning by a cleaning agent using the simple rule system 1.

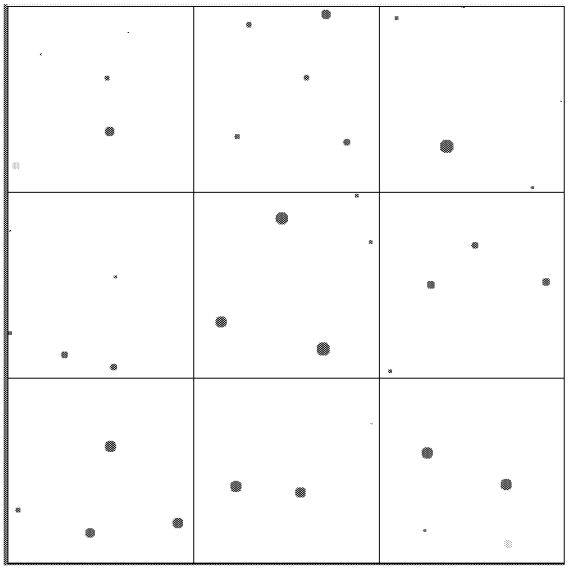


Figure 9. Forest cleaning by a cleaning agent using the simple rule system 2.

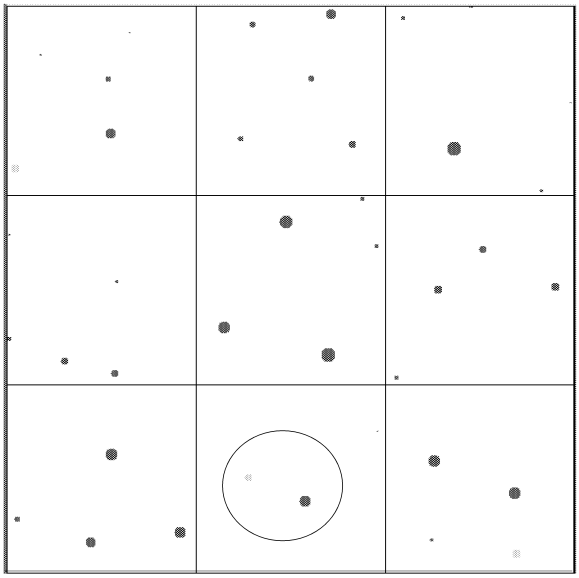


Figure 10. Forest cleaning by a cleaning agent using the simple rule system 3. Differences to figure 4 and 5 marked by a circle.

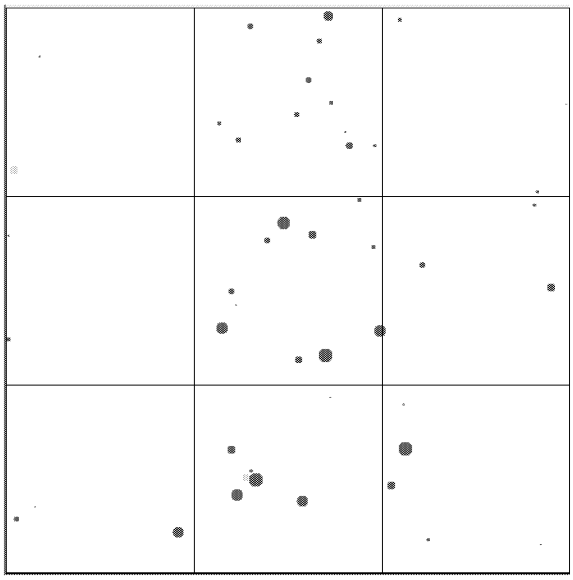


Figure 11. Forest cleaned using the corridor cleaning.

Table 1. Summary of the simulated cleaned forest stand.

Algorithm	Amount trees	Amount damages	Amount birches	Proportion birches
Ap	36	0	3	0,08
Rule system 1	40	2	2	0,05
Rule system 2	41	1	2	0,05
Rule system 3	41	1	3	0,07
Corridor	46	3	2	0,04

Table 2. Single values for different elements (E1 – E6) and total evaluation value (E tot.) for the cleaned forest stands.

Algorithm	E1	E2	E3	E4	E5	E6	E tot.
Ap	18,41	10,8	-10	23	5,0	5,0	52,21
Rule system 1	20,71	10,8	0	-18	-5,0	5,0	13,51
Rule system 2	21,34	11,7	0	-19	-5,0	5,0	14,04
Rule system 3	21,01	11,7	0	-14	-5,0	5,0	18,71
Corridor	20,62	12	-33	-14	-5,0	-5,0	-24,38

CONCLUSIONS AND FURHER RESEARCH

Cleaning is the first forest management treatment in a young forest. The forest simulator is a tool for developing and studying cleaning algorithms and evaluation functions for cleaning. This is one step toward an automation of cleaning and also toward construction of a learning simulator for the education of seasonal cleaners. For experiments and testing it is possible to simulate forest stands that belong to the same forest type as the forest stand that has been the model for simulations.

When we construct efficient algorithms for an artificial cleaner it is an advantage if we do not limit ourselves to imitate the behavior of human beings in computer environments and trust that we have 1) extracted knowledge from experts in a relevant way and 2) that the way a human being performs cleaning is the best way to do it. It can be very difficult to extract knowledge from experts, partly because it is difficult for a human being to describe how he or she actually solves tasks.

Different methods for gathering data by using global positioning system (GPS) are developing and becoming more powerful. In the future it maybe is possible to gather information about trees with an accuracy that makes it possible to extract numerical values for different aspects of trees before a cleaning agent enters the forest stand to be cleaned. This opens new possibilities for planning cleaning for an artificial agent compared to an on-line situation when all the data is not available at the beginning of the cleaning.

In the forest simulator we have implemented the Ap algorithm that is based on the earlier work of Vestlund et al. (2006). We have also implemented variations of simple rule systems that are easier for a human being to follow. One area for further research is to what degree good cleaning results can be achieved with algorihms that are based on simple rule systems compared to more complicated algorithms. In the forest simulator, it is possible to change the rule system by modifying, adding and removing rules. Another tool that is interesting for investigations of cleaning is DALMAS (deontic action-logic based multi-agent system) that is based on a normative system consisting of norms expressed in an algebraic notation (Odelstad & Boman 2004, Hjelmblom & Odelstad 2009). Another interesting area for further research is to investigate how human beings understand and regulate systems with different degrees of complexity. Thus evaluation functions need to be further developed. When this

is done it is possible to construct a cleaning agent that uses an evaluation function or part of an evaluation function directly as it performs cleaning.

REFERENCES

Ahonen-Jonnarth, U., Odelstad, J. 2006. "Evaluation of simulations with conflicting goals with application of cleaning of young forest stands." In *Proceedings of ISC* (Fourth Annual International Industrial Simulation Conference, Palermo, Italy, June 5-7), Eurosis, 498-503.

Bergström, D. 2009. "Techniques and systems for boom-corridor thinning in young dense forests." Doctoral diss. Dept. of Forest Resource Management, SLU. *Acta Universitatis agriculturae Sueciae vol. 2009:87*.

Hjelmblom M, Odelstad J. 2009. "jDALMAS: A Java/Prolog Framework for Deontic Action-Logic Multi-Agent Systems." In *Agent and Multi-Agent Systems: Technologies and Applications*, Håkansson A. et al. (eds.), LNAI 5559, pp. 110–119. Springer.

Odelstad J, Boman M. 2004. "Algebras for Agent Norm-Regulation." *Annals of Mathematics and Artificial Intelligence* 42, 141–166.

Russell, Stuart & Norvig, Peter. 2003 *Artificial Intelligence. A Modern Approach.* Second edition. Prentice Hall.

Vestlund 2004. "Assessing rules and ideas for stem selection in cleaning." *Baltic Forestry* 10: 61-71.

Vestlund K, Nordfjell T, Eliasson L and Karlsson A. 2006. "A decision support system for selective cleaning." *Silva Fennica* 40(2): 271-289.

An Intelligent Interface using a Fuzzy Model in Prevention of Forest Fire

Pilar Fuster-Parra
Sebastià Galmés
University of Balearic Islands, Cra. Valldemossa km. 7.5
07122-Palma de Mallorca, Spain
E-mail: {pilar.fuster|sebastia.galmes}@uib.es

Antoni Ligęza†
†Institute of Automatics AGH, Al. Mickiewicza 30
30-059 Kraków, Poland
E-mail: ligeza@agh.edu.pl

KEYWORDS
Wireless sensor networks, qualitative knowledge, fuzzy coefficients, linguistic variables.

ABSTRACT

Fire plays a key role in most forest ecosystems, whose outcome affects the economy, the environment and even human lives. Therefore it is crucial the prevention. In this paper we present an intelligent interface that helps to prevent wildfire using sensor networks. From the data collected by a proactive network of wireless sensors and other quantitative and/or qualitative data obtained by other means, the system colors the region under observation according to the degree of combustibility of the zone based on the qualitative and quantitative knowledge obtained and the following landmarks (in order to identify potential fire risk regions): low, moderate, high, very high and extremely high. A set of fuzzy coefficients is introduced to establish a partition in the range of each landmark, which allows us to have more detailed information from our system and therefore a more accurate assessment of fire risk.

INTRODUCTION

The wild fires constitute a problem of great gravity in our society and is one of the most dreaded natural disasters on the earth. Once they happen their effects can affect people (deaths, displaced, etc.), goods (total or partial destruction of isolated houses, electrical and telephone infrastructures, residential development, factories, agricultural and cattle operations, etc.) and environment (loss of the vegetal cover that constitutes the ecosystem of many forest inhabitants, lack of protection of the ground against the erosion, damages in the landscape and the potential recreational facilities, etc) (European, 2010). For that reason it is important to act before they happen.

Most of the works related to the subject of the wild fire using a wireless network concentrates on detection but not in their prevention (Angayarkkani, 2010), (Pripuzic, 2008), (Ramachandran, 2008). Particularly, in (Sahin, 2007) some mobile biological sensors have been used on animals to assist in early detection of forest fires.

Other works on this topic use cameras to detect smoke (Celik, 2007), (DenBreejen, 1998), with the disadvantage of the lack of contrast between the sky and green trees. In our approach we use sensors because they can be obtained at a relatively low cost and preserve some privacy. These sensors are equipped with some actuators which are able to start aspersion in case of very high or extremely high potential risk of fire.

In this article we present an intelligent interface that monitors a forest region where a proactive wireless sensor network has been strategically deployed (see Figure 1). Whereas most of the works focus exclusively on the temperature, here we consider three main meteorological factors in the assessment of fire risk: temperature, relative humidity and wind (speed and direction). Among these factors the wind is the most random and hard to predict, and the one that more changes at smaller scales. This information along with other meteorological variables like the precipitation and the solar irradiation would constitute the basic conditioner of the temporary variability of risk of wildfire occurrence in a certain zone. Other non meteorological factors like the topography, the humidity (of the ground), the amount of dead vegetal and the characteristics and hydric state of the alive vegetation that would also affect the propagation of the fire, are also taken into account.

In our system numeric features of process measures, like actual values, derivatives, means or trend estimations, must be available together with qualitative representations such as qualitative tendencies, labels or landmarks, in order to be useful for both analytical and knowledge-based supervisory methods. We use qualitative reasoning due to its applicability in real-time monitoring.

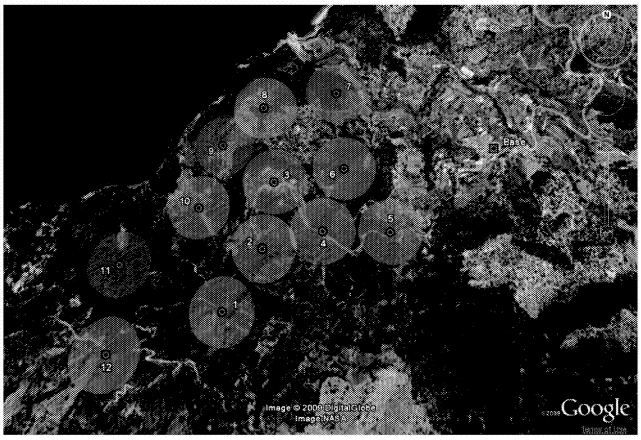


Figure 1: A proactive wireless sensor network.

Accordingly, the paper is organized as follows. In Section 2 a motivational discussion is presented. In Section 3 the concept of linguistic variables is showed and the different kind of variables that occur in our system are introduced. In Section 4 fuzzy models and rules are presented. In Section 5 an application to forest fire prevention is introduced. Finally, in Section 6, the main conclusions and suggestions for further research are drawn.

MOTIVATIONAL DISCUSSION

The uncontrolled forest fires are a serious problem in our society, since about 350 million hectares of land are burnt every year. It is known that developed countries are more susceptible to these disasters, although their damaging effects occur at a global scale (desertification, loss of bio-diversity, etc.) and can only be measured in the medium and long term.

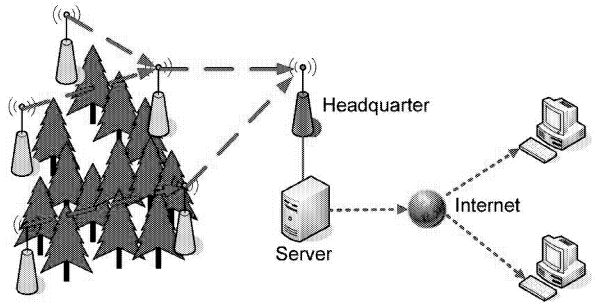


Figure 2: Getting data about temperature, humidity, wind speed and wind direction.

