SIMUL 2012

The Fourth International Conference on Advances in System Simulation

ISBN: 978-1-61208-234-9

November 18-23, 2012

Lisbon, Portugal

SIMUL 2012 Editors

Petre Dini, Concordia University, Canada / China Space Agency Center, China

Pascal Lorenz, University of Haute Alsace, France

SIMUL 2012

Forward

The Fourth International Conference on Advances in System Simulation (SIMUL 2012), held on October November 18-23, 2012 in Lisbon, Portugal, continued a series of events focusing on advances in simulation techniques and systems providing new simulation capabilities.

While different simulation events are already scheduled for years, SIMUL 2012 identified specific needs for ontology of models, mechanisms, and methodologies in order to make easy an appropriate tool selection. With the advent of Web Services and WEB 3.0 social simulation and human-in simulations bring new challenging situations along with more classical process simulations and distributed and parallel simulations. An update on the simulation tool considering these new simulation flavors was aimed at, too.

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

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

We take here the opportunity to warmly thank all the members of the SIMUL 2012 technical program committee as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors that dedicated much of their time and efforts to contribute to the SIMUL 2012. We truly believe that thanks to all these efforts, the final conference program consists of top quality contributions.

This event could also not have been a reality without the support of many individuals, organizations and sponsors. We also gratefully thank the members of the SIMUL 2012 organizing committee for their help in handling the logistics and for their work that is making this professional meeting a success. We gratefully appreciate to the technical program committee co-chairs that contributed to identify the appropriate groups to submit contributions.

We hope the SIMUL 2012 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in simulation research.

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

SIMUL 2012 Chairs

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

SIMUL 2012 Research Liaison Chairs
Tae-Eog Lee, KAIST, Korea
Marko Jaakola, VTT Technical Research Centre of Finland, Finland

SIMUL 2012 Industry Liaison Chairs
Diglio A. Simoni, RTI International – RTP, USA
Shengnan Wu, American Airlines, USA
Ann Dunkin, Palo Alto Unified School District, USA
Tejas R. Gandhi, Virtua Health-Marlton, USA

SIMUL 2012 Special Area Chairs

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

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

SIMUL 2012 Publicity Chairs
Nuno Melao, Catholic University of Portugal - Viseu, Portugal

SIMUL 2012

Committee

SIMUL Advisory Chairs

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

SIMUL 2012 Research Liaison Chairs

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

SIMUL 2012 Industry Liaison Chairs

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

SIMUL 2012 Special Area Chairs

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

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

SIMUL 2012 Publicity Chairs

Nuno Melao, Catholic University of Portugal - Viseu, Portugal

SIMUL 2012 Technical Program Committee

Chrissanthi Angeli, Technological Institute of Piraeus - Athens, Greece
Simonetta Balsamo, Università Ca' Foscari Venezia, Italy
Ateet Bhalla, NRI Institute of Information Science and Technology - Bhopal, India
Georgiy Bobashev, RTI International -Research Triangle Park, USA
Dilay Celebi, Istanbul Technical University, Turkey
Soolyeon Cho, North Carolina State University - Raleigh, USA

Duilio Curcio, University of Calabria - Rende (CS), Italy
Andrea D'Ambrogio, University of Roma TorVergata, Italy
Yuya Dan, Matsuyama University, Japan
Luis Antonio de Santa-Eulalia, Université du Québec à Montréal, Canada
Luís de Sousa, Public Research Centre Henri Tudor - Luxembourg, Luxembourg
Tom Dhaene, Ghent University - IBBT, Belgium
Atakan Dogan, Anadolu University, Turkey
Bing Dong, United Technologies Research Center - East Hartford, USA
Ann Dunkin, Palo Alto Unified School District, USA
Khaled S. El-Kilany, Arab Academy for Science - Alexandria, Egypt
Paul Fishwick, University of Florida-Gainesville, USA
Ian Flood, University of Florida, USA
Tejas R. Gandhi, Virtua Health-Marlton, USA
Genady Grabarnik, St. John's University, USA
Christoph Grimm. TU Wien, Austria
Xiaolin Hu, Georgia State University, USA
Michael Hübner, Karlsruhe Institute of Technology, Germany
Marko Jaakola, VTT Technical Research Centre of Finland, Finland
Matthias Klumpp, University of Applied Sciences – Essen, Germany
Christoph Laroque, TU-Dresden, Germany
Paola Lecca, The Microsoft Research - University of Trento, Italy
SangHyun Lee, University of Michigan, USA
Fedor Lehocki, National Centre of Telemedicine Services / Slovak University of Technology in Bratislava, Slovakia
Jennie Lioris, CERMICS, France
Francesco Longo, University of Calabria, Italy
Ricardo Marcelín-Jiménez, Universidad Autónoma Metropolitana, Mexico
Marco Massetti, Universitat Ramon Llull - Barcelona, Spain,
Steve McKeever, University of Oxford, UK
Don McNickle, University of Canterbury - Christchurch, New Zealand
Nuno Melao, Catholic University of Portugal - Viseu, Portugal
Jürgen Melzner, Bauhaus-University Weimar, Germany
Marco Mevius, HTWG Konstanz, Germany
Lars Mönch, University of Hagen, Germany
Muaz Niazi, Bahria University, Islamabad, Pakistan
Michael North, Argonne National Laboratory, USA
Aida Omerovic, SINTEF & University of Oslo, Norway
Gregor Papa, Jozef Stefan Institute - Ljubljana, Slovenia
Marta Pla-Castells, Universitat de València, Spain
Jean-Francois Santucci, University of Corsica, France
Guodong Shao, National Institute of Standards and Technology - Gaithersburg, USA
Larisa Shwartz, IBM T. J. Watson Research Center - Hawthorne, USA

Eric Solano, RTI International, USA
Alexander Tatashev, Moscow University of Communications and Informatics, Russia
Michele Tizzoni, Institute for Scientific Interchange, Torino, Italy
Andrij Usov, Fraunhofer Institut IAIS, Germany
Shengyong Wang, The University of Akron, USA
Philip A. Wilsey, University of Cincinnati, USA
Shengnan Wu, American Airlines Inc. - Fort Worth, USA
Levent Yilmaz, Auburn University, USA
Yao Yiping, National University of Defence Technology - Hunan, China

Copyright Information

Table of Contents

Simulator for Smart Load Management in Home Appliances

Michael Rathmair and Jan Haase
Vienna University of Technology, Institute of Computer Technology
1040 Vienna, Gusshausstrasse 27-29/384, Austria
[rathmair|haase]@ict.tuwien.ac.at

***Abstract*—This paper describes a simulator for smart load management of standard household appliances as a base for later on-chip integration. The idea is to optimize and balance the overall power consumption of a household. Implemented devices in this work provide different stages of internal management intelligence in the view of shifting and forecasting their energy load profiles. This enables customers and energy suppliers to utilize the available electrical power in an optimized way. The simulated appliances can be run together with real household appliances to predict energy consumptions for bigger and more complex households.**

Keywords-load profile modelling; electrical load management strategies; virtual appliances simulation.

I. Introduction

Today's advances in computational technologies combined with state of the art electrical energy management systems will become a key technology for future home applications. Modern home automation systems offer for customers continuous information about their device's power consumption [1]. Energy feedback and optimized customer promotion for energy saving actions increases the usage of available electrical energy in a more effective way. As of now, no standard appliances available at the market offers the feature of being directly controlled by an external instance. That is why external measurement plugs or additional sensor/actuator hardware inside of a device are the common for consumption measurement and control function realization. Another way for power measurement of individual devices is through nonintrusive load monitoring systems analyzing a household's aggregated consumption [2].