In this paper an intelligent interface that monitors a forest region in real time and cost-effectively through a wireless sensor network is presented (see Figure 2).
Our goal is to assess the degree of risk of fire in a particular region which will be monitored in real time. Although simple binary models seem to fit well with the majority of problems, in our particular problem a high risk could come from a defined medium risk and other possible considerations. Furthermore, in assessing the risk of forest fire using labels that characterize the situation seems essential. Therefore it makes no sense to think in terms of risk or no risk, and alternatively it seems more effective to think of different states that could occur in a particular region during monitoring.

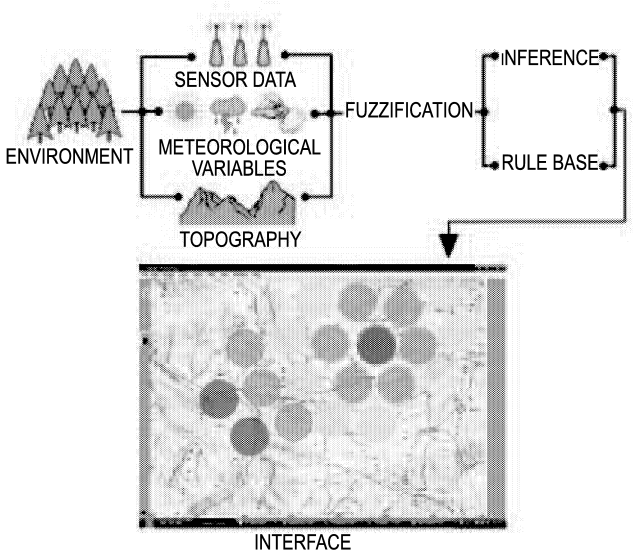


Figure 3: Taking into account the whole process to design the interface.

To establish a degree of risk, different variables are analyzed. The degree of risk generally is given by domain expert or user system, we also have in mind that to each variable a degree of risk is assigned by using natural language. In order to be close to human-like way of reasoning we use fuzzy logic (Klir, 1995) which helps in the definition of intermediate values.
In this problem there are different variables to be taken into account. The value of each variable is set to a landmark value, similarly as in (Kuipers, 1986), but here the variable has also assigned a fuzzy value.
Our ultimate goal is to get an interface (see Figure 3) to monitor a region for the prevention of forest fires.
As not all the variables have the same influence, and some of them can be obtained with qualitative knowledge by experts in the domain, a fuzzy rule base is created to help prevention. Fuzzy sets and fuzzy rule bases are created according to data obtained by sensors, data obtained by other means and domain experts.

LINGUISTIC VARIABLES

The whole set of variables and factors to be analyzed in forest fire prevention are taken from a dynamical system. The use of qualitative knowledge is essential in our problem, since it is closer to human reasoning.

Table 1: Example of data in real time.

Node	Wind(s)	Temperature	Humidity
1	5.4	22.122	72.173
2	5.3	22.214	72.383
3	5.4	22.916	72.454
4	5.2	21.314	74.528
5	5.2	21.571	71.984
6	5.3	22.001	72.062
7	5.1	22.998	72.125
8	5.2	21.921	72.287
9	5.3	22.924	71.852
10	5.3	22.173	71.900
11	5.2	23.128	72.023
12	5.1	21.852	73.311

Any physical variable of our system is considered to be a function of the form:

$$f : [a, b] \longrightarrow \mathcal{R} \tag{1}$$

Let us consider first the qualitative representation of a single variable T. The values of T variable are assigned to linguistic concepts, such as the landmarks *low, moderate, high, very high and extremely high* interpreted in the context of wild fire prevention, the resulting constructs are called *linguistic variables* (Klir, 1995).

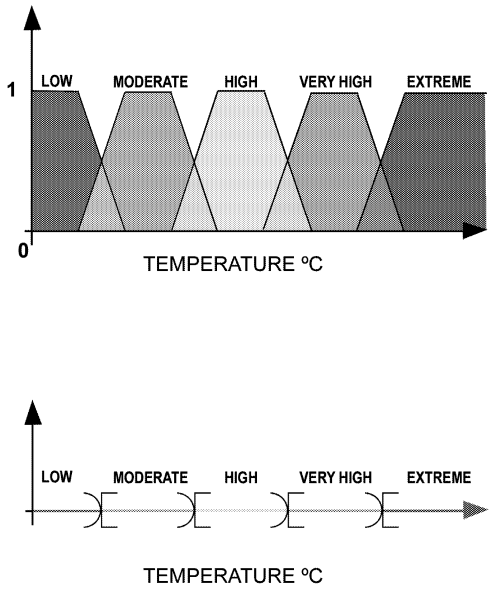


Figure 4: The temperature conceived as a fuzzy variable and a traditional (crisp) variable.

Some real values of Temperature (T) expressed in Celsius degrees, relative Humidity (H) expressed as a percentage and Wind (W) speed expressed in m/s from the wireless sensors can be seen on Table 1.

Therefore there is a finite set of landmarks for each considered variable. The set is of the form $L = \{l_1 = low, l_2 = moderate, l_3 = high, l_4 = very\ high, l_5 = extremely\ high\}$, and it is assumed that:

$$l_1 < l_2 < l_3 < l_4 < l_5 \tag{2}$$

As stated before in our problem we consider three different types of variables:

1. Temperature (T), relative Humidity (H), Wind speed (Ws) and Wind direction (Wd) variables which values are collected by a proactive network of wireless sensors.

2. Other meteorological variables: Precipitation (P) and Irradiation (I).

3. No meteorological variables: Topography (G), Amount of dead vegetal (A), Hydric state of the alive vegetation (V).

The qualitative representation of the state of a variable consists of specifying the landmark to which the actual value belongs. Any variable may have a different landmark value. So, we define the *general landmark* of the situation to be a vector of the following form:

$$\phi = [l(T), l(H), l(Ws), l(Wd), l(P), l(I), l(G), l(A), l(V)] \tag{3}$$

where any variable may have possibly a different set of its landmark values. In fact, the qualitative state vector corresponds to a qualitative state formula.
Note that not all variables have the same influence or the same strength to obtain general landmark of the situation, which will allow us to color the different areas. To get more precision in the definition of the qualitative state of any variable, fuzzy coefficients are introduced.

FUZZY MODELS AND RULES

After having the values of the different variables we assume that any variable has assigned some rough fuzzy estimation of its hazard. For instance, the membership function of a fuzzy set T is denoted by μ_T; that is,

$$\mu_T : X \longrightarrow [0, 1] \tag{4}$$

For each "fuzzy state" $\alpha_i \in [0, 1]$ we also consider the complementary $\bar{\alpha} \in [0, 1]$ such that $\alpha_i + \bar{\alpha} = 1$. This seems necessary because variables such as temperature and humidity works in an opposite way.
In order to perform operations to propagate the fuzzy measures among the variables intersection and union of fuzzy sets are considered.

The Risk of Forest Fire

Forest fire risk is defined as the probability of a fire in an area. In this approach the forest fire risk is understood as the membership function of a fuzzy set R is denoted by μ_R; that is,

$$\mu_R : X \longrightarrow [0, 1] \tag{5}$$

The combination of hazard and frequency-causality let us to determine the risk. From studies of fires, the analysis of frequency-causation is performed through two indices that reflect the frequency of fires (Frequency Rate), and the dangerousness of the causes (Index of Causality). The frequency rate is given by:

$$F_i = \frac{1}{y} \sum_{1}^{y} n_i \tag{6}$$

where:
F_i = frequency rate
n_i = number of fires of each year
y = number of years
(See Table 2 for reference data)

Table 2: Assessment of frequency rate.

FREQUENCY RATE	ASSESSMENT
0-1	low
1.1-2	moderate
2.1-4	high
4.1-6	very high
>6	extremely high

The index of causality is given by:

$$C_i = \frac{1}{y} \sum_{i=1}^{y} \frac{\sum_{j=1}^{m} cn_{jc}}{n_i} \tag{7}$$

where:
C_i = Index of causality
c = coefficient of each cause-specific hazard
n_{ic} = number of fires of each cause in each year
m = number of considered causes
n_i = number of fires of each year
y = number of years

Table 3: Assessment of index of causality.

INDEX OF CAUSALITY	ASSESSMENT
0-1	low
1.1-2	moderate
2.1-4	high
4.1-6	very high
>6	extremely high

In this paper the index of causality is established to determine the risk of fire (See Table 3 for assessment of index of causality).

APPLICATION TO FOREST FIRE PREVENTION

In order to established the potential risk of fire of a determined area all the variables have to be taken into account. So, the system has at any moment the qualitative representation of the state of all variables specified by the landmark to which the actual value belong and its fuzzy coefficient associated. So, the *general landmark* of the situation would be a vector of the following form:

$$\Phi = [(l(T), t_\mu), (l(H), h_\mu), (l(Ws), ws_\mu), (l(Wd), wd_\mu), (l(P), p_\mu), (l(I), i_\mu), (l(G), g_\mu), (l(A), a_\mu), (l(V), v_\mu)] \tag{8}$$

In short, the set of fuzzy coefficients allows us to establish a partition in the range established for each landmark, which allows us to have more detailed information from our system and therefore a more accurate assessment of fire risk.

Table 4: Risk of fire depending on temperature and wind speed.

	Wind speed				
Temp.	low	mod.	high	v. h.	e. h.
low	low	low	mod.	mod.	mod.
mod.	mod.	mod.	high	high	high
high	mod.	mod.	high	high	v. h.
v. high	high	v. h.	e. h.	e. h.	e. h.
e. h.	v. h.	v. h.	e. h.	e. h.	e. h.

Table 5: Fuzzy coefficients associated with the risk of fire depending on temperature and wind speed.

Ψ		Wind speed				
		low	mod.	high	v. h.	e. h.
Temp.	low	w_{11}	w_{12}	w_{13}	w_{14}	w_{15}
	mod.	w_{21}	w_{22}	w_{23}	w_{24}	w_{25}
	high	w_{31}	w_{32}	w_{33}	w_{34}	w_{35}
	v. h.	w_{41}	w_{42}	w_{43}	w_{44}	w_{45}
	e. h.	w_{51}	w_{52}	w_{53}	w_{54}	w_{55}

In Table 4 we can see the decision making by experts on the domain when only two variables are considered. Our problem is more complex as different variables have to be taken into account. So, what we obtain is a multi-valued situation which is coded in rules. In Table 5 the fuzzy coefficient associated to the landmarks of Table 4 are showed.

In order to increase the efficiency of our prevention model several levels of detail are required. The proposed approach offers different levels of detail as a "human oriented approach" with the goal to increase the effectiveness in our model of prevention.

CONCLUSIONS

In real problems we frequently have a fuzzy characterization of variables. The main difference with respect to other approaches which use wireless sensors is that: we consider more variables than just temperature and humidity, we focus on prevention not in detection and finally we use expert knowledge. The presented interface can be used in different ways:

1. Fuzzy model: under the assumption that we can establish the strength of influence among certain fuzzy variables, this approach could be used to establish the different landmarks of the variables, and thus the interface could prevent potential wild fires.

2. Simulation: the whole model could be used for simulation of influence of fuzzy values and therefore it could help to anticipate measures to be taken into account to protect the nature.

Further work could be oriented to other application fields, such as atmospheric contamination, drinking water quality testing or health monitoring of buildings.

ACKNOWLEDGEMENTS

This work is supported in part by the Spanish Ministry of Science and Technology under contract TIN2009-11711, with respect to the second author. Also, we would like to express our gratitude to Luis Berbiela Mingot, Head of the Forest Management and Soil Protection Service at the Regional Ministry for Environment and Mobility of the Balearic Islands, for his valuable comments on forest fire prevention and control.

REFERENCES

Angayarkkani, K., Radhakrishnan, N.(2010): An Intelligent System For Effective Forest Fire Detection Using Spatial Data. International Journal of Computer Science and Information Security (IJCSIS), vol. 7, pp. 202–205.

Celik, T., Demirel, H.,Ozkaramanli, H. and Uyguroglu, M. (2007): Fire detection using statistical color model in video sequences. J. Vis. Commun. Image R. 18, pp. 176–185.

Den Breejen, E., Breuers, M., Cremer, F, Kemp, R., Roos, M., Schutte, K. and Devries. J.(1998): Autonomous Forest Fire Detection. III International Conference on Forest Fire Research. 14th Conference on Fire and Forest Meteorology, vol. II, pp. 2003–2012.

Klir, G.J. and Yuan, B.(1995): Fuzzy Sets and Fuzzy Logic. Theory and Applications. UK: Prentice Hall International.

Kuipers, B.(1986): Qualitative simulation. Artificial Intelligence, 29(3), pp. 289-338.

Pripuzic, K., Belani, H., Vukovic, M.(2008): Early Forest Fire Detection with Sensor Networks: Sliding Window Skylines Approach. In: Lovrek, I., Howlet, R.J., Jain, L.C. (eds.) KES 2008. LNAI, vol. 5177, pp. 725–732. Springer-Verlag, Heidelberg.

Ramachandran, C., Misra, S., Obaidat, M.S.(2008): A probabilistic zonal approach for swarm-inspired wildfire detection using sensor networks. International Journal of Communication Systems, vol. 21, pp. 1047–1073.

Sahin, Y.G.(2007): Animals as Mobil Biological Sensors for Forest Fire Detection Sensors, vol. 7, pp. 3084–3099.

WEB REFERENCES

European C. (2010): European Commission Joint Research Centre (2010). Institute for Environment and Sustainability from http://ies.jrc.ec.europa

BIOGRAPHY

PILAR FUSTER-PARRA received his degree of Mathematical Sciences from the Universitat de València (València, Spain) in 1988 and his Ph. D. degree in Computer Science from the Universitat de les Illes Balears (Palma, Spain) in 1996, where she is currently Associate Professor in the Department of Mathematics and Computer Science.