In the near future, appliances will get more and more intelligent in the view of an increased economic and flexible operation. A main fact is that a device has the intelligence integrated itself and can react on commands and data provided by an energy management gateway. Advanced functionalities are:

- Power measurement information as previously mentioned. The current device's power consumption can be returned to the customer and energy supplier for informal feedback purposes.
- Coordinated switching and control of appliances to combine the fields of energy supply and informational technology to a smart grid application [3].
- Load forecasting for effective planning and power plant management at the energy supplier's side and similarly for customers to avoid device activations at times of high energy prices [4] [5].
- Different behaviors of the device at varying internal parameters caused by a shifted activation must be managed. The influence of user interactions have to be taken into consideration at any time. For some device classes the customer can set up a point in time the appliance must have finished its operation. For some other devices there must be immediate support.

To fulfill all these functional requirements the development of embedded hard- and software with a dedicated integration in each appliance is needed. This paper introduces in a preliminary software implementation which is able to support all the bullets listed above. The output of the simulation is a real power value indicating the current energy consumption. As a target platform for the simulation an portable Android tablet PC is chosen (see Figure 1). This hardware offers an integrated touchscreen, a high resolution display and enough computational power for load forecasting, virtual appliance management and communication tasks. Finally this portable simulator application can be used for development, prototyping and demonstration purposes of home automation and appliances control systems.

The paper follows the following structure: In Section 2, modeling techniques of home appliances are introduced; a concrete realization in the simulator application is described in Section 3. In Section 4, the implementation of the concept is stated. Finally, the paper concludes and identifies future activities in Section 5.

II. Appliances Modeling

For the simulation of typical household appliances, their load profiles must be modelled. Each device has its unique signature in real and reactive electrical power consumption. The abstraction level of the used model is chosen in a way that specific characteristics of an appliance are highlighted. To increase the clarity in the process of device power profile modelling three types of basic characteristics are defined [6]:

- ON/OFF states
- Finite State Machine (FSM)
- Continuously variable

ISBN: 978-1-61208-234-9

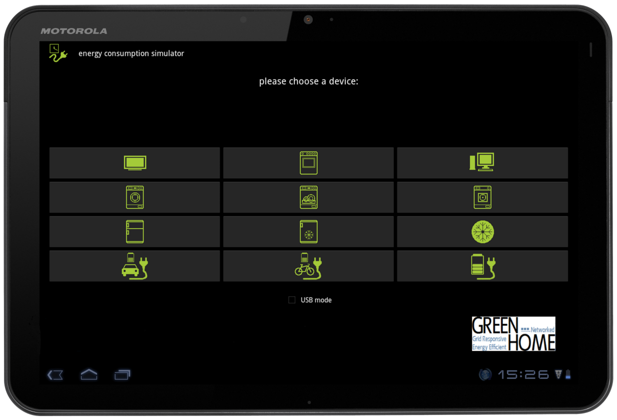

MOTOROLA
energy consumption simulator
please choose a device:
USB mode
GREEN HOME
Networked Grid Responsive Energy Efficient
15:26

Figure 1. Simulator main screen on a Motorola Xoom tablet PC. The screen shows the twelve virtual household appliances which can be selected for simulation.

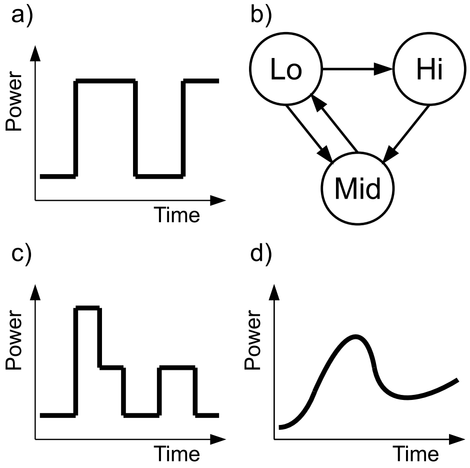


Figure 2. a) ON/OFF model with only two defined power states, b) Finite state machine model for the power profile illustrated in c), d) Continuously variable power profile.

For an ON/OFF state model a boolean variable is used to identify whether a virtual appliance is activated or deactivated. Only one single power level during the on state is allowed [6]. The off state power level can be defined by a standby consumption of the device. This type of model is ideal for the description of two level stepwise loads like heating elements, light bulbs, water pumps, etc. (see Figure 2-a).

Finite state machine (FSM) models enable a defined number of constant power states which describe the operational characteristics of a device [6]. At a graphical representation of the model, these power states (which can be specified by the activation of internal appliance components) are linked by transitions describing all possible sequences of operations (see Figure 2-b). The duration of staying in one power state is not defined in this model [6]. Devices, which are described with this modelling type, are dishwashers, washing machines, coffee machines and an increasing number of microprocessor controlled appliances (see Figure 2-c).

Finally, continuously variable load models describe appliances which power consumption is not specified as a discrete stepwise profile. The continuous load characteristic is often based on internal closed loop controllers where a rotation speed, servo position, etc., is influencing the electrical power consumption [7]. Appliances that can be described with a continuously variable model are typically all HVAC (Heating, Ventilating and Air Conditioning) devices (see Figure 2-d).

A. Modelled Load Profiles

Load profiles used for this simulator have basically a rectangular shape. They can be described as a stepwise composition of discrete and steady power levels. The advantage of modelling a load is having a mathematical representation of the power consumption signature. The description of the model and as a consequence the behaviour of the virtual device can be modified during a running simulation by updating its parameters. This modification possibility is important for a coordinated load management. The more management functionalities implemented in an appliance the more capabilities in the view of shifting the load profile in time, delaying, or a dynamic forecast of consumptions are provided [4]. These methodologies are integrated in the described simulation and discussed in more detail in Sections III-A and III-B. For time shifting and load management functions, simulation input parameters are also required. These input values can be set by widgets at the graphical user interface or received via an external hardware interface.

B. Measured Load Profiles

Figure 3-a illustrates a typical load profile structure for refrigeration devices. A two-level controller thermostat will keep the inner temperature of the appliance between the specified bounds (hysteresis of the controller). The time periods for switching on and off depends on the adjusted set-point temperature, temperature difference between in- and outside and user actions (e.g. opening and closing the device’s door). Figure 3-b shows a measured real power load profile of a washing machine. The profile can be partitioned into several blocks which identifies the internal operations of the machine. At 300 seconds of measurement time a high amplitude power consumption block of $1900W$ with a duration of eight minutes caused by the heating up phase of water is shown. The constant height power peaks between 1000 and 2000 seconds describe the cyclic rotation of the washing drum. The peaks with a height of $600W$ at

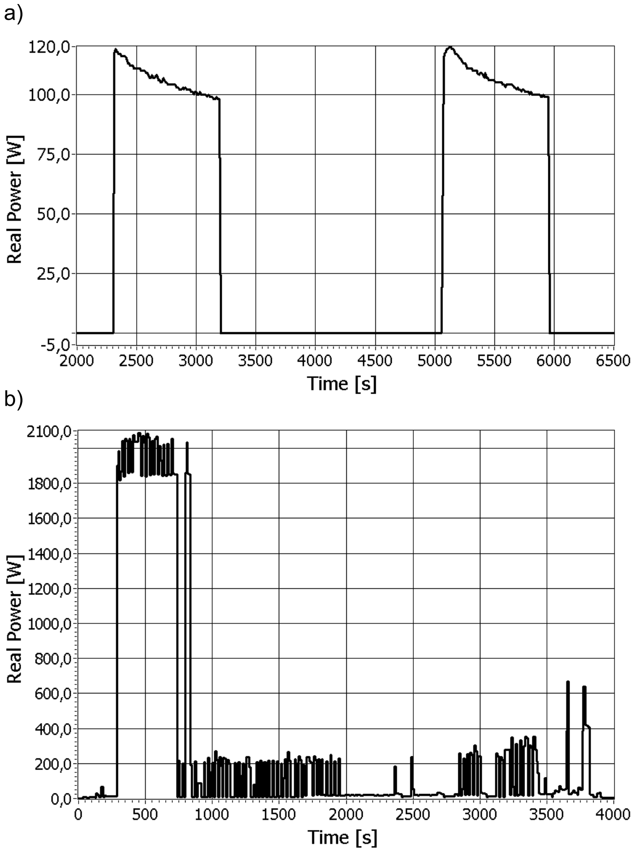


Figure 3. a) Measured real power load profile of a typical cooling device. b) Measured load profile of a washing machine at one dedicated programme.

2700 to 3900 seconds are caused by the spinning-dry and finalization period of the washing programme.

The measured load profiles of the refrigerator and the washing machine are used as a basis for the design of realistic simulation models. Power states and possible transitions can be directly applied to an appropriate model.

As a proof of a high influence of model parameters on the behaviour of a virtual device, and in the real world on the energy efficiency potential, we will give an example. If the temperature tolerance in a refrigerator for stored frozen products is defined with 1°C, all warehouses distributed in the European Union could act as a 50000 MWh "virtual energy buffer" [5].

III. Simulation System Design

The simulation application is basically partitioned into three stages. The first one is the selection of a device type. In the current version, the simulation application offers a choice between twelve typical household appliances. The second stage is the configuration of the selected virtual device. Parameter default values are displayed at appropriate screen widgets. The third stage is the simulation itself. At this simulation mode a diagram area containing the current load profile, an information field and control buttons are displayed.

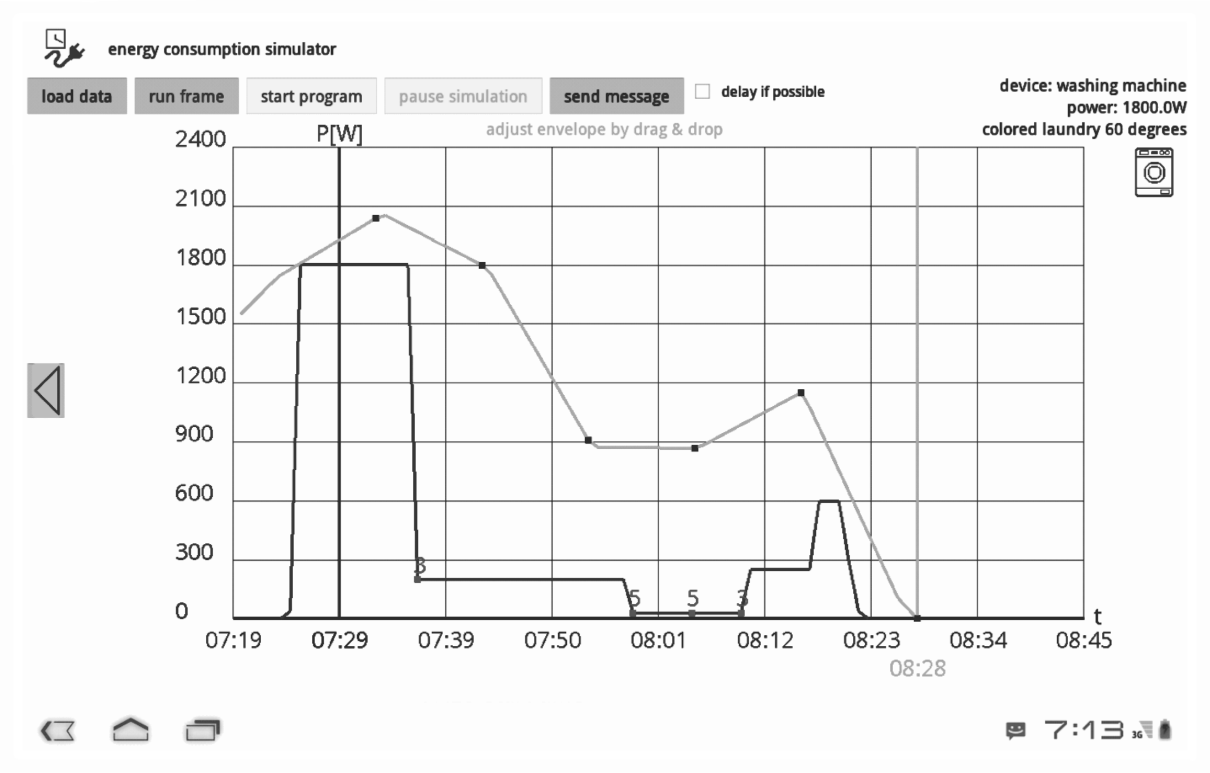
Requirements on the simulation system can be divided into functional software requirements and hardware properties, which must be fulfilled at the target system. Functional requirements describe the methods provided by the application to fulfill the mentioned objective of the simulation. The model based simulated appliances are introduced in the following itemization.

- **TV, electrical heater, computer:** These devices are represented by an ON/OFF model. The power consumption level of each virtual appliance can be set in a configuration field. The simulated virtual appliance can be turned on an off by a button widget. The activation of these appliances may not be shifted in time and the operation of the device must begin immediately.
- **washing machine, laundry dryer, dish washer:** These three types of devices allow to choose a specified operation program provided by the appliance. Depending on the program a load profile based on a FSM model with steady power states is displayed on the simulator's diagram area.
- **refrigerator, freezer, air conditioning, e-car charger, e-bike charger, general accumulator charging device:** For this class of loads the power profile shape is not fixed in time and influenced by the operation of the device. As an example the on and off periods of the refrigerator depend on the temperature inside the device and can change without notice if, e.g., the door is opened. The parameters shaping the basic power profile can be configured at the appliance's setup page.

The next two subsections describe types of devices stated at the last two itemization bullets having automatized internal load management functionalities.

A. Virtual Devices with Static Load Profiles

Appliances with a static load signature (i.e. a washing machine programme) are described. The setup of the device contains two parameters for timing configuration. At the first parameter date and time of the load profile's earliest start is configured. The second timing parameter holds the latest allowed end time and date of the washing process. If the chosen time interval is too short for the selected profile a conflict message is displayed. Finally, the energy profile for simulation is displayed at a diagram view. A second trace, configured for the simulation is an envelope function. This describes the maximum level of allowed power consumption for the operation of the appliance. This is provided by the energy supplier in order to limit the consumption of a device at high demand times [4]. The setup of the envelope function can be accomplished by drag and drop functionality at significant timing points.



device: washing machine
power: 1800.0W
colored laundry 60 degrees

Figure 4. Screenshot of a washing machine simulation with a typical load profile. The dark curve is the objected load profile (corresponding to the chosen washing programme, in this case ”colored laundry 60 degrees”) and the gray curve is the envelope function showing the maximum allowed (by a central energy balancing unit, e.g. an enhanced smart meter) energy consumption at all times. The numbers stated at the load profile curve depict minutes the point of the profile can be delayed without influencing the washing process (e.g. the start of the spin cycle might be delayed a few minutes). 08:28 is the configured latest allowed end for the completion of the washing programme.