SEBASTIÀ GALMÉS received his degree of Electrical Engineer from the Universitat Politècnica de Catalunya (Barcelona, Spain) in 1989 and his Ph. D. degree in Computer Science from the Universitat de les Illes Balears (Palma, Spain) in 1999, where he is currently Associate Professor in the Department of Mathematics and Computer Science. He is a member of the IFIP WG 6.3 Performance of Computer Networks since 1999 and of IEEE ComSoc from 2010.

ANTONI LIGEZA is a full professor in the domain of computer science at the AGH University of Science and Technology at Krakow, Poland. His principal area of investigation is Artificial Intelligence and Knowledge Engineering. He lectures on knowledge engineering, databases, Prolog, automated diagnosis, discrete mathematics and logics. He is a member of ACM and IEEE Computer Society.

TURTLES ARE THE TURTLES

Yassine Gangat[a], Mayeul Dalleau[b], Daniel David[a],
Nicolas Sebastien[a], Denis Payet[a].

[a] EA2525-LIM/IREMIA University of La Réunion
email: {yassine.gangat, daniel.david, nicolas.sebastien, denis.payet}@univ-reunion.fr

[b] Kélonia, l'observatoire des tortues marines - IFREMER of La Réunion
EA12-CREGUR, University of La Réunion - CNRS-CEFE, Montpellier, France
email: mayeuldalleau@kelonia.org

KEYWORDS
Agent-based simulation, NetLogo, Green turtles

ABSTRACT

Green sea turtles *Chelonia mydas* inhabit tropical and subtropical oceans worldwide. Living in the marine environment and laying eggs on the beach, they are mainly threatened by human activities (poaching, fisheries bycatch, habitat destruction, etc.). In Reunion Island, the Kélonia observatory and IFREMER develop various scientific programs to study and protect sea turtles. One of them consists in studying migrations of green sea turtles for mating purpose. As existing mathematical models struggle to take spatial dimension into account, we propose an agent-based model to study some of the numerous questions regarding green sea turtles migrations.
Coming with high expectations, experts in sea turtles also provide many heterogeneous but incomplete data. Considering available or obtainable data in one hand and the various questions of experts in the other hand, we defined an innovative modelling process in which we simultaneously conduct discussion with experts and prototyping. This paper aims at presenting our simulation model but also our approach as well as the data-collection and modelling roadmap it produced.

INTRODUCTION

Environment and biodiversity protection is one of the major actual issue. At this point, some decision-support tools are needed to assist decisioners in the process of biodiversity conservation.
This is where simulation comes into play: firstly we define the problem with as much detail as possible, then we build a conceptual model of the real system and eventually we implement it. The aim is to evaluate the consequences of multiple strategies to preserve threatened species.
In this paper, we first describe a simulation approach dedicated to green sea turtle populations. Then we discuss the design method to develop an Agent-Based Simulation (ABS) meeting thematician requirements using a platform called *NetLogo*. Lastly, we present some interesting considerations resulting from this full process.

THE GREEN TURTLES : A FRAGILE AND ENDANGERED SPECIES

Chelonia mydas, also known as *Green Turtle*, inhabits tropical and subtropical oceans and is classified among the "endangered" species of the IUCN (International Union for Conservation of Nature) Red List of Threatened Species. Like other marine turtles species, survival of green turtles is mainly dependent of direct and induced anthropogenic threats affecting all life stages. One of the major threat may certainly be intentional harvest of eggs, juvenile and adults individuals. Habitats degradation or fisheries bycatch are also known to have detrimental consequences on green turtle populations.
In order to understand the impact and possibly control such threats, a deeper understanding of the green turtle ecology is needed. This is a major concern of researchers from IFREMER (French Research Institute for Exploitation of the Sea) and Kélonia (the observatory of marine turtles), who develops scientific programs to study marine turtles in the South-West Indian Ocean (SWIO).
The usual way to build a model in population ecology is to take advantage of mathematical approaches in population dynamics model such as (Chaloupka 2002) or in individual based model such as (Mazaris et al. 2006). Such models gave some precise formalism in population parameters but some questions like spatialization are left aside.
Indeed, green turtle is a migratory species that moves across three habitat types.
This is where our expertise in the field of Agent-Based Simulation (ABS) offers an original perspective. ABS enables experts to simulate turtles behavior and ecology while taking into account the spatial aspects of the migration events. Even if our proposal is specifically applied to the SWIO, the genericity of our model would allow us to consider any population of green sea turtles around the world.
The main difficulty of this kind of project lies in the interaction of two specific groups of scientific experts. On

one hand there are experts in the field of complex simulation and ABS. On the other hand, are the thematicians, experts in another domain of Science, such as marine turtle ecologist in our case. The latter being usually not familiar with ABS, ABS experts must then be able to support them in the development of a simulation that will help them to improve their understanding of their system. In order to work hand-in-hand, we used an innovative approach described in the next part.

MODELLING PROCESS

In an ABS, agents are entities that interact through mechanisms of perception and influence within one or more environments. Those interactions would result in fluctuations in the variables of agents and environments. It is obvious that in the development of an ABS model relative to green turtles, these turtles would be the main agents. The originality of this article's title "Turtles are the turtles", lies in the fact that the keyword `turtle` is well known in the MAS community as being the historical word for agents in the NetLogo platform (Wilensky 1999). This glance blended with the need to pinpoint our objectives has led us to establish a modelling approach, where conceptual discussion and prototyping using NetLogo were done concurrently.
In the SWIO, more than 25 years of study have produced a large amount of data, sometimes incomplete. However, despite this long term studies, large gaps still remain for instance regarding migration or physiology. Behind the apparent simplicity of green turtle life history, there are very complex interacting mechanisms driving population dynamics. The construction of the model itself help in the comprehension of the biological system.
This is the reason why we conducted modelling and prototyping simultaneously. This allow us to identify problems that we hoped to answer, thus narrowing down the amount of data that are really exploitable, to identify missing data that should be collected.
Due to the fact that we already had some experiences in development of applications with NetLogo, we were able to test various hypotheses and focus more on the thematic conception of the model than the technical side. It allowed us to close-in on the thematic patterns we wanted to modellize.
The experience earned after this step (modelling and prototyping) is the first brick to the elaboration of a more mature conceptual model, which can be afterwards implemented on a poweful simulation platform more efficient in terms of calculation and detailled in terms of knowledge representation.

GENERAL PRINCIPLE OF THE MODEL

The main question that we choose to deal with was the evolution of a green turtles population on several generations. The idea was to ultimately find some conditions leading to a stabilization, an increase or a decrease in the number of individuals ; the latter possibly leading to the extinction of the population.
The particular life cycle of the green turtle brings the adult to regularly travel long distances between feeding sites and hatching sites. The feeding sites are generally coastal areas where food, mainly seagrasses, is present. In the SWIO, they are mainly located on the east coast of Africa and around Madagascar. The hatching sites are beaches where turtles mate and lay their eggs. In the SWIO, they are mostly located on islands around Madagascar and in the Mozambique channel.
At this stage of the modelling process, we chose to represent only adult individuals as agents because the early phases of their life cycle are barely known to thematicians. In addition, only female turtles were taken into account. If hypothesize a balanced sex-ratio (fifty percent of female), it allows us to double the size of the population that we can simulate. Moreover, females are the ones that usually migrate and mate every two to four years, while males migration is less documented. Representing only female adult individuals allows us again to gain more calculation power when we want to simulate with NetLogo.

IMPLEMENTATION

Following our discussion with thematicians, we came up with a model that we describe in this section. This model focuses on population dynamic and will serve as a base to answer thematical questions.

Our environment: the SWIO Area

The spatial environment is a grid applied on the SWIO, where feeding and hatching sites are implemented as `patches` in NetLogo (database structure related to a portion of the space). These `patches` are created according to the real world: their geographical coordinates provided by experts are loaded in the prototype.

- **The feeding sites.** `Patches` representing those sites have a certain quantity of food that regenerates at a determined rate. However the amount of food on this site diminishes when eaten by the `turtles`.
- **The hatching sites.** Unlike feeding sites, `patches` representing the hatching sites have no internal evolution in the current version of our prototype. They are geographic locations where green turtles come to lay their eggs.

Our agents: the turtles

In our implementation, our green (real) turtles are the (agents) `turtles` and interact with other `turtles` through the environment. They have the following properties.

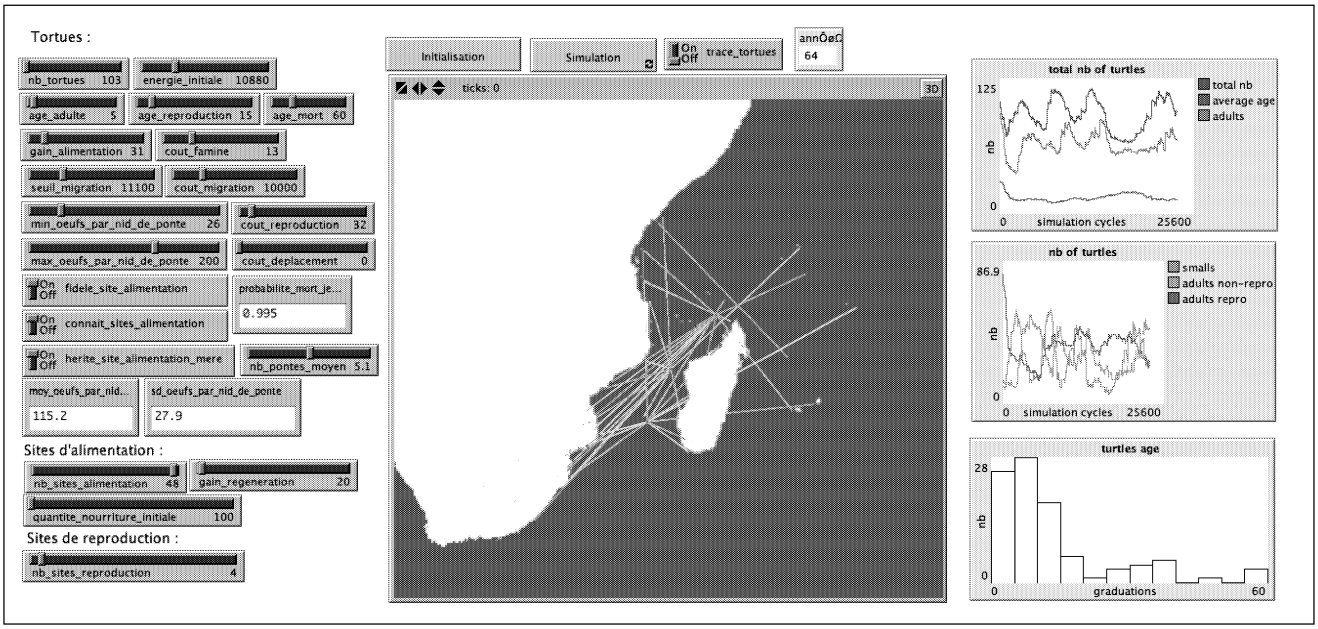


Figure 1: Prototype's GUI

- **Actions.** The main actions that `turtles` can execute are: eating (increasing their energy), migrating and reproducing (both decreasing their energy).
- **Spatial knowledge.** A green turtle almost always returns to lay eggs on the island where it was born. Each `turtle` is intrinsically linked to the `patch` representing its hatching site. However, `turtles` have access to all feeding sites' `patches`. Through this, we are able to test the influence of the fidelity of a turtle to its feeding site.
- **Birth.** The number of new `turtles` to be instantiated is calculated after each clutch, using the survival rate of juveniles (ranging from 1‰ to 1%).
- **Death.** A `turtle` is removed by the system either if it reaches the maximum age (fixed entry given by thematicians) or if it has no more energy.
- **Life Cycle.** The `turtles` stay on feeding sites where they eat available food. When a `turtle` has accumulated enough energy, it starts a migration toward its hatching site, then mates and lays eggs a certain number of times. Then it migrates back to feeding sites where it will rebuild its energy stock before the next migration cycle.

PROTOTYPE

The Graphical User Interface (GUI) of our prototype (Figure 1) is classically composed of four main areas of information that shows (from left to right):

- Variable parameters, to define simulation scenarios
- Control elements, to initialize and launch simulations
- A representation of our environment, to visualize the movement of the turtles in the SWIO
- Graphical output, to follow the evolution of system's indicators requested by thematicians.

The first simulations were just for model calibration. Most of the parameters have been determined based on thematicians knowledge (experiments and litterature). Some were not enough documented and hence went through a more delicate process; particularly at this stage, parameters which fell under the energetic aspect have been determined fairly empirically.
When our parameters were set up, we were able to launch simulation tests to study the influence of individual variations on the evolution of global population of green turtles and visually observe their behaviours: changing of feeding sites, modifying of migration periodicity or duration, etc. even if some parameters are still ignored at this stage (water currents, air temperature, etc.)

RESULTS

Simulation results

Our prototype allowed us to put in evidence evolution trends of the population of green turtles on several generations. Thus, in the upper right corner of the figure 1, we can notice important oscillations in the number of turtles. Those oscillations are in relation with inter-annual variations observed in the real world. But this prototype and the underlying simple model quickly reach certain

limits. For example, it is impossible at this stage to highlight the intra-annual cycle that corresponds to seasonal turtles' reproduction.
We identify some conditions that could lead to the stabilization, increase or decrease of the turtles' population. One of the most sensitive parameter is the low probability of survival from egg to adult. Energy parameters although played a key role in population trend, eg. food regeneration rate on feeding sites: if it is not swift enough or the number of turtles present on the site is too important, the turtle population is either greatly reduced or goes to extinction.
Another important result is that through our prototype we were able to overview the consequences of a sudden disappearance of a specific feeding site, thanks to the spatialization.