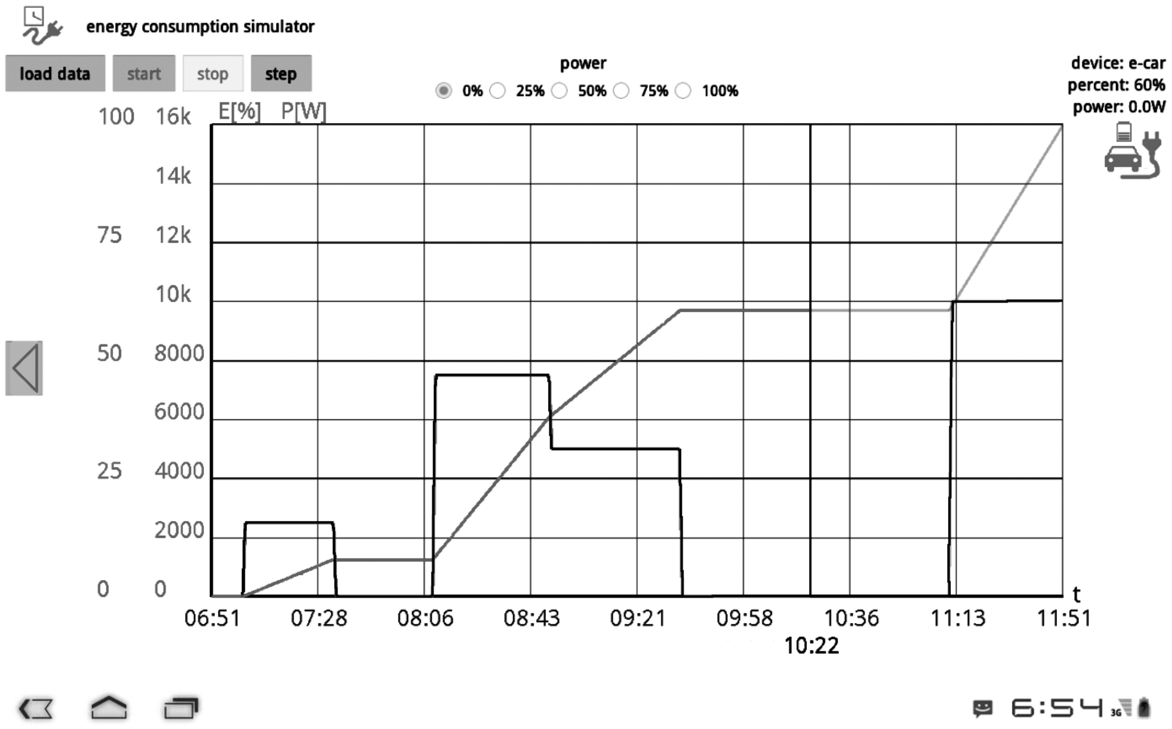
During a simulation, the program is started automatically if the complete desired load profile is less or equal than the envelope function. In other words the area under the envelope function must form a full inclusion of the simulation profile. If this condition cannot be fulfilled before the configured finish time a `no start in envelope possible` warning message is displayed. The start of the program can also be manually forced by the user. The simulation can be paused for modifying the envelope function shape. In this case, it can happen that a successful continuation of the profile is not longer possible and an error message `conflict unavoidable` is displayed. So these types of devices provide an increased ability in managing the objected load profile. The simulator can also receive the described envelope trace via the implemented external hardware interface form a higher instance energy gateway controller or home automation system.

Figure 4 illustrates a static load profile of a washing machine. The dark curve in the diagram is the desired load profile. The gray trace is an envelope function, which is defined by the user. It is shown that the starting point of the load profile is shifted in a way to fulfill the defined envelope function. In the shown simulation run, it is possible to finish the full operation of the washing machine before the latest allowed endpoint and under the constraints defined by the envelope.

B. *Virtual Devices with Dynamic Load Profiles*

These types of household appliances have a load profile which is not constant over its operation process. The load signature depends on external parameters influencing the device’s behaviour. This can be user interactions performed on the virtual appliance, environmental parameters or configurations at the virtual appliance. In simulation run mode a forecast of the load profile and internal parameters of the device are calculated. If one of the simulation input parameters change, the according forecast is not longer valid and must be recalculated. The load forecasting ability of a device is a significant advantage for the energy supplier. These data can be used for planning proposes, coordinated load management and increasing the efficiency of generation and distribution of electrical energy.

A class of appliances having a dynamic load profile implemented in this simulator is charging devices



power
● 0% ○ 25% ○ 50% ○ 75% ○ 100%

device: e-car
percent: 60%
power: 0.0W

Figure 5. Simulation screenshot of an electrical car charger. The black curve is the resulting load profile dynamically influenced by the user. The gray curve is the corresponding charging state of the accumulator. 10:22 is the current time of simulation. The curves displayed on the right side of 10:22 are the predicted (by the simulation) load profile and percentage of the charging process.

for electrical accumulators (i.e. e-car, e-bike). At the configuration page start and the latest desired end time and date can be set up. Two additional parameters configure the capacity of the accumulator and the maximum charging power. If the device can not be fully charged under the specified timing and accumulator properties, a warning message is displayed. During the simulation a planned load profile for fully charging the accumulator and the charging percentage level is displayed. The charging power can also be modified manually. After this action, the forecasted profile is recalculated in a way that the timing requirement for the end of the charging process is still fulfilled. If the manual setup of charging power would not lead to a fully charged accumulator at the specified end time, the charging power value will be forced and increased automatically to 100%.

Figure 5 for example shows the charging process of an electrical car. With the radio buttons the charging power can be manually modified. The slope of the charging percentage state is proportional with the configured charging power. The forecasted increase of charging power to 100% at 11:13 is calculated automatically to fulfill the requirement of a fully charged accumulator at the configured end time.

IV. Implementation

This section describes the concrete implementation of the functionalities described in prior sections. Some details about system requirements, software structure, user interaction and external interfaces are introduced.

A. *Software Structure*

The simulation software is implemented as a package of Java classes. The main class instantiates all other modules and manages the control flow through the application. The user can select a device and is lead to the configuration page of the virtual appliance. For each type of appliance, a specific configuration class is implemented. After configuring the device parameters the simulation can be started. All simulation algorithms of a device are implemented in a simulation run class and, called after each step of simulation time. The run can be executed in real time or in an accelerated simulation time scale where one second equals to one minute of real time. The simulation run class of a device also handles the touchscreen input events and update visible information fields. At the start of the simulation, an additional thread for the handling of the graph view is started. All calculations, update methods, redrawing functions, etc. concerning the diagram area are outsourced to this additional thread. Basic thread functionalities (create, cyclic call, etc.) are realized by using implemented Android API classes.

A process started in the background handles the USB (Universal Serial Bus) communication for simulation input parameters. Functions for receiving and sending commands are sequentially called in this process. The functions for handling the USB interface are described in more detail in the next subsection.

B. Hardware Interface

Outputs of the simulation (current power consumption of a virtual device) can be written into a data buffer. The buffer content can be forwarded into a file or provided for external units via the USB-OTG interface of the tablet PC. For the simulation of virtual devices which are able to manage their load consumption automatically also additional simulation input parameters, such as envelope functions, start commands, etc. are required. For the demonstration of the simulator in the GREEN HOME project this commands will be transmitted by a smart energy gateway board. The external interface is realized with a USB to serial conversation chip. For a successful communication between tablet PC and external conversation chip the USB interface of the tablet PC must work as a host device. Finally, for future applications and compatibility with different controller units an interface software layer at the Android system supporting byte transfer is implemented. For test and demonstration purposes, a human readable text based command and data protocol is realized. This allows setting power states, requesting information about the simulation and the manipulation of envelope curves.

V. Conclusion and Future Work

In this paper, the development of a household appliances simulator was described. The simulator is implemented on the freely usable operating system Android. Three classes of virtual appliances are implemented: devices which can be turned on and off, devices with a static load profile, and devices with a dynamic reconfiguration of the load profile. A desired point in time for the end of the device operation can be configured. Depending on the current price constraints for electrical power the activation is delayed. Another management functionality is the forecast of a load profile. Depending on the device's internal parameters and under the consideration of a conventional operation, the load profile can be forecasted for a defined future time window. To be able to couple the simulator with a real world application, a serial communication is implemented. As a summary, it can be said that the implemented simulator is optimized for demonstration purposes and can act as a portable prototyping platform for energy management functions which will be included in future home appliances.

For future work and research, several additions and functional extensions are identified to improve the performance and usability of the simulator application.
A parallel simulation of more than one device can be implemented. The virtual devices can be simulated on one or distributed on several tablet PCs. An appropriate management of the simulation instances (i.e., time synchronization) must be fulfilled. An additional task at running multiple device simulations on a single tablet PC is a performance and benchmarking analysis.

A class of appliances not implemented yet are devices with a continuously load profile as described in Section II. For a future implementation of such appliances, a model of internal electrical drives and control loops must be considered.

Advanced features at the charging devices can be simulated, e.g., the desired charge state of the accumulator at the end of the charging process might be given by the user as opposed to always ”full charge”.

Acknowledgment

This work was supported within the research project Grid Responsive Energy Efficient Networked Home (GREEN HOME) and partially funded by the Austrian government and the City of Vienna within the competence center program COMET.

References

[1] G. Wood and M. Newborough, “Energy-use information transfer for intelligent homes: Enabling energy conversation with central and local displays,” in *Energy and Buildings 39 (2007)*. Edinburgh, Scotland EH14 4AS, UK: School of Engineering and Physical Sciences, Herriot-Watt University, 2 June 2006 2006, pp. 495 – 503.

[2] S. Drenker and A. Kuder, “Nonintrusive monitoring of electric loads,” in *IEEE 0895-0156/99*, October 1999, pp. 47 – 51.

[3] M. Hashmi, S. Hanninen, and K. Maki, “Survey of smart grid concepts, architectures, and technological demonstrations worldwide,” in *Innovative Smart Grid Technologies (ISGT Latin America), 2011 IEEE PES Conference on*, Oct. 2011, pp. 1 –7.

[4] A. Grein and M. Pehnt, “Load management for refrigeration systems: Potentials and barriers,” *Energy Policy*, no. 40, April 2011.

[5] Y. Zong, T. Cronin, O. Gehrke, H. Bindner, J. Hansen, M. Latour, and O. Arcauz, “Application genetic algorithms for load management in refrigerated warehouses with wind power penetration,” in *PowerTech, 2009 IEEE Bucharest*, 28 2009-july 2 2009, pp. 1 –6.

[6] G. Hart, “Nonintrusive appliance load monitoring,” *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870 –1891, Dec 1992.

[7] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong, “Power signature analysis,” *Power and Energy Magazine, IEEE*, vol. 1, no. 2, pp. 56 – 63, Mar-Apr 2003.

Open Source, Simple, Concurrent Simulator for Education and Research

Miguel Bazdresch
ECTET Department
Rochester Institute of Technology
Rochester, NY, USA
Email: miguelb@ieee.org

***Abstract*—In engineering education and training, it is desirable that students develop their own simulators, or inspect the source code, modify it, and design their own functionality based on an existing simulator. Existing commercial and open-source simulator software are not always appropriate for this purpose, given their complexity or their closed-source nature. We propose a simulator that is simple enough to be used for this purpose, while offering powerful features (such as concurrent computation and graphical user interfaces) that make it adequate also for more advanced research work.**

Keywords–Modeling, Computer Simulation, Parallel Processing, Educational Technology

I. Introduction

In engineering education or training, it is often desirable that students build their own simulators, inspect and/or modify the source code of an existing one, or design their own code and insert it into a simulator. It is difficult to accomplish these objectives in the context of an engineering course, with a reasonable time investment on the part of the students. It would be useful to have an educational simulator with the following features:

- Open-source, so that its code may be freely inspected and modified.
- Coded in a simple, dynamic language.
- With high performance.

In this paper, we present a simulator that meets this description. The simulator is similar to Simulink [1] or LabView [2]. It allows simulation of models that may be divided into independent, interconnected blocks. Data and/or events flow among the blocks, of which some are sources, some sinks, and some process their input into an output. It may be useful for discrete simulation, for model-based design, for implementing stencil codes and for simulating other types of distributed models.

The paper is divided as follows. In Section 2, we present an introduction to the Julia language; the simplicity and performance of the proposed simulator are, in large part, due to this language. In the third section, we describe our proposal in some detail. Then, we compare it with other existing simulators offering similar capabilities. We finish the paper by presenting our conclusions and our plans for the future.

II. The Julia Language

Julia [3], [4] is a recent open-source numerical computing language. It offers the simplicity and expressiveness of more well-known dynamic languages such as Python [5] or Matlab [6], with execution times that are frequently no more than two times slower than C or C++ code. In comparison, algorithms coded in Python or Matlab often are thousands of times slower than their equivalents written in C.

Julia achieves this unprecedented (for a dynamic language) execution speed thanks to a “just in time” compilation strategy, based on the Low-Level Virtual Machine (LLVM) [7].

The combination of speed and simplicity is one of Julia’s main attractions; however, it has other features that make it interesting as a simulation programming language: its typing system and its built-in parallel and distributed computing features. Julia’s development is led by a team at the Massachusetts Institute of Technology. Although still in its early stages, it has attracted considerable interest. It is available for the three major operating systems, Windows, Linux and OS X.

A. Julia’s typing system

Julia’s typing system is powerful and flexible. It offers a nominative type hierarchy with bits (numbers), composite (structures) and dictionary types, among others. Types may be used as arguments. The compiler creates efficient code by using run-time, dynamic type inference on variables. Julia allows the option of declaring (annotating) a variable's type, which helps the compiler produce faster code.

In the context of simulation, besides its execution speed benefits, a powerful type system helps to insure that interconnected blocks receive and produce data of the expected types, avoiding run-time failures and coding errors.

B. Julia's parallel and distributed computing environment

Multi-core, distributed, clustered, cloud and otherwise parallel and concurrent computing platforms promise significant gains in simulation speed [8]. While parallel programming is not a trivial task, it can be made more tractable by appropriate support from the programming language. Julia provides a multiprocessing environment based on message passing. Each processor runs its own Julia code on its own memory, and all data sharing is done explicitly by passing messages between CPUs.

Multiprocessing lends itself to simulating a model which can be divided into independent blocks, each of which operates on data produced by other blocks and hands its data to other blocks. Each block runs independently on each processor, and exchanges data with other blocks using message passing. The main drawback of this technique is the overhead involved in passing messages.

Julia also supports automatic compilation for multiprocessing and cluster computing. For instance, arrays may be distributed, in the sense that it is divided into subarrays, each existing on a different CPU's memory. Operations on the array are performed by all CPUs in parallel, behind the scenes and with very little explicit intervention by the user.

III. The proposed simulator

In this section, we describe a simulator that takes advantage of Julia's strengths, and that is sufficiently simple to be adopted in an educational context, while being powerful enough to be useful in research.

Assume that we wish to simulate a model that has been divided into a number of discrete blocks. Each block has several input and/or output *pins*, each of which is connected to one or more pins of other blocks.

In a concurrent simulator, a scheduler assigns blocks to available computing resources according to certain rules. We now describe our proposal in more detail.

A. Blocks

A block is a structure with the following fields:

- A configuration, implemented as a hash table. Each block defines its own configuration and how it is interpreted.
- A *work* function, which processes the inputs to produce the outputs.
- A *stop* function, which is called when the scheduler stops the simulation.
- A *state*, where the block may store its state.
- A set of input and/or output pins, each of a given type.
- A flag that tells the scheduler that the simulation should be stopped.
- A flag that tells the scheduler that the block's work function is currently being executed.

Some configuration items may specify how the block is to be executed and scheduled. For example, if a block opens a file, then the file descriptor will be valid only for the CPU where the file was opened. In such a case, it is required that the block is executed always in the same CPU. As another example, multiprocessing (in contrast to multithreading) makes it easier to manually partition work among the available CPUs. This may be achieved by indicating to the scheduler that a block should only be executed on a set (or subset) of the available processors.

After blocks are defined, they are instantiated. A given block instance has its own configuration and state, which may be different from those of other instances. Once blocks are instantiated, their pins are connected using a simple function. In this way, a dataflow graph is built, with data pipes connecting the blocks.

B. Scheduler

The scheduler executes in a single processor. It traverses the dataflow graph, trying to find a block that meets a set of criteria for execution. Once a block is found, it calls the block's work function on a free CPU. Then, it continues the search.

A block that meets all the following criteria is eligible for execution:

1) The block must not be running on another CPU.
2) If the block specifies execution on a particular processor, then said processor must be free.
3) All of the block's input pins must be connected to pipes that are not empty; likewise, all its output pins must be connected to pipes that are not full.

This scheduler is very simple, but it allocates work in an efficient manner and, if there are more blocks than CPUs, it is able to keep all CPUs busy.