Approach results

Even if the simulation results are thematically relevant, our process method are in fact our main interest, because without the involvement of this domain's experts, we would end up solving wrong or irrelevant issues. This is why we choose to develop the prototype together with the modelling stage. As soon as our experts change something in their conceptual model, we tried to incorporate the changes into our prototype.
We worked together to build up a roadmap which contain the to-do list for both simulation and fieldwork. With this approach, we were able to identify:

- which objectives should be successful or not
- which data will be useful or not
- which data should be collected in order to reach our goals (eg. energy cost for movement, tracking of some turtles' migration path, etc.)

In the forthcoming development, we will first develop a model that implements the energetic aspect of actions. Using this base, we will then build two different models:

- The first model will deal with the variability (periodicity and quantity) of turtle's track on the beach to test if it does rely upon environmental parameters (sea surface temperature, ocean currents, etc.).
- The second model will take into account the strategy of sites selection (both feeding and nesting sites) to assess long term population viability.

When those models will be validated, we will try to merge them into one global model in order to find a temporal pattern and spatial partition. In the outlook of this project, we are going to take advantage of this first experience on NetLogo in order to achieve a more mature conceptual model, whose implementation will be made on GEAMAS-NG (a complex system's simulation platform that we develop in our laboratory).

CONCLUSION

In this paper, we presented a collaborative method and a NetLogo prototype focused on green turtles in the SWIO, in order to bring the right support to the project by clarifying what should be done or not, both in the real and virtual world (respectively fieldwork's collect of data and simulation's step of development).
Practically, we worked on two description documents:

- the description of the conceptual model
- the description of the computational model.

The conceptual description specifies the model according to how the experts think turtles shoud be modelled, while the computational description specifies the model as it is (or will be) implemented. Our task is to reduce as much as possible the gap between these two documents (called implementation gap). This reduction imposes changes on both documents, and therefore efforts have to be done by both experts in simulation and experts in fields to:

- Improve the computational description in order to make it closer to the experts' expectations
- Re-formulate the conceptual model, or even simplify it, in order to meet the technical constraints (and limits) of the computing.

In the end, the smaller this implementation gap, the more relevant our prototype. Only when this relevancy will become acceptable, the effort to migrate from "prototype" to "end-user tool" should be considered.
Our experience through this project has shown that using of NetLogo (as prototyping platform) contributes to significantly reduce the implementation gap. Indeed due to its simplicity of use, it brought both kind of experts around a set of common bases. Moreover due to a rapid development with it, it accelerates interaction between those communities.

REFERENCES

Chaloupka M., 2002. *Stochastic simulation modelling of southern Great Barrier Reef green turtle population dynamics. Ecological Modelling*, 148(1), 79 – 109.

Mazaris et al., 2006. *An individual based model of a sea turtle population to analyze effects of age dependent mortality. Ecological Modelling*, 198(1-2), 174 – 182.

Wilensky U., 1999. *NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University Evanston, IL.*

Production Planning in the Aquaculture Industry: A Simulation-based Approach

Evangelos Bellos
Vrassidas Leopoulos
Mechanical Engineering Dept
National Technical University of Athens
Iroon Polytechniou 9, 15780, Greece, Tel.: 0030 210 7723585,
email: {vbel|vleo}@central.ntua.gr

Michalis Menicou
Department of Mechanical Engineering
Frederick University
Cyprus, PO BOX 24729
Nicosia Tel:00357-22345159 (ext. 112)
email:eng.mm@fit.ac.cy

Marios Charalambides
General Studies Department, Mathematics & Physics Group,
Frederick University, Cyprus

KEYWORDS
Simulation, production planning, aquaculture industry

INTRODUCTION

Effective production management is considered to be a critical issue for the Aquaculture Industry, especially for Small and Medium Enterprises (SMEs). Important decisions concerning the fish-production planning, the sustentation of cultured fish population, the fish feed supply/inventory planning and the sales planning are affected by major uncertainty factors, related to fish biological aspects, growing/environmental conditions and business environment aspects.

Several approaches have been proposed in order to face the problem of uncertainty in fish-production, indicating a continuous research effort towards the efficiency improvement of Aquaculture companies. Within this framework, significant work has been performed in developing analytical models to investigate biological parameters affecting sea-bream breeding, while economic or operational models focus on the synergistic effect of a wide range of production parameters, proposing various optimisation strategies. Despite the scientific contribution of the aforementioned approaches, most of them are perceived as mathematically intensive optimization models, hindering their practical implementation in SMEs.

Based on the above, the present paper proposes a simulation-based approach for production planning in the Aquaculture Industry, which is stated to be rather practical and easy to implement. In particular, the proposed model aims to facilitate production planning decisions by calculating critical parameters of the production process (e.g. timeframe and delivery quantity, retained stock, fish feed quantity required, inventory cost, degree of demand satisfaction, etc.), taking into consideration the element of uncertainty and the stochastic behaviour of production parameters. Although the proposed approach is highly customised for the case of sea bream fish species breeding (Sparus Aurata), it can be applied to other industries after the necessary adaptation.

The rest of this paper is organized as follows: In the following section, a short literature review presents the status of research in modelling approaches related to the aquaculture production. The next section describes the proposed simulation-based approach for production planning. The paper ends with the conclusion section, where the expected benefits and the findings of the current research are discussed.

LITERATURE REVIEW

Several approaches exist in the literature concerning the modelling and efficiency improvement of aquaculture production systems. The basic types of models proposed are: (a) the biological models and (b) the economic models.

According to Bjorndal (1988), biological models describe the production system and its relationships with the environment, whereas economic models provide a link between the biological production system and the business/economic parameters. The construction of a biological model is usually the most difficult part of the modelling process due to the complexity of the biological organism and its interaction with the environment. Biological models can be divided into empirical and analytical ones. Empirical models are developed according to empirical data in order to investigate potential relationships among alternative environmental or breeding parameters, which affect the breeding cycle (e.g. Halachmi et al. 2005). They describe breeding techniques and investigate a limited scope of affecting parameters. On the other hand, analytical models are used to investigate biological parameters affecting sea-bream breeding (Gasca-Leyva et al. 2003; Hernandez et al. 2003; León et al. 2001; Rodrguez et al. 1997). An indicative example is the work of Hernandez et al. (2007) describing the combined effect of (a) initial fish weight, (b) ration size to the fish and (c) water temperature to the overall sea-bream breeding cycle.

Economic or operational models investigate the synergistic effect of a wide range of production parameters, such as food conversion ratio, fishing age/weight, mortality/escapees, density of fish stock per cage, number of

cages, feeding frequency, fingerling acquisition cost, fish feed cost, personnel cost, sale price coupled with sales procedure. Economic models adopt a holistic approach in order to optimise economic merits, taking into account various uncertainty elements affecting overall economic dividends. Moreover, a considerable number of optimisation tools have been developed and applied (Bjorndal et al., 2004) to optimise profit, such as linear programming (Shaftel and Wilson 1990; Pelot and Cyrus 1999), dynamic programming (Cacho et al. 1991; Arnason 1992; Heaps 1995) and multi-criteria analysis (El-Gayar and Leung 2001; Martinez-Cordero and Leung 2004).

PROPOSED APPROACH

The proposed approach incorporates simulation in the production planning process through the following steps:

Step 1. Problem formulation - Overview of the Production Process

During the first step, an extensive analysis of the fish production (growing) process was performed in order to define the simulation model's requirements. Fish stock production begins with the supply of fingerlings of a particular fish species (e.g. sea-bream) and their placement in a cage of a certain capacity. From there and on, the breeding process can be divided into five discrete phases of growing, depending on the fish average weight (Phase1 – P1: 3-10gr, Phase2 – P2: 10-40gr, Phase3 – P3: 40-100gr, Phase4 – P4: 100- 250gr and Phase5 – P5: 250-350gr). In each phase, a different type of feed (D1 to D5 accordingly) is consumed. The duration of each phase (T1 to T5 accordingly) depends on the season that the cage is placed and the environmental conditions (water temperature, oxygen level, etc). During the various phases, fish population may be reduced due to various reasons, causing their mortality (s1 to s5 mortality level %). The available fish quantity, at the end of each phase (q_1 to q_5), encompasses the alive population. At the end of the phase 5, the cultured fish stock (quantity q_5) is ready for sale.

The main uncertain factors affecting the production process are related to the:
- Quantity of fingerlings. The available quantity of fingerlings cannot be easily estimated, due to their unknown mortality level.
- Duration of growing phases. Exact estimation of the production (growing) phases' duration is impossible due to a wide variety of factors, such as quality of fingerlings placed, water temperature, oxygen levels, quality level of fish feed provided etc. Due to these factors, the estimation of the final available fish quantity, its delivery date and the required quantity of fish feed is extremely difficult to be determined.
- Fish stock production quantities per production phase. The mortality level in each production phase is theoretically known, but practically it also depends on the quality of fingerlings, the environmental conditions and the feed quality.
-

The above mentioned factors, along with the sales uncertainty (fish demand) influence critical decisions concerning the fingerling quantity needed to be placed at each production season, the date of fingerling placements, the production planning schedule, the fish feed inventory required per season, the fish feed order schedule and the fish stock quantity needed to satisfy demand per seasonal needs.

Step 2. Determination of Input and Output Variables

As soon as the overview of the production process was completed, input and output variables of the model were defined. The input variables taken into consideration for the development of the production process simulation model are: Fingerling quantity (stochastic), Fingerling placement date, Growing Phase Duration (days - stochastic), Fish Mortality Levels (% - stochastic), Fish feed (unit) consumption at each phase (Kg/10^5 fishes - stochastic), Fish feed (unit) cost (€/tn), Inventory unit cost (€/fish/day), Date of demand initiation, Demand forecast (fishes/day) (stochastic). The definition of the probability distributions needed to describe the stochastic variables was based either on extensive analysis of historical data, using ARENA software or on company experts' judgments (see Figure 1). The data presented in Figure 1 are indicative and concern only the case of cage placement in winter.

Variable	Characteristics				
Fingerling quantity (stochastic)	Triangular, TRIA(178.000, 180.000, 182.000)				
Fingerling placement date	10/01/2008				
Growing Phase Duration (days) (stochastic)	T1	Triangular, TRIA(30, 61, 92)			
	T2	Beta, 30 + 62 * BETA(0.033, 0.0185)			
	T3	Triangular, TRIA(30, 61, 92)			
	T4	Triangular, TRIA(61, 73.2, 183)			
	T5	Uniform, UNIF(122, 244)			
Fish Mortality Levels (%) (stochastic)	s1	Triangular, TRIA(9.6,10.4, 12.8)			
	s2	Triangular, TRIA(1.6, 3.2, 4.0)			
	s3	Triangular, TRIA(0,0.8,1.0)			
	s4	Triangular, TRIA(0,0.8,1.0)			
	s5	Triangular, TRIA(0. 6,0. 8,1.6)			
Fish feed (unit) consumption at each phase (Kg/10^5 fishes) (stochastic – mean value)	P1	P2	P3	P4	P5
	14.4kg D1	60.8kg D2	175kg D3	250.8kg D4	60kg D5
Fish feed (unit) cost (€/tn)	D1	D2	D3	D4	D5
	1253	1047	872	872	872
Inventory unit cost (€/fish/day): 0,001, Date of demand initiation: 30/04/2009, Demand forecast (fishes/day) (stochastic): Normal, NORMAL(600,150)					

Figure 1. Input Variables

The output variables of the model are: proposed date for fingerling placement, total fish stock production (growing) duration (days), fish stock quantity available for sale, fish feed consumption per phase, fish feed cost per phase, daily fish inventory, inventory cost, customer service level – type a and type b (fill rate). They are either of deterministic or stochastic nature, depending on the respective input variables. Figure 2 presents the formulas used for the calculation of the output variables of the model.