If it is desired, it is very easy to modify the scheduler or create a new one, due to the simplicity of the language and its dynamic nature.

C. Interfaces

The simulator supports two kinds of interfaces. The first is support for graphical user interfaces. These are blocks that represent data graphically on the screen, and also allow the user to interact with the simulator via traditional user interface elements such as buttons or menus. One example would be an oscilloscope block that displays a signal. Another example would be a waterfall display.

The second kind of interface blocks are those that connect to external hardware elements and allow interaction with physical signals or data. This allows the model to be tested with the same kind of data an eventual system implementation would interact with. For instance, we provide source and sink blocks that connect to a computer's sound card, which in this case acts as a general, low-bandwidth signal acquisition and generation system.

IV. Example

We present an example of a simple block. The block takes as input a vector of floating point numbers. Its output is the square of the input's elements. This block has one input pin and one output pin. We need to create a function to instantiate the block:

```
function square_float_inst()
  b=Block({},              # empty dictionary
    x::Vector{Float}->x.*x # work function
    ()->nothing            # stop function
    0,                     # initial state
    [Pin(Vector{Float})]   # input pin
    [Pin(Vector{Float})])  # output pin
  return b
end
```

In this code, `Block()` is a constructor that returns a block with the given arguments, and `Pin()` returns a pin of the given type. Note we can use types as constructor arguments in a very simple way. We use an empty dictionary as configuration, since this block has no configurable options.

In this case, the stop function is an empty function. Using anonymous functions, this is specified as `()->nothing`. The work function is defined also as an anonymous function, but note that in this case we used type annotations to ensure correctness and to help the compiler. For more complex cases, the work function can be defined separately.

The user would create a block instance `squarer` by calling `squarer = square_float_inst()`, and would connect its input pin to an already instantiated source block `source` with `connect(source,1,squarer,1)`. With this code, output pin 1 of the source block is connected to output pin 1 of the squarer block. The function `connect()` verifies that pin types match, and builds the simulation dataflow graph. With the function `run()`, the scheduler launches the simulation.

V. Comparison with similar simulators

Our proposal is similar to tools such as LabView, Simulink and GNU Radio. Of these, only GNU Radio [9] allows code inspection and modification. For this reason, we focus on it for comparison.

On one hand, GNU Radio is a much more mature tool, with many well-tested, high-performance blocks available. It is focused on modeling signal processing, software-defined radio and telecommunications systems. It offers real-time performance in some applications. It has interfaces to many kinds of hardware. It offers multithreading parallelism.

On the other hand, GNU Radio is a complex system. The blocks and scheduler are written in C++ to achieve the required performance, while the dataflow graph is defined using the Python programming language. This mixture of languages, together with other decisions in the simulator's design, introduce some level of complexity. This is exemplified in [10], a tutorial to write a block that squares its input. The block's code consists of more than 200 lines of C++, and its implementation requires the knowledge of concepts such as virtual functions and inheritance. While this complexity also provides flexibility and robustness, it may also preclude the study of GNU Radio in some educational environments. In comparison, the example presented in the previous section achieves essentially the same functionality in less than ten lines of code, and involves much less complex programming concepts.

Our proposed simulator uses a single, simple language (Julia), which offers performance competitive with C++ and C. The language's expressiveness means that inspecting the scheduler's and any block's source code is feasible within the scope of beginner or intermediate engineering courses. This is achieved without sacrificing features; on the contrary, the parallel and distributed computing facilities offered are often superior to the alternatives.

In Table I, we summarize the comparison between our proposal and GNU Radio.

TABLE I
Our proposal compared to GNU Radio

	Proposal	GNU Radio
Mature	No	Yes
Large library of blocks	No	Yes
Single language implementation	Yes	No
Concurrency	Multiprocessing	Multithreading
Distributed computing support	Yes	No
Code complexity	Low	High
Interface to external hardware	Yes	Yes
Graphical user interfaces	Yes	Yes

VI. Conclusion and future work

We have presented a simulator simple enough to be used, inspected and modified by undergraduate engineering students. Even though it is simple, it offers performance that in many cases is much higher than, and at least comparable to, Matlab's, Simulink's, or Python's. This is possible due to its use of the Julia programming language. It also offers advanced multiprocessing and distributed computing support. It supports graphical user interface blocks and interfaces to external hardware.

There is much work to be done in the future:

1) The available block library needs to grow and mature.
2) Blocks that interface to a variety of hardware and computer ports have yet to be developed.
3) We wish to have an interface to blocks written in different languages, from LabView's G to Verilog.
4) The scheduler's performance has to be optimized after being measured under a variety of workloads and with different computing resources.
5) Other scheduling algorithms should be proposed and evaluated.
6) Discrete-event simulation has yet to be implemented. This may be achieved by an appropriate scheduler and by creating block pins of a time-related type.
7) The simulator has to be tried in a classroom environment, to verify its educational value.
8) In order to enable more complex scenarios, a formalism such as DEVS [11] will have to be implemented. A comparison with the large number of DEVS-based frameworks (in multiple languages) will be then possible.

We believe the Julia language, although still in its early stages, has the potential to become the premier numerical computing language. Its speed and flexibility will undoubtedly allow it to enter into computing areas now dominated by other languages, including simulation.

The simulator's source code is available for browsing and download at [12].

References

[1] "Simulink R2012b documentation center," MathWorks, Retrieved: Nov. 2012. [Online]. Available: http://www.mathworks.com/help/simulink/index.html

[2] "Labview technical resources," National Instruments, Retrieved: Nov. 2012. [Online]. Available: http://www.ni.com/labview/technical-resources/

[3] J. Bezanzon, S. Karpinski, V. Shah, and A. Edelman, "Julia: A fast dynamic language for technical computing," in *Lang.NEXT*, Retrieved: Nov. 2012. [Online]. Available: http://julialang.org/images/lang.next.pdf

[4] J. Bezanzon, S. Karpinski, and V. Shah. (Retrieved: Nov. 2012) Julia Project, software and instructions. [Online]. Available: http://julialang.org/

[5] "Python documentation," Python Software Foundation, Retrieved: Nov. 2012. [Online]. Available: http://www.python.org/doc

[6] "Matlab R2012b documentation center," MathWorks, Retrieved: Nov. 2012. [Online]. Available: http://www.mathworks.com/help/matlab/index.html

[7] C. Lattner and V. Adve, "LLVM: a compilation framework for lifelong program analysis transformation," in *Code Generation and Optimization (CGO 2004), International Symposium on*, Mar. 2004, pp. 75–86.

[8] C. Vecchiola, S. Pandey, and R. Buyya, "High-performance cloud computing: A view of scientific applications," in *Pervasive Systems, Algorithms, and Networks (ISPAN 2009), 10th International Symposium on*, Dec. 2009, pp. 4–16.

[9] "GNU Radio Project, software and instructions," Eric Blossom and others, Retrieved: Sept. 2012. [Online]. Available: http://www.gnu.org/software/gnuradio/

[10] E. Blossom. (Retrieved: Sept. 2012) How to write a signal processing block. [Online]. Available: http://www.gnu.org/software/gnuradio/doc/howto-write-a-block.html

[11] B. P. Zeigler, "Hierarchical, modular discrete-event modelling in an object-oriented environment," *Simulation*, no. 5, pp. 219–230, Nov. 1987.

[12] M. Bazdresch. (Retrieved: Nov. 2012) Proposed simulator project, software and instructions. [Online]. Available: https://bitbucket.org/mbaz/chango

A Generic Monte Carlo Simulation Algorithm For The Availability Prediction Of The Devices With Cold Stand-By Units

Ekin Kaya
Department of Chemical Engineering
Hacettepe University
Ankara, Turkey
ekinkaya@hacettepe.edu.tr

Can Pervane
Department of Physical Engineering
Hacettepe University
Ankara, Turkey
canpervane@hacettepe.edu.tr

Haydar Demirhan
Department of Statistics
Hacettepe University
Ankara, Turkey
haydarde@hacettepe.edu.tr

***Abstract*—Simulation is a strong alternative of analytical methods in the area of reliability science. Availability prediction of devices within a defined period by Monte Carlo simulation is an application of the corresponding area. Devices that are connected parallel with their stand-by units are of concern as well. This study gives an outline of a method for generating a generic algorithm that enables the simulation of 'n' parallel devices. The new method is called LSLB (refers to the two main parameters used in the algorithm) method and the originality may be highlighted in twofold: firstly, it enables the engineer easily generate many scenarios for the stand-by device selection strategy when the operating device breaks down and secondly; it helps to generate a generic, short and comprehensible algorithm for the simulation of several devices. In the present paper, application of the method is exemplified to develop an algorithm for a parallel system in which no device has a priority as a main device and stand-alone operation of each device until a break down is considered as the case. Complete implementation in a MATLAB code is also achieved with 2 x 6 optional scenarios for device assignment strategy regarding to the potential difference in mechanical behavior of the devices.**

Keywords-stand-by; parallel; Monte Carlo simulation; generic; availability; LSLB method.

I. INTRODUCTION

Availability may be defined as measure of the ratio of operating time of the system to the sum of operating time of the system plus and down time [1]. There are analytical and Monte Carlo based simulation methods for prediction of availability. Each has some advantages but also shortcomings, some of which are common for both. There are also some suggestions to overcome those issues [1]. However, Monte Carlo simulation is reported to be more advantageous in complex systems availability analysis where some operational issues (different types of redundancy, different types of failure, preventive maintenance, start-up failure, switching, etc.) are taken account and where a flexible modeling algorithm is aimed [2-4]. It is also necessary to mention that the simulation results achieved by the Monte Carlo method are shown to be in accordance with the Markov Process calculations which may be considered as a more conventional method [2][3].

In simulation methods, the probabilistic prediction of the availability of a mechanic system is achieved by using random numbers generated from probabilistic distributions which are used to substitute operation time lengths (without any problem, i.e., full performance when operating) and repair time lengths. Probabilistic evaluation is needed since the failure time of a device is often random. When a limited time period is investigated in terms of a device's availability, device is expected to fail for some times and repaired for a new run. So, random numbers from an appropriate probability distribution are generated several times, until the limiting time is achieved.

There are some applications of the simulation method for availability prediction in the literature [2-4][6-7]. Cristina [2] simulated the operation of two parallel electro pumps that are simultaneously serving. Some availability and reliability indicators that are predicted is presented in the study. Held et al. [3] used a commercial software for simulating an optical network system in which 4 of the 5 units is working simultaneously. One of the most important aspects of that study is the sensitivity analysis held to understand the effect of variation in mean time to failure and repair time of each unit. Variations in the parameters of some units are reported to have significant effect on the overall reliability and availability while that of some units do not. It is also reported that sensitivity analysis would be valuable when there is limited data about the mechanic history of the simulated system [3]. Chowdhury et al. [4] simulated a back-up gas turbine system in which some of the turbines are simultaneously operating while some are waiting to supersede any failed turbine. Start-up probabilities of the units are taken into account and the objective of the study is reported as to determine the optimal number of gas turbines in order to satisfy the reliability criteria of the system.

Availability indicators are also presented in that study. Miao et al. [5] investigated a ring-standby structure in order to assess the corresponding reliability. Ring-standby structure is reported to be an important form of redundancy in which operating parallel units are linked to separate stand-by units, i.e., some units can supersede only some units [5]. Availability assessment is not included in that study.

When the devices (or a collection of devices, i.e., systems) have some stand-by units that are connected in parallel, using Monte Carlo simulation is not an easy process especially for the parallel systems that have more than two units. The more unit the more cumbersome is to think about ‘if’ and ‘else’ commands in programming. On the other hand, the computer codes such as MATLAB or FORTRAN do not work with real time. In programming art, time may be defined as time steps by the user but for a probabilistic simulation, in which the random times are generated iteratively if only the devices are operated, it is not so easy to think about the state of the devices. There is no means to put a clock on each device when programming with MATLAB (etc.). So; determining which device / or devices is under repair and which device / or devices is at stand-by state usually becomes a difficult task in parallel system’s simulation. In addition, when an algorithm is generated for a defined number of devices, it is often hard to use the same algorithm for a different parallel system. A creative method seems to be necessary to think simple and simulate the parallel system in a comprehensible, controllable and short way. Any ability to develop a decision making structure for selecting the next device, when one another breaks down, would be a great chance. This paper will introduce the engineers such an algorithmic structure through the use of the LSLB method.

II. ABOUT THE MODEL

A. *Scope of the Simulation*

In the simulation of parallel systems that are built by parallel connection of stand-by units, there may be two main operational strategy in terms of a decision that whether any device will dominate the operation. Regarding to the economic and / or mechanical considerations, one of the devices may be assigned as a main device and all of other stand-by units are operated just to hold the system on when the main device is under repair. As soon as the main device is ready for operation, stand-by device does not wait until a breakdown and leave the task to the main device. In the second type of operational strategy, neither of the devices has a priority and all may operate as a stand-by unit. Additionally, every operating device runs until breakdown occurs. In both types of strategy, the operating device may do the task alone or with some stand-by units operating simultaneously. If the operation of the system is achieved by one device while the stand-by units are waiting in switch-off condition, that is defined as cold redundancy [3]. Alternatively, the stand-by units may support the system by simultaneously operating with the main device. That is defined as hot redundancy [3]. One more arrangement may be the warm redundancy, i.e. stand-by unit(s) are partly loaded and deteriorate so [3].

In the present study, the suggested method is outlined for the second type of strategy with cold redundancy. A parallel system in which only one device operates with no priority is simulated. It is important to mention that the algorithmic structure would be considerably different for the first type.

Preventive maintenance, logistics, fatigue, capital cost of the renewal and similar mechanic & economic considerations that would effect an availability work are not included in the study. Their inclusion would be definitely important for a strong availability assessment tool but the objective of this study is not to effectively simulate a mechanic system or make the optimum economical decision. Introducing the engineer with the generic use of LSLB (Length Short Length Big) method and its use for configuration of a decision making structure is aimed. However, it is expected by the authors that the integration may be easily welcomed by use of the LSLB algorithm.

B. *Main Assumptions*

Some assumptions and exclusions that would affect the algorithmic structure are given in this section. They are not only for simplification, but also for the definition of the system. Some of the following assumptions are common in the literature for similar studies [2-5] and it is reported that many of them are in accordance with the practical considerations [3].

- Failures are considered to occur randomly.
- Only one device runs the parallel system.
- Any fail of any component inside the units lead to breakdown of the unit. So ‘breakdown’ word refers to the fail of units in the parallel system.
- Units are independent and none of the failures effect or agitate the other.
- Devices start to operate with no delay when assigned. It is assumed that there would be a perfect switching.
- Failures are considered not to change the mechanical behavior of the device. In other words, history has not an influence on the mechanical behavior. That is one of the stand-points to the use of exponential distribution for random number generation.
- For all devices; a repair period starts instantaneously after the breakdown of the device.
- The simulation does not deal with any system having interaction between the simulated system and by so continuous operation of the simulated system is considered.
- Decision making procedures for stand-by device assignment are managed by the user of the code and decisions are not optimization based in terms of cost minimization.

It is necessary to add that the above assumptions should be revised if some additional mechanic data are desired to be included in the simulation, when the system is considered to be shut down periodically (or randomly) for a preventive maintenance, imperfect switching has a possibility, start-up failure of the unit has a possibility, a repaired unit is not as good as new, fail word does not mean complete breakdown of the unit, any fail of one unit (or a system connected in series with the parallel system) significantly effects the failure characteristics of others, repair of the failed unit can't be achieved without interrupting the system's operation, etc...