Variable	Calculation	Variable	Calculation
Proposed date for fingerling placement, taking into account expected demand initiation date.	Deterministic variable.	Fish feed cost per phase.	Stochastic variable. $C_{Di} = F_{Di} \times \hat{c}_{Di}$ Where, F_{Di}: fish feed consumption per phase i $\hat{c}_{Di}$: Feed unit cost (€/tn)
Total fish stock production (growing) duration (days)	Stochastic variable. $T = \sum_{i=1}^{5} t_i$ Where, T: Total production duration t_i: Duration of phase i (stochastic) i= 1 to 5	Daily fish inventory	Stochastic variable. Backorders not permitted. $I_n = I_{n-1} - \lambda_n$ Where, I_n: Fish inventrory at the end of n-th day of demand λ_n: Demand rate on n-th day (stochastic)
Fish stock quantity available for sale	Stochastic variable. $q_5 = Q \times \prod_{1}^{5} (1 - s_i)$ Where, q5: Fish stock quantity (phase 5) Q: Fingerling quantity (stochastic) s_i: Fish Mortality Levels (%) per phase i (stochastic), i= 1 to 5	Inventory cost.	Stochastic variable. $C_I = \sum_{i=1}^{N} C_{In} = \sum_{i=1}^{N} (I_n \times \hat{c}_I)$ Where, I_n: Fish inventrory at the end of n-th day of demand $\hat{c}_I$: Inventory unit cost
Fish feed consumption per phase.	Stochastic variable. $F_{Di} = q_{i-1} \times \hat{F}_{Di}$ Where, F_{Di}: fish feed consumption per phase i q_{i}: fish quantity at the end of the phase i-1 $\hat{F}_{Di}$: Fish feed (unit) consumption at phase i (Kg/10^5 fishes) i= 1 to 5	Customer Service Level – Type A. Customer Service Level – Type B (fill rate)	Probability that all customer orders arriving within the given demand period will be completely delivered from stock on hand, i.e. without delay The proportion of total demand, which is delivered without delay from stock on hand

Figure 2. Output variables

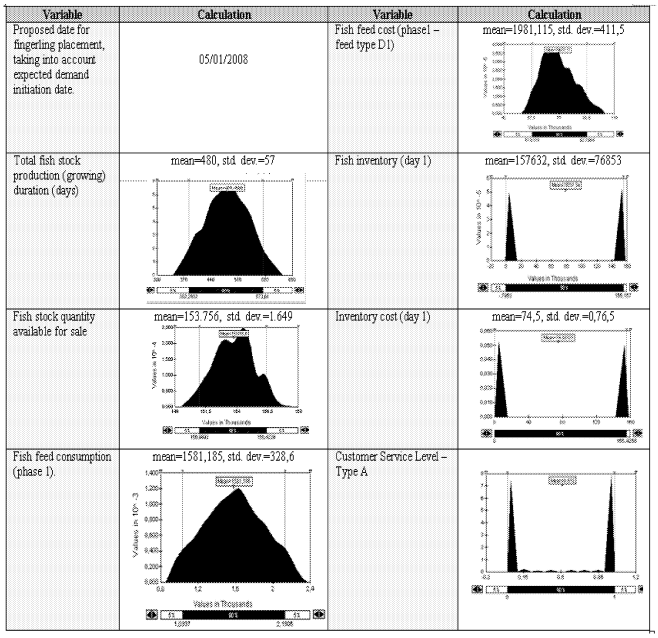


Figure 3. Simulation results (indicative)

The construction of the model takes into account the dynamic nature of the production process, where both input and output variables are highly interrelated. For example the date of fingerlings placement depends on the stochastic duration of each production phase and the expected demand initiation date. The total fish stock quantity and the fish feed consumption (and cost) depends on the duration of the production phases, the fish mortality levels and the initial fingerling quantity. Finally, the daily fish inventory (and cost) depends on the fish stock quantity produced and the demand rate.

Step 3. Simulation – Results

The Monte Carlo Simulation (MCS) was used for the calculation of the stochastic output variables. MCS is a stochastic method used to solve mathematical problems (Diamantas et al. 2007). In MCS, an objective function is defined, along with the input variables, and a random selection process is repeated many times so as to create multiple scenarios. MCS uses a pseudo-population of randomly produced alternative scenarios from prescribed statistical distributions of each input variable (Rentizelas et al. 2007). Each time a value is randomly selected for every variable of the objective function, a possible scenario is formed that leads to a certain outcome for the objective function. This process is called an iteration. The synthesis of all iterations gives an efficient number of scenarios, so that the depiction of the respectively large number of results in a density function diagram could attribute in a reliable manner the needed distribution of the objective function, showing in parallel the possibility of occurrence for each value and marking out extreme or probable results (Vose 2000). The implementation of MCS was performed by the use of Palisade @Risk™ software. Indicative results of the simulation are presented in Figure 3.

CONCLUSIONS

The aim of the paper was the development of a simulation based approach for Production Planning in the Aquaculture Industry. The proposed simulation model is able to support the medium to long term management decisions in an effective manner.

The basic conclusions drawn from the implementation of the simulation model can be summarised into the following:

- It describes in a great extent the fish production (growing) process.
- Critical parameters of the production process can be estimated in a clear and effective manner.
- The proposed model enables comprehensive modelling of uncertain parameters involved both in the production process and during demand period. However, in order to effectively draw the relative probability distributions it is advisable to collect and manage historical data.
- The simulation approach provides user the ability to examine alternative scenarios (what-if scenarios) by modifying the date of fingerlings' placement and visualise the effects of uncertainty on critical parameters, such as production (growing) duration, available fish stock quantity and Customer Service Level..
- Although the results of the proposed approach in the case of sea-bream fish species breeding (Sparus Aurata) were very promising, future research envisages more implementations in order for the model to be validated.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the contribution made by the Cyprus Research Promotion Foundation, which partly

funded this research, through the project "Operational models for process optimisation in Cyprus' fish farming industry". In addition, the authors would like to express sincere gratitude to all the partners of the aforementioned project and the organisations' managers who provided valuable time and information for the research. Finally, the authors wish to thank the editor and the anonymous reviewers for their constructive critique that helped the improvement of this paper.

REFERENCES

Bjorndal, T. 1988. "Optimal harvesting of farmed Fish.", *Marine Resource Economics*, No.5, 139–159.

Diamantas, V.; Kirytopoulos, K.; and Leopoulos, V. 2007. "Project's duration prediction: traditional tools or simulation?" *World Review of Entrepreneurship, Management and Sust. Development*, vol. 3, no.3/4, 317-333.

Gasca-Leyva, E.; Leon, C.; Hernandez, J.; and Vergara, J.M. 2003. "Bioeconomic analysis of production location of sea-bream (Sparus aurata) cultivation." *Aquaculture*, No.213, 219–232.

Halachmi, I.; Simon, Y.; Guetta, R.; and Hallerman, E. M. 2005. "A novel computer simulation model for design and management of re-circulating aquaculture systems." *Aquaculture Engineering*, No.32, 443-464.

Hernandez, J.M.; Gasca-Leyva, E.; Leon, C.J.; and Vergara, J.M. 2003. "A growth model for gilthead seabream (Sparus aurata)." *Ecological Modelling*, No.165, 265–283.

Hernandez, Juan M.; Miguel Leon-Santana; and Carmelo J. Leon. 2007. "The role of the water temperature in the optimal management of marine aquaculture." *European Journal of Operational Research*, No. 181, 872–886.

León, C.J.; Hernández, J.M; and Gasca-Leyva, E. 2001. "Cost optimization and input substitution in the production of gilthead seabream" *Aquacult. Econ. Manage.*, No.5 (3/4), 147–170.

NEPRO/0506/03 Report. 2009. "Operational models for process optimisation in Cyprus' fish farming industry." Final Report to Cyprus Research Promotion Foundation.

Rentizelas, A.; Tziralis, G.; and Kirytopoulos, K. 2007. "Incorporating uncertainty in optimal investment decisions" *World Review of Entrepreneurship, Management and Sust. Development*, No.3/4, 273-283.

Rodrguez, C.; Pérez, J.A.; Daz, M.; Izquierdo, M.S.; Hernández-Palacios, H.; and Lorenzo, A. 1997. "Influence of the EPA/DHA ratio in rotifers on gilthead seabream (Sparus aurata) larval development." *Aquaculture*, No.150, 77–89.

Vose, D. 2000. *Risk Analysis: A Quantitative Guide, Second Edition*. John Wiley, New York

BIBLIOGRAPY

EVANGELOS BELLOS is a Senior Researcher at NTUA, Research & Teaching Associate at University of Thessaly and Tutor at Hellenic Open University. He holds a Mechanical Engineering Diploma and a PhD (2004) in the field of manufacturing cost estimation and risk management. His main research interests cover project and risk management, process management and production management. He has participated in major research and consulting projects and has published number of papers at international conferences and journals.

VRASSIDAS LEOPOULOS is an Associate Professor at NTUA. He studied Mechanical and Industrial Engineering at NTUA (1980) and followed post-graduate studies in Universite Paris Ix (Dauphine). He holds a PhD (1985) in Petri – Nets simulation technique earned from the aforementioned University. His research interests cover risk management, project management and quality. He has undertaken or supervised several applied projects in the Greek industry concerning project management, risk management, e-business and ERP implementation.

MICHALIS MENICOU is an Assistant Professor of Mechanical Engineering at Frederick University. His research work can be identified in quality assurance tools development, and engineering economics applications. He applied his research expertise in a wide range of industries ranging from the construction industry, offshore aquaculture, renewable energy generation and more recently food technology.

MARIOS CHARALAMBIDES is a Lecturer of Applied Mathematics at Frederick University. His research work is concentrated mainly on numerical methods with emphasis on spectral methods and on the area of geometry of polynomials. Recently he developed expertise in the area of industrial applied optimization.

EDUCATIONAL SIMULATION

USING MULTILEVEL RANDOM COEFFICIENT MODELS TO ASSESS STUDENTS' SPELLING ABILITIES

Liberato Camilleri[1], Christine Firman[2], Victor Martinelli[3], Frank Ventura[4]

[1] Department of Statistics and Operations Research, University of Malta;
[2] Directorate of Quality and Standards in Education, Ministry of Education;
[3] Faculty of Education, University of Malta
[4] MATSEC Examinations Board, University of Malta

E-mail: liberato.camilleri@um.edu.mt

KEYWORDS
Hierarchical nested data, random coefficient model, intra class correlation, multilevel model.

ABSTRACT

This paper presents statistical models that analyze cross-sectional data related to student attainment in English and Maltese spelling. For each spelling test a random sample of 2040 students, whose age ranged from 6.5 to 16 years, was selected to examine the progression of spelling skills over time. The sample comprised equal numbers of male and female students attending state, church and private schools to investigate gender and school bias in students' spelling abilities. This hierarchical nested data can be deemed as a type of two-level data, in which the students spelling scores are level-1 units and schools are the level-2 units. This multilevel approach provides an adequate framework for modelling hierarchical data at several levels of nesting. To inspect the effect of age on student performance in English and Maltese spelling in different schools, a random coefficient model is fitted. This allows the school-specific coefficients describing individual trajectories to vary randomly when the spelling scores are regressed against the student age.

1. DATA COLLECTION

To examine the progression of spelling skills of Maltese students in primary and secondary schools, twenty age cohorts (6½, 7, 7½ … 15, 15½, 16) were identified. All students, who at the time of the test administration were two weeks younger or two weeks older than any of the specified ages, were included in the study. Moreover the selected students were also stratified by gender and school type to guarantee a representative sample. To ensure sufficiently large numbers of participants and proportionate allocations of students in each age group, it was decided that each cohort should include 60 pupils from state schools, 24 students from church schools and 18 pupils from private schools with an equal balance of boys and girls within each group.

Maltese secondary state schools are classified as Junior Lyceums, which take in students who pass the 11+ exams and Area secondary schools, which take in students who fail these exams. To ensure a proportionate allocation of participants, 30 students from Junior Lyceums, 30 pupils from Area Secondary, 24 students from church schools and 18 pupils from private schools were selected with equal numbers of boys and girls within each cohort. Each of the twenty age groups comprised 102 students making an overall total of 2040 participants for each test. The ages of all participants fell within a four week time window centred on the chronological age assessed.

The Maltese and English spelling manual compiled by (Camilleri, Firman, Martinelli 2010) was used to measure spelling skills of school children. To reduce disruptions each test, which consists of a list of 87 words ranked by order of difficulty, was administered to whole classes rather than selecting the few students from each class who fitted the 4-week age window. A sample of approximately 12,000 participants was collected for each test to get the required number of the 2040 students that fall within the required age cohorts. This sample amounted to about 20% of the whole school population in 2009 and guaranteed a maximum margin of error of around 2%. Schools were selected from all the six districts of the Maltese islands to ensure a good geographical representation. For both spelling tests the students were asked to provide the name of school, gender, age and date of birth. This information was essential to identify the students who would fall within the four week time window of each specified age and the type of school s/he attended. After finalizing data collection the scripts were sorted by date of birth. All students who satisfied the stipulated 4-week window criterion were included in the survey sample. The selected scripts were then corrected and marked, where spelling scores ranged from 0 to 87. Scripts that did not provide the required information were excluded.

2. PRELIMINARY ANALYSIS

One of the aims of the study is to inspect and evaluate the progression of spelling attainment with age and compare the spelling scores between male and female students attending different school types.

Figure 1 displays that females tend to attain higher scores in English spelling compared to males. This difference becomes more conspicuous with age. Another interesting fact is that English spelling scores increase more rapidly during primary than secondary school years.

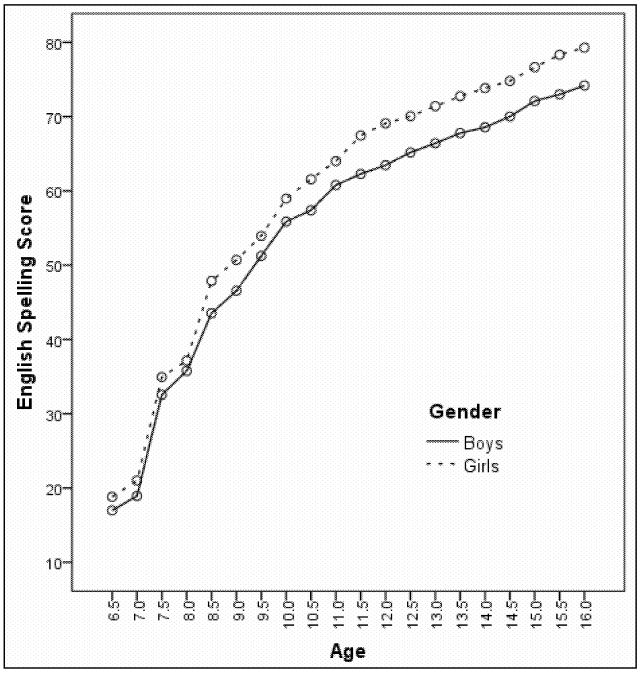


Figure 1: Mean English spelling scores categorized by age and gender

Figure 2 displays a similar attainment in English spelling of primary school children attending private and church schools. However, achievement in English spelling of primary students attending state schools is poorer. At the age of 6 the disparity in the mean spelling scores is small; however, it becomes more evident with age.

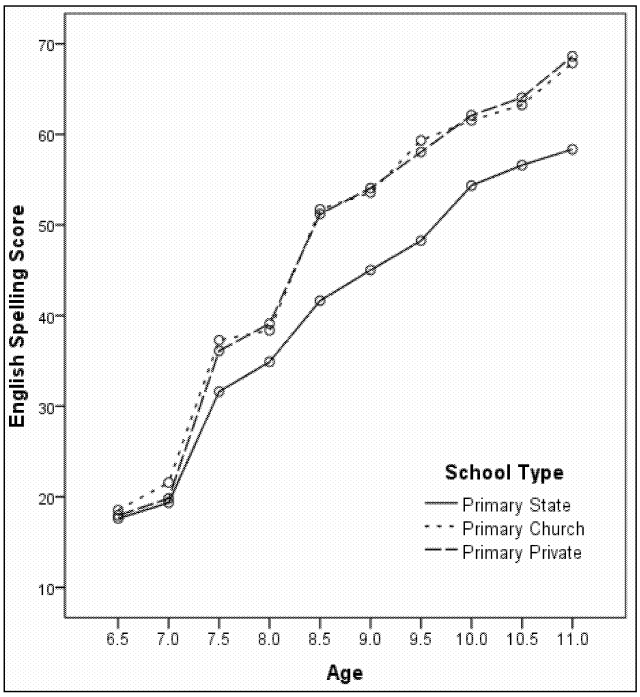


Figure 2: Mean English spelling scores of primary pupils categorized by age and school type

Figure 3 shows that increments in English spelling scores of secondary pupils are linear and less steep than those of primary students. An obvious fact is that attainment of Area secondary students in English spelling is inferior to their counterparts attending other schools. These students failed the 11+ examinations.

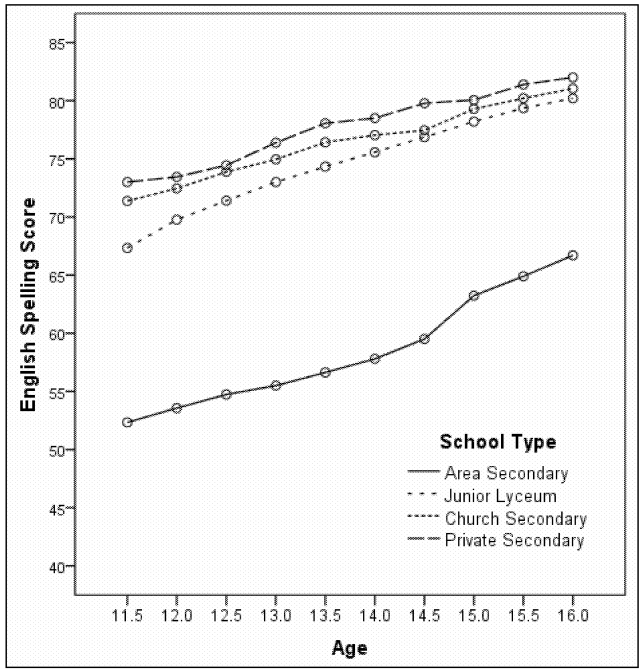


Figure 3: Mean English spelling scores of secondary students categorized by age and school type

Figure 4 displays similar patterns as Figure 1. Girls tend to outperform boys in Maltese spelling at almost all ages and discrepancies in spelling abilities increase with age. The stepwise, rather than linear, increase in the spelling scores is explained by the fact that 7½ year old students are one year ahead academically than 7-year old pupils.

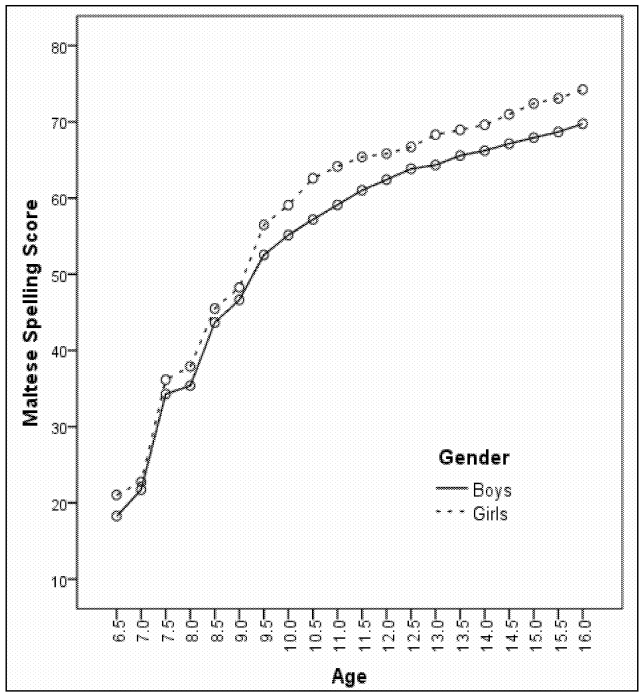


Figure 4: Mean Maltese spelling scores categorized by age and gender

Figure 5 exhibits a contrasting pattern to Figure 2. Pupils attending primary state schools perform better in Maltese spelling but worse in English. This is partly attributed to the fact that church/private schools assign more time for English than Maltese; moreover, pupils are encouraged to speak in English during school hours.

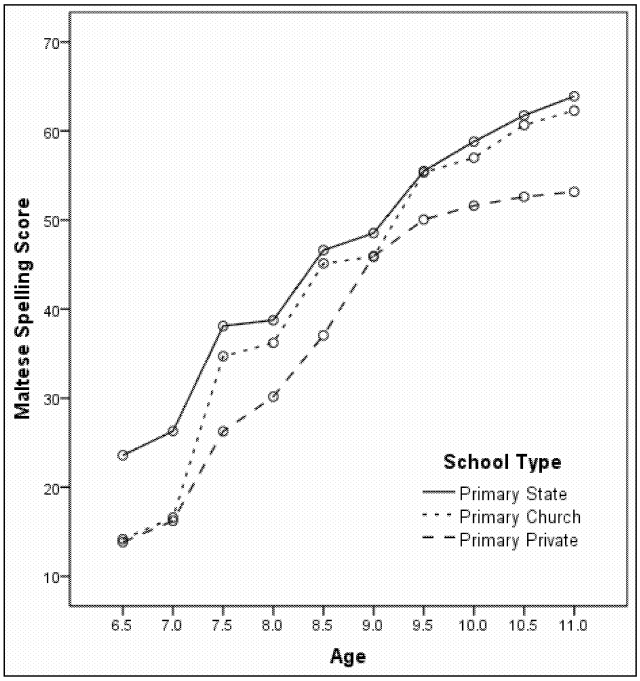


Figure 5: Mean Maltese spelling scores of primary pupils categorized by school type and gender

Figure 6 shows similar trends as Figure 3. Increments in Maltese spelling scores of secondary students are linear and less sharp than those of primary pupils. Differences in Maltese spelling attainment are conspicuous between schools. Pupils attending Junior Lyceums tend to do best, followed by church, private and area secondary schools.

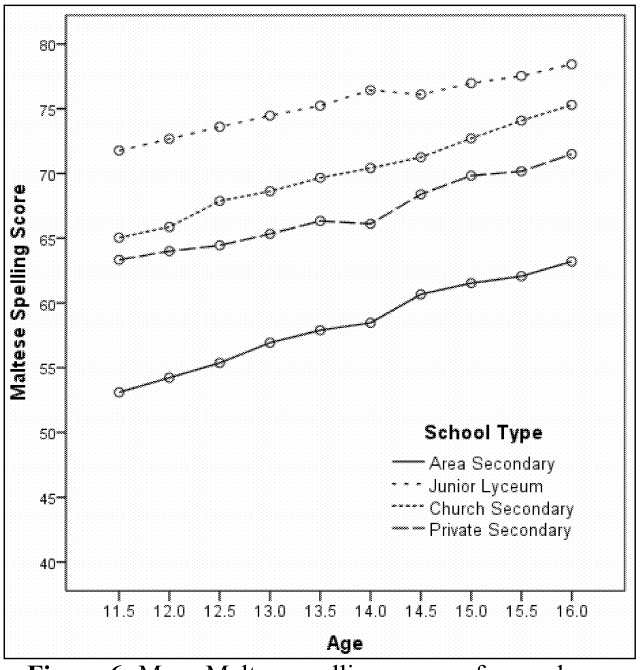


Figure 6: Mean Maltese spelling scores of secondary students categorized by school type and gender

3. A MULTILEVEL MODEL

Generalized linear mixed models are linear in the parameters and the predictors involve a mix of fixed and random effects. Linear mixed models for Normal responses can be written in the form:

$$\mathbf{y} = \mathbf{X\beta} + \mathbf{Z\eta} + \boldsymbol{\varepsilon} \tag{1}$$

$\mathbf{y}$ is a vector of responses; whereas, $\mathbf{X}$, $\mathbf{Z}$ are design matrices. $\boldsymbol{\beta}$ are fixed effects, and $\boldsymbol{\eta}$ and $\boldsymbol{\varepsilon}$ are random effects both assumed to be independent and Normally distributed.

$$\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma}) \text{ and } \boldsymbol{\eta} \sim N(\mathbf{0}, \boldsymbol{\Psi}) \tag{2}$$

$\mathbf{X\beta}$ is the fixed component and $\mathbf{Z\eta}$ is the random part of the model. Traditional linear regression models are special cases of linear mixed models with $\mathbf{Z} = \mathbf{0}$.

One of the assumptions of linear regression models is that the responses y_i are independent. This assumption is not realistic, particularly when observations are nested within hierarchical structures or are repeated measurements in a longitudinal study. Multilevel modeling is an alternative approach that facilitates the analysis of hierarchical data particularly when observations are nested within higher levels of classification. A two-level linear mixed model can be written in the form:

$$y_{ij} = \underbrace{\mathbf{X}_{ij}^{'}\boldsymbol{\beta}}_{\text{Fixed part}} + \underbrace{\sum_{m=0}^{M} \eta_{mj}^{(2)} z_{mij}^{(2)} + \varepsilon_{ij}}_{\text{Random part}} \tag{3}$$

Unobserved level-2 heterogeneity is accounted for by the inclusion of the random effect $\eta_{mj}^{(2)}$ in the linear predictor. This multilevel model accommodates well the levels of our clustered data set in which students are nested within schools.

In this application, y_{ij} is the Maltese/English spelling score attained by student *i* attending school *j*, X_{1ij} and X_{2ij} respectively stipulate the age and gender of this student and X_{3ij} specify the type of school attended by this pupil. X_{1ij} and X_{2ij} are student related predictors; whereas, X_{3ij} is a school-related explanatory variable.

The student level-1 model is:

$$y_{ij} = \delta_{0j} + \delta_{1j} X_{1ij} + \delta_{2j} X_{2ij} + \varepsilon_{ij} \tag{4}$$

The school level-2 model is:

$$\begin{aligned} \delta_{0j} &= \beta_0 + \beta_3 X_{3ij} + \eta_{0j} \\ \delta_{1j} &= \beta_1 + \eta_{1j} \\ \delta_{2j} &= \beta_2 \end{aligned} \tag{5}$$

The combined model is of the form:

$$y_{ij} = \beta_0 + \beta_1 X_{1ij} + \beta_2 X_{2ij} + \beta_3 X_{3ij} + \eta_{0j} + \eta_{1j} X_{1ij} + \varepsilon_{ij} \tag{6}$$

$\beta_0 + \beta_1 X_{1ij} + \beta_2 X_{2ij} + \beta_3 X_{3ij}$ is the fixed component, while $\eta_{0j} + \eta_{1j} X_{1ij} + \varepsilon_{ij}$ is the random component. This model implies that level-2 units are characterized by two random

effects – intercept and slope, which means that regression lines relating spelling scores to age differ between schools. It is assumed that the random effects ε_{ij}, η_{0j} and η_{1j} are independent and Normally distributed with mean 0 and variances σ_ε^2, σ_0^2 and σ_1^2 respectively.

$$\varepsilon_{ij} \sim N(0,\sigma_\varepsilon^2) \quad \begin{pmatrix}\eta_{0j}\\ \eta_{1j}\end{pmatrix} \sim N_2\left[\begin{pmatrix}0\\0\end{pmatrix},\begin{pmatrix}\sigma_0^2 & \sigma_{10}\\ \sigma_{10} & \sigma_1^2\end{pmatrix}\right] \qquad (7)$$

The GLLAMM program is a subroutine of STATA that can estimate generalized linear latent and mixed models. It can fit multilevel latent variable models for (multivariate) responses of mixed type including continuous responses, counts, survival data, dichotomous, ordered and unordered categorical responses. It maximizes the marginal log-likelihood using Newton Raphson algorithm. GLLAMM software uses numerical first and second derivatives of the log-likelihood and produces standard errors as a by-product. The intraclass correlations can be computed using the variances of the random effects.

The intraclass correlation for level-1 units is:

$$\frac{\sigma_\varepsilon^2}{\sigma_\varepsilon^2+\sigma_0^2+\sigma_1^2} \qquad (8)$$

The intraclass correlations for level-2 units are:

$$\frac{\sigma_0^2}{\sigma_\varepsilon^2+\sigma_0^2+\sigma_1^2} \qquad \frac{\sigma_1^2}{\sigma_\varepsilon^2+\sigma_0^2+\sigma_1^2} \qquad (9)$$

4. RESULTS FOR PRIMARY SCHOOLS

For reliable estimation of the parameters, adaptive quadrature was used instead of ordinary quadrature. Performance of adaptive quadrature is much better, particularly for large cluster sizes and large intraclass correlations. Moreover, adaptive quadrature is likely to give good estimates for normally distributed responses given that a sufficient number of quadrature points are used (Rabe-Hesketh, Skrondal and Pickles 2005). Adaptive quadrature required six iterations to converge. The iterative procedure required a further eight iterations running Newton Raphson to update the parameters while retaining quadrature locations and weights fixed until convergence criteria were met.

Table 1 shows the parameter estimates and standard errors of the multilevel random coefficient models for English and Maltese spelling scores of primary school children. The β_1 estimates indicate that for every 1-year increase in the age of primary school children, the spelling score is expected to increase by 10.36 for English and 9.62 for Maltese. The β_2 estimates indicate that, on average, male primary students score 3 points less than female pupils in both Maltese and English spelling. The β_3 estimates show that the mean English spelling score of students attending primary state schools is respectively 5.78 and 5.37 lower than the mean English spelling score of students attending church and private schools. Conversely, the mean Maltese spelling score of students attending primary state schools is respectively 1.63 and 7.94 higher than the mean Maltese spelling score of students attending church and private schools. Almost all parameters are significant at the 0.05 level of significance.

Primary Schools		English Spelling		Maltese Spelling	
		Est.	S.E.	Est.	S.E.
β_0	Constant	-48.2	5.842	-44.97	3.991
β_1	Age	10.36	0.595	9.62	0.381
β_2	Males	-2.97	0.886	-2.90	0.921
	Females	aliased		aliased	
β_3	State	-5.37	1.677	7.94	1.774
	Church	0.41	1.486	6.31	2.049
	Private	aliased		aliased	
σ_ε^2	$\mathrm{var}(\varepsilon_{ij})$	177.2	8.041	188.7	8.459
σ_0^2	$\mathrm{var}(\eta_{0j})$	171.3	116.2	69.46	79.38
σ_1^2	$\mathrm{var}(\eta_{1j})$	2.37	1.497	0.548	0.738
σ_{10}	$\mathrm{cov}(\eta_{0j},\eta_{1j})$	-20.14	13.04	-6.165	7.653

Table1: Parameter estimates and standard errors (English and Maltese spelling tests for primary school children)

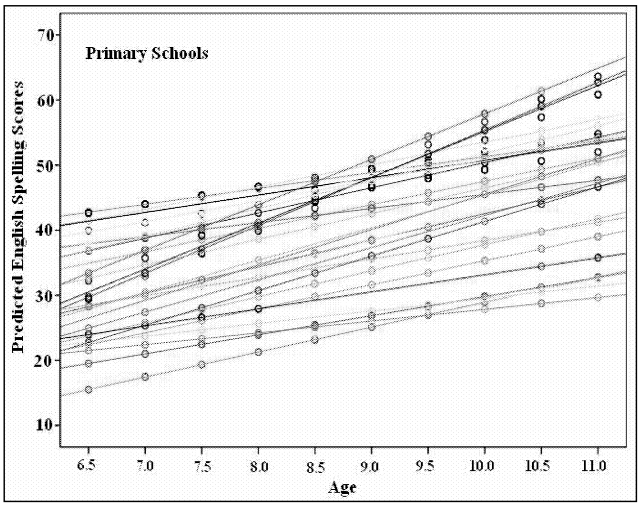


Figure7: Individual primary school trajectories displaying predicted English spelling scores against age.

The student level-1 variances σ_ε^2 for English (177.2) and Maltese (188.7) are respectively larger than the school level-2 variances $\sigma_0^2+\sigma_1^2$ for English (173.7) and Maltese (70.0). This implies that variations in spelling scores, particularly Maltese, are more attributed to differences between students than differences between schools.

For both models, the negative covariances (-20.14, -6.165) suggest that schools with lower intercepts tend to have steeper slopes. The correlations between intercepts and slopes are both close to -1. This is clearly displayed in Figure 7.

Primary Schools	Intraclass correlation	
	English	Maltese
Student level -1	0.505	0.729
School level-2 (intercept)	0.488	0.268
School level-2 (slope)	0.007	0.003

Table2: Intraclass correlations at student and school levels

Table 2 displays the intraclass correlations at student and school levels. At primary level the student level-1 variance explains about 51% and 73% of the total variability in the English and Maltese spelling scores. For both data sets the variability in the intercepts is significantly larger than the variability in the slopes.

5. RESULTS FOR SECONDARY SCHOOLS

Table 3 shows the parameter estimates and standard errors of the multilevel random coefficient models for English and Maltese spelling scores of secondary school children. The β_1 estimates indicate that for every 1-year increase in the age of secondary school children, the spelling score is expected to increase by 2.64 for English and 2.01 for Maltese. The β_2 estimates indicate that, on average, male secondary students score 5 and 4 points less than female pupils in English and Maltese spelling respectively.

Secondary Schools		English Spelling		Maltese Spelling	
		Est.	S.E.	Est.	S.E.
β_0	Constant	43.66	3.428	41.23	4.029
β_1	Age	2.64	0.227	2.01	0.261
β_2	Males	-5.01	0.675	-4.06	0.808
	Females	aliased		aliased	
β_3	Area Sec.	-18.80	1.195	-8.45	1.418
	Junior Ly.	-2.92	1.137	8.28	1.421
	Church	-1.31	1.189	3.54	1.531
	Private	aliased		aliased	
σ^2_ε	$\text{var}(\varepsilon_{ij})$	90.39	4.087	108.9	4.901
σ^2_0	$\text{var}(\eta_{0j})$	36.57	59.99	79.79	83.24
σ^2_1	$\text{var}(\eta_{1j})$	0.139	0.259	0.288	0.348
σ_{10}	$\text{cov}(\eta_{0j},\eta_{1j})$	-2.25	3.944	-4.79	5.388

Table3: Parameter estimates and standard errors (English and Maltese spelling tests for secondary school children)

The β_3 estimates illustrate that the mean English spelling score of students attending area secondary schools is respectively 15.88, 17.49 and 18.8 lower than the mean English scores of students attending Junior Lyceums, church and private schools. Conversely, the mean Maltese spelling score of students attending area secondary schools is respectively 16.73, 11.99 and 8.45 lower than the mean Maltese scores of students attending Junior Lyceums, church and private schools.

Secondary Schools	Intraclass correlation	
	English	Maltese
Student level -1	0.711	0.576
School level-2 (intercept)	0.288	0.422
School level-2 (slope)	0.001	0.002

Table4: Intraclass correlations at student and school levels

For both models, the negative covariances (-2.25, -4.79) indicate that schools with higher intercepts tend to have gentler slopes. The intraclass correlations, shown in table 4, indicate that at secondary level the student level-1 variance explains 71.1% and 57.6% of the total variability in the English and Maltese spelling scores. This implies that variations in spelling scores, particularly English, are more due to differences between students than differences between schools. The fact that the intraclass correlations for school slopes are very small indicate that multilevel random intercept models would have provided comparably good results as well. This implies that the random effects associated with the levels of the random factors enter the model as random intercept rather than random coefficient.

A recommendation for future work is to incorporate latent variables in the model fit such as student engagement. A multilevel structural equation model would then be used to accommodate a hierarchy of nested clusters when some of the variables of interest are latent.

REFERENCES

Camilleri, L., Firman, C and Martinelli, V. (2010), Manual of *Standardised Tests for Dyslexia*.

Goldstein, H. (2003), *Multilevel Statistical Models* (third edition). London: Arnold.

Longford, N. T. (1993), *Random Coefficient Models*. Oxford: Oxford University Press.

Rabe-Hesketh, S., Pickles, A. and Skrondal, A. (2001), GLLAMM: A General Class of Multilevel Models and Stata Program, *Multilevel Modelling Newsletter*, 13, 17-23.

Rabe-Hesketh, S., Skrondal, A. and Pickles, A. (2005). Maximum likelihood estimation of limited discrete dependent variable models with nested random effects. *Journal of Econometrics* 128, 301-323.

Skrondal, A. and Rabe-Hesketh, S. (2004), *Generalized Latent Variable Modelling*, Chapman & Hall/CRC.

AUTHOR BIOGRAPHY

LIBERATO CAMILLERI studied Mathematics and Statistics at the University of Malta. He received his PhD degree in Applied Statistics in 2005 from Lancaster University. His research specialization areas are related to statistical models, which include Generalized Linear models, Latent Class models, Multi-Level models and Random Coefficient models. He is presently a lecturer in the Statistics department at the University of Malta.

A Lightweight Material Library for Scientific Computing in C++

Josef Weinbub, René Heinzl,
Franz Stimpfl, Siegfried Selberherr

Institute for Microelectronics, TU Wien
Gußhausstraße 27-29 / E360
1040 Vienna, Austria
{weinbub|heinzl|stimpfl|selberherr}@iue.tuwien.ac.at

Philipp Schwaha

Shenteq s.r.o.
Záhradnícka 7
81107 Bratislava, Slovak Republic
schwaha@shenteq.com

ABSTRACT

Simulations in the field of scientific computing require often the availability of large sets of material properties. We propose a convenient approach for a lightweight material library using available open source tools. The presented approach is therefore suited for embedding into larger projects, such as simulators. The XML file format as well as an XML parser library is used to store, load, and manage the data. The location of data items or data sets is specified using XPath query language. Furthermore, an utility is provided for the conversion of the initially untyped data items to the numerical data types required by the simulation package. As performance is an issue in this context, we present a simple use case.

INTRODUCTION

Simulation tools require often a large set of (material) parameters to carry out scientific simulations [1, 2], due to the use of equations which include material parameters to model the physical environment. Among such equations, partial differential equations are especially wide spread in the description of complex phenomena and are therefore of special interest for scientific computing. A very prominent example is the system of Maxwell equations [3]:

$$\vec{\nabla} \times \vec{E} = -\partial_t \vec{B} \tag{1}$$

$$\vec{\nabla} \cdot \vec{B} = 0 \tag{2}$$

$$\vec{\nabla} \times \vec{H} = \vec{J} + \partial_t \vec{D} \tag{3}$$

$$\vec{\nabla} \cdot \vec{D} = \rho \tag{4}$$

These four equations can be split into two almost independent pairs. The first pair consists of Equation 1 and Equation 2, which relates the spatial and time derivatives of the vector fields $\vec{E}$ and $\vec{B}$ representing the electrical field strength and the magnetic flux density, respectively.

The second pair is based on Equation 3 and Equation 4, which links the spatial and time derivatives of the the magnetic field strength $\vec{H}$ and the electrical flux density $\vec{D}$. Note, that $\vec{J}$ denotes the vector field of the current density and ρ the charge density.
The Maxwell equations themselves set up a formal structure in each of the pairs, which links magnetic and electric field components. However, only by combining both sets of equations a complete dynamical system which carries energy and momentum can be achieved. This attachment is accomplished using the material relations (Equations 5a, 5b), which emphasizes the important role of material properties:

$$\vec{D} = \varepsilon \vec{E} \tag{5a}$$

$$\vec{B} = \mu \vec{H} \tag{5b}$$

These equations are of special interest, as they relate flux densities to field strengths by material properties. Here ε denotes the permittivity, and μ the permeability. Both relations appear very simple, but both permittivity and permeability may need modeling using complex, nonlinear functions depending not only on the material, but also on the magnitude of the encountered field quantities.
Considering the vast number of phenomena and the related sets of equations for which simulation environments have been and are currently being developed, it becomes apparent that many different material parameters have to be made available in a consistent and reliable manner. The challenge lies not only in the efficient storage of the material data but also in the convenient and fast data access. Another important design goal is to embed the material library into simulator environments in an orthogonal fashion, as is schematically depicted in Figure 1.
This conceptual design not only results in basic modularity during software development, but also entails that the individual modules can be changed freely without affecting the other modules. This design therefore facilitates extendability and maintainability. While flexibility is a major goal, it must not compromise the consumption of computing resources.

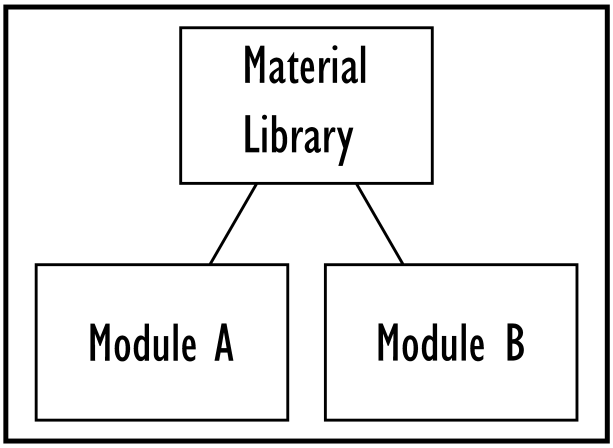


Figure 1: The material library can be part of a module set. Combining different modules results in a full scale application.

Therefore, the implementation of the material library should be as lightweight as possible. As such, memory consumption and the size of the implementation are also considered, when measuring the lightweight nature of the developed library, besides the run time performance. The modular nature ensures, that the use of the material library does not compromise the application which it is part of logically, while the lightweight nature ensures, that resources are not squandered needlessly.
At last, the run time performance of access to data is especially important, as repeated data access is typical during simulations. Therefore, minimizing the access times is important to maintain simulation performance.

LIBRARY STRUCTURE

XML [4] has been chosen as the storage format for the material data (Figure 2). The underlying data associated with materials is inherently hierarchical, as shown in Figure 3, and can be mapped to a tree naturally, which is stored using XML. The variation due to the fact that not all parameters are available or useful for all materials can also easily be accommodated by the flexible nature of XML.

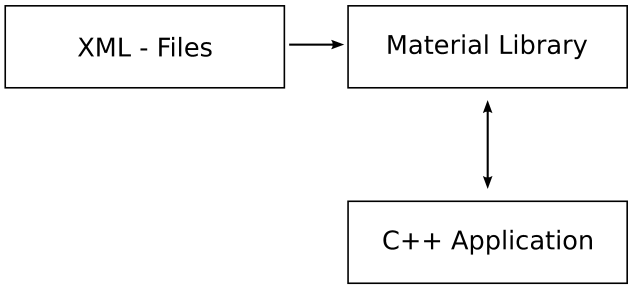


Figure 2: XML files are used as input for the material library. Data can be accessed by C++ applications.

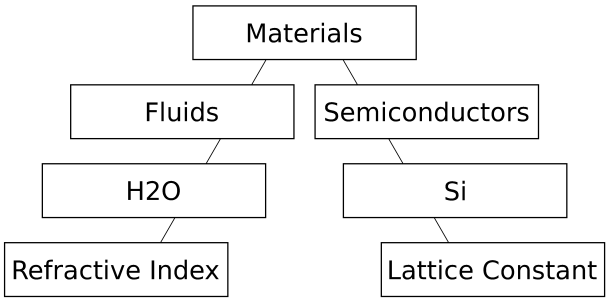


Figure 3: Material Properties schematically mapped on a Tree. Materials do not necessarily share the same properties.

The provided examples and use cases are related to embedding the XML parser library in a simulator from the field of device simulation. Since the goal is to provide a lightweight module with an already existing framework, existing client/server database approaches have not been used, as for example with PostgreSQL [5].

The material library is implemented based on a lightweight XML parser library named PugiXML [6], which is implemented in C++. Due to the basic functionality of the parser library, it is well suited to be embedded in a C++ framework. In addition to the bare XML parsing facilities, PugiXML also offers an implementation of the XPath 1.0 [7, 8] querying language. Note, that other parsing libraries have been investigated as well. Prominent examples include Xerces-C [9] and XQilla [10]. Although, XQilla offers support for XPath 2.0, it is based on the rather old and monolithically implemented Xerces-C parser. It has therefore been discarded, as it conflicts with the goals of a lightweight C++ implementation.
RapidXML has also been dismissed, as it does not offer any support for a querying language for data access. On the other hand, TinyXML [11] allows for support for a querying language to be added by a separately implemented XPath 1.0 library, named TinyXPath [12]. However, due to the fact, that PugiXML natively provides XPath support, it is the XML library of choice. It should, however, not go unnoticed that should the need arise, the XML library back end can easily be exchanged, as long as it provides XPath facilities.

XML FILE SETUP

This section discusses the chosen file setup used for the XML input files. The following XML snippet depicts parts of the schematic mapping introduced in Figure 3 thus yielding a hierarchical database.

```
<db>
  <ele>
    <id>Materials</id>
    <ele>
      <id>Semiconductors</id>
      <ele>
        <id>Si</id>
      </ele>
    </ele>
  </ele>
</db>
```

Each element has the following general setup:

```
<ele>                  // introduce a new element
  <id>name</id>        // identifiy this element
</ele>
```

Note, that the tag names do not change. The name specified within the *id* tag is used for identification. In order to uniquely accommodate several materials with the same name, unique ids, for example assigned consecutively, may be introduced.

The actual data, only text in this case, is stored in the value fields of the nodes corresponding to the various tags. This approach enables to setup hierarchies of arbitrary depth. Hence, imposing no restrictions on the setup of the database.

To store actual data values, a special node hierarchy has been chosen to not only support the commonly used floating point numbers, but also different representations, which may be more suitable under certain circumstances. The following XML snippet depicts a property node with data and representation child nodes.

```
<props>
  <data>
    <id>Lattice Constant</id>
    <repr>
      <double>0.543072</double>
    </repr>
    <unit>nm</unit>
  </data>
  <data>
    <id>Dielectric Constant</id>
    <repr>
      <double>11.8</double>
      <rational>59/5</rational>
    </repr>
  </data>
</props>
```

Note, that for each new property a new data node, with a related identifier, is introduced. Different representations can be embedded within the data node. This approach is especially of interest for robust applications, as the most suitable representation of a data value can be chosen at execution time. Additional representations can be added as the need arises. Similarly to the different representations, different units can also be specified in distinct nodes.

QUERY

The access to data is implemented by using the XPath query language. The use of a query language greatly enhances the flexibility of data access. There is no need for an additional data access layer via the Application Programming Interface (API), as the query language allows direct access to the data.
In the following, several queries are discussed to illustrate the functionality. These queries make use of the following XML data structure:

```
<db>
  <ele>
    <id>Materials</id>
    <ele>
      <id>Semiconductors</id>
      <ele>
        <id>Si</id>
        <props>
          <data>
            <id>Lattice Constant</id>
            <repr>
              <double>0.543072</double>
            </repr>
            <unit>nm</unit>
          </data>
          <data>
            <id>Dielectric Constant</id>
            <repr>
              <double>11.8</double>
              <rational>59/5</rational>
            </repr>
          </data>
        </props>
      </ele>
    </ele>
  </ele>
</db>
```

Common queries return the subtree of the data structure with the root node derived from the lowest query element. The following query accesses a specific property, for example.

```
db/ele/ele/ele/props/data[id="Lattice Constant"]
```

The XPath syntax syntax facilitates the intuitive descent along the structure of the tree holding the data. The values of the selected subtrees can then be investigated as the need arises.
As a result, the following subtree is returned.

```
<data>
  <id>Lattice Constant</id>
  <repr>
    <double>0.543072</double>
  </repr>
  <unit>nm</unit>
</data>
```

Note, that the whole node is again returned in XML format. This enables further processing of the returned data using the same mechanisms. It enables to conveniently investigate the database. This approach can be used to quickly partition and browse large databases.

However, in the case of a material database for simulations, the most common task is to directly access the data values. To directly access the values, the *text()* node can be used for an arbitrary node, as is shown in the following query

```
db/ele/ele/ele/props/data[id="Lattice Constant"]
/repr/double/text()
```

which results in:

```
0.543072
```

CONVERSION

For numerical applications the retrieved string values must be converted to numerical data types, for example, *double*. A polymorphic data type with a run time evaluation system based on the Boost Spirit Parser facilities [13] is used to this end.
At the moment, the conversion utility is only capable of dealing with double values. If the value being parsed can not be handled as a double, it is kept as a string. However, the parser facility can be conveniently extended to support additional conversion targets, such as, integer values or rational numbers for example.
The following C++ code snippet depicts the behaviour.

```
// parse the value result, get poly result
poly_data pres = convert(value_string);

// test poly result on a certain data type
// extract the data accordingly
if( is< numeric_type >(pres) ) {
   numeric_type dblval=get<numeric_type>(pres);
}
if( is< string_type >(pres) ) {
   string_type strval=get<string_type>(pres);
}
```

PERFORMANCE

This section introduces a few performance statistics for different input XML files. The query execution performance, the peak memory usage and the time required to load a file is investigated. The test platform is a PC with an AMD Phenom II X4 - 965 CPU and 8GB of RAM. The operating system is a Funtoo Linux [14] 64-bit with a 2.6.34 kernel.
Table 1 depicts the query execution performance for input files of different sizes. To investigate high load, 1E6 queries have been executed. Note, that the similar query execution times of the larger files is due to the fact, that the query depth is equally long, which is 9 for those files. Whereas, the query depth of the smaller files, is 3 levels smaller. Apparently, the query depth influences the performance more significantly, than the file size. To improve the execution performance of queries, especially for large files, the XML hierarchy should be as flat as possible, so the query depths are kept small.

XML File Size	*Total Time*	*Time per Query*
1.5 KB	5.66 s	5.66 μs
42.2 KB	18.15 s	18.15 μs
3.1 MB	121.63 s	121.6 μs
6.2 MB	126.86 s	126.8 μs
9.7 MB	132.54 s	132.5 μs

Table 1: Overview of query execution performance for 1E6 queries based on input XML files of different sizes.

XML File Size	*Peak Memory*	*File Loading*
1.5 KB	38.17 KB	<1ms
42.2 KB	181.1 KB	<1ms
3.1 MB	10.66 MB	13ms
6.2 MB	21.26 MB	26ms
9.7 MB	25.40 MB	32ms

Table 2: Overview of the peak memory consumption and the file loading performance based on input XML files of different sizes.

Table 2 depicts the peak memory usage, which has been measured with Valgrind [15], and the file loading time. Note, that the peak memory consumption can be considered exceptional as for a input file of roughly 10 MB the maximum amount of required memory is only around 25 MB. This fact emphasizes the applicability of this approach as a lightweight database for applications in the field of scientific computing.
Furthermore, the implementation of PugiXML is based on only four source files, which have roughly 280 KB of total size. Therefore, it can be easily added to a project as an external, third-party library.

USE CASE EXAMPLE

This section depicts a usecase to illustrate the application of the introduced approach in a C++ environment. The goal is to setup the library, load input XML data, and access the data by the query language. Finally, the result of the query is converted to a numerical datatype.

```
// the datastructure is instantiated
pugi::xml_document doc;

// the xml file is loaded
pugi::xml_parse_result result =
   doc.load_file("input/dev.xml");

// a query string is set up
std::string query_string("db/ele/ele/ele/props
   /data[id=\"Lattice Constant\"]
   /repr/double/text()");

// the query string is processed ..
pugi::xpath_query query(query_string.c_str());

// .. and evaluated
pugi::xpath_node_set tools =
   query.evaluate_node_set(doc);
```

Finally the string typed value has to be converted into a numerical datatype.

```
// the result is converted to a string
std::stringstream resultstream;
tools[0].node().print(resultstream, "  ");
std::string value_string(resultstream.str());

// parse the value result, get poly result
poly_data pres = convert(value_string);

// the result is tested if it is a numeric
// type and the data is extracted accordingly
if( is< numeric_type >(pres) ) {
   numeric_type quan=get<numeric_type>(pres);
```

CONCLUSION

A fast and lightweight application of a XML parser library as a material library has been introduced. The XML file setup has been discussed as well as the usage of the query language to access the data. A conversion utility enables the use of this approach in numerical applications. The peak memory consumption as well as the file loading times and the query execution times have been investigated for input files of different sizes. A use case example depicts the implementation details for using the presented approach, as well as a possible application scenario as part of a simulator environment is introduced.

ACKNOWLEDGMENTS

The authors want to thank Karl Rupp from the Christian Doppler Laboratory for Reliability Issues in Microelectronics from TU Wien for his valuable input. This work has been supported by the European Research Council through the grant #247056 MOSILSPIN and by the Austrian Science Fund FWF, project P19532-N13.

REFERENCES

[1] M. Gayer and G. Iannaccone, "A Software Platform for Nanoscale Device Simulation and Visualization," in *Advances in Computational Tools for Engineering Applications, ACTEA*, 15-17 2009, pp. 432–437.

[2] A. Logg and G. N. Wells, "DOLFIN: Automated Finite Element Computing," *ACM Transactions on Mathematical Software*, vol. 37, no. 2, pp. 1–28, 2010.

[3] J. C. Maxwell, *A Treatise on Electricity & Magnetism*. New York: Dover Publications, 1873.

[4] *Extensible Markup Language (XML) 1.0*, http://www.w3.org/TR/REC-xml.

[5] *PostgreSQL*, http://www.postgresql.org.

[6] *PugiXML*, http://code.google.com/p/pugixml.

[7] *XML Path Language (XPath) 1.0*, http://www.w3.org/TR/xpath.

[8] M. Benedikt *et al.*, "XPath Leashed," in *In ACM Computing Surveys*, 2007.

[9] *Xerces-C++ Parser*, The Apache Software Foundation, http://xerces.apache.org.

[10] *XQilla*, http://xqilla.sourceforge.net.

[11] *TinyXml*, http://sourceforge.net/projects/tinyxml.

[12] *TinyXPath*, http://tinyxpath.sourceforge.net.

[13] J. Weinbub *et al.*, "A Dispatched Covariant Type System for Numerical Applications in C++," in *International Conference of Numerical Analysis and Applied Mathematics, ICNAAM*. AIP Conference Proceedings, 2010, accepted.

[14] *Funtoo Linux*, http://www.funtoo.org.

[15] *Valgrind*, http://valgrind.org.

BIOGRAPHY

JOSEF WEINBUB studied electrical engineering and microelectronics at the *Technische Universität Wien*, where he received the degree of *Diplomingenieur* in 2009. He is currently working on his doctoral degree, where his scientific interests are in the field of scientific computing, with a special focus on algorithms and datastructures, modern programming techniques, and high-performance computing.

RENÉ HEINZL studied electrical engineering at the *Technische Universität Wien*, where he received the degree of *Diplomingenieur* in 2003 and his PhD in technical sciences in 2007. His research interests include programming paradigms, high performance programming techniques, data structural aspects of scientific computing, performance analysis, process simulation, solid modeling, scientific visualization, algebraic topology, and mesh generation and adaptation for TCAD.

FRANZ STIMPFL studied computer science at the *Technische Universität Wien*, where he received the degree of *Diplomingenieur* in 2007. He joined the Institute for Microelectronics in October 2007, where he is currently working on his doctoral degree. His research activities include mesh generation and modern software paradigms.

PHILIPP SCHWAHA studied electrical engineering at the *Technische Universität Wien*, where he received the degree of *Diplomingenieur* in 2004. He is currently working on his doctoral degree. His research activities include circuit and device simulation, device modeling, and software development.

SIEGFRIED SELBERHERR was born in Austria in 1955. He received the degree of *Diplomingenieur* in electrical engineering and the doctoral degree in technical sciences from the *Technische Universität Wien* in 1978 and 1981, respectively. Dr. Selberherr has been holding the *venia docendi* on computer-aided design since 1984. Since 1988 he has been the chair professor of the Institute for Microelectronics. His current research interests are modeling and simulation of problems for microelectronics engineering.

AUTHOR LISTING

AUTHOR LISTING

AUTHOR LISTING

AUTHOR LISTING