On the other hand, some mechanical considerations may be easily taken into account by adjusting the utilized distribution functions. There would be no need for algorithmic modifications; for example, if the preventive maintenance is done without interrupting the system, the fair effect of preventive maintenance may be taken into account in the distribution function of mean time to failure. However, the mean value would be increased if the preventive maintenance is added to the simulation. Any reasons for delay in repair procedures may be handled in a similar manner.

C. *Methodology*

The availability prediction is achieved by simulating the trouble-free operation time (MTTF-mean time to failure) and repair time. Since the corresponding time periods show a random behavior, appropriate probability distribution functions are utilized to represent the probabilistic behavior of operation and repair. That approach refers to the 'event simulation' by Monte Carlo method. MATLAB programming is used for complete implementation of the method in a user-friendly computer code.

In the study, it is assumed that the probabilistic distribution of the trouble-free operation and repair times are fairly matched with exponential distribution under some assumptions. That is also in accordance with the literature [2][4][8]. Exponential distribution is adjusted by the mean time data supplied by the site engineer and random numbers generated from that distribution stand for the corresponding times. Each device has it's own characteristic exponential distribution. It is possible to use a inverse-transform method for generating random numbers from the exponential distribution [2][9]. However, considering the ease of use, MATLAB function 'exprnd' is utilized in the study. In addition, it is necessary to mention that the algorithm is not distribution dependent. The user of the method can easily utilize another kind of a distribution, if necessary.

Generating one random number means that the device has operated once until a breakdown. Then, the device assignment algorithm assigns an appropriate stand-by device according to the strategy made at the beginning of the simulation. Concurrently, a random number is generated as well to represent the repair time of the failed device.

This sequential procedure is carried out until the limiting time for the simulation period is lasted. Then the availability is calculated. However, it is necessary to repeat the procedure *'m'* times to obtain the mean value in terms of availability. The resulting average is regarded as the predicted availability.

D. *Essence of the LSLB Method*

The essence of the LSLB method lies in the creation of the following parameters ;

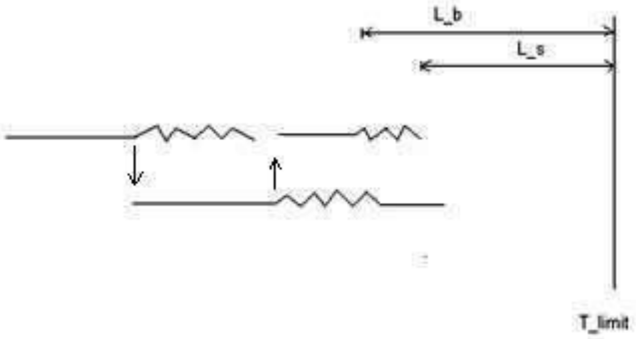


Figure 1. The main parameters used in the LSLB method

Figure 1 is schemed for any time period within the simulation time. The solid lines refer to the operation of a device while the zig-zags for repair. The direction of the rows shows the newly assigned stand-by device. As mentioned before, only one device is running the parallel system. *t_limit* is the overall time period in which the availability prediction is of concern. *L_b* refers to the big length and *L_s* refers to the small length. Sampling for each operation and repair period assigns a new *L_s* and *L_b* value to the corresponding device.

Nature of the method necessitates the iterative definition of the values for *L_s* and *L_b*. If the operating device fails, the decision making structure (see Section II.F) looks for an appropriate device to continue the run and when a device is assigned; *L_s* and *L_b* of that device is updated. The key point is to relate *L_b* and *L_s* of the assigned device with the parameters of the previously operated device; *L_s* of the newly assigned device is defined by using the *L_b* of the lastly operated device. Then by means of the *'q'* index, *L_b* of the operating device is defined.

The index *'i'* refers to the assigned operating device while the index *'q'* is used both for the previously operated device and the newly assigned device such that; just before the new assignment, *'q'* refers to the last operation which is ended by the fail of the device. If a new *'i'* is assigned, that means a stand-by device which have the i^{th} position on the assembly platform is switched-on. Subsequently, *L_s (i)* is defined by using the *L_b (q)*. Then, *q* is increased by one (*q=q+1*) and the *L_b* of the operating device is defined. *L_b* is always indexed as *L_b (q)* and never linked with the index *'i'*. *L_b* is a continuosly growing vector by the sequential increase of the *'q'*. As a result, the last *'q'* value reached at the end of the simulation shows; how many times the stand-by device assignment has been done. In addition, since the assignments are done when a fail occurs, *L_b(q)* is used for the determination of the fail time and fail period, if the fail occurs as a parallel system fail. On the other hand, it is necessary to emphasize that *L_s* parameter of any device is not associated with the previous *L_s* of that device. *L_b (q)* is used for the calculation of the new *L_s (i)* so; *L_s* history of a device is meaningless. However the last values of the *L_s* parameters of each device is used to make 'system_fail' or 'system_on' decision. MATLAB saves the current *L_s* values in a vector (with *n* -total number of devices-elements). New value of the *L_s (i)* changes the i^{th} element of the *L_s* vector inherently. By means of the *'i'* index, vector *'D'* (1xn) is updated too. This vector contains the data of how many times each device is assigned. That data may be valuable when the wearing of the devices is important or when the repair costs are different for the devices.

In the method, 'fail' or 'on' decision is structured on the following basis;

System_fail decision: if the *L_b* of the operating device is larger than all of the other's *L_s*, then the parallel system fails.

System_on decision: if there is any *L_s* bigger than the operating device's *L_b*, then it means that there is a device which is not under repair and ready for operation. The algorithm assigns that device for operation.

E. Structure of the Algorithm

The structure of the simulation algorithm may be summarized in five main categories ;

- Determination of the system, strategy making for the stand-by device assignment and the data input for the average mechanic behavior of the devices
- Generation of the random times (repair time etc.)
- Assessment structure for the 'sytem_fail' or 'sytem_on' decision.
- Repetition of the simulation for *'m'* times
- Processing of the simulation outputs

The whole algorithmic structure is explicitly schemed in the Figure 2.

F. Decision Making Modes for Device Assignment

In both cases (system_fail or not) the decision making structure has some different modes to make the assignment. They are generated by the authors of the present paper. It has been considered that their inclusion would be valuable when the mechanistic research in terms of stand-by device selection is being interested.

The (2 x 6) operating modes that are allowed for user choice in the algorithm :

Modes for system_fail;

1 When the system fails, the rule is to assign the appropriate device that will run the system as soon as possible but; if the number of the closest device is bigger than one, then the least assigned device (least weared) so far is selected. If that number is also not unique, any device among the corresponding ones is assigned randomly.

2 If the number of the closest device is bigger than one, any device among them is assigned randomly.

Modes for system_on;

1 The device that has the max value of actual *L_k* vector is assigned to continue the run of the system. If there is more than one device that has the *max (L_k)*, then the device that has the minimum index number is assigned (MATLAB default function for 'max' evaluation selects the first element in a vector if there is more than one maximum).

There is no mechanical meaning of this mode. It is only for the simplification of the algorithmic structure. However, after a sensitivity analysis, if it is seen that the other modes do not significantly favor the availability, it is unnecessary to punish the computer. Case-based further discussion may be necessary to decide about a decision making strategy for a parallel system's control loop in terms of simplification.

2 Among the selectable devices; assignment is done in a sequence following the index numbers (position on the assembly platform) of the devices. For instance; if the failed device is the second device, the assignment algorithm looks for the availability of the third device and then the fourth device if the third is not available. The sequence proceeds as a cycle of the index numbers.

3 Among the selectable devices; the one that has the maximum mean operating time (trouble-free) is assigned. If there is more than one device that has the same mean, any of them is assigned randomly. Random number generator of the MATLAB from a discrete uniform distribution is used for random assignment throughout the study.

That mode is considered to be more meaningful when there is a periodic maintenance procedure included in the study.

4 Among the selectable devices; the one that has the minimum mean repair time is assigned. If there is more than one device that has the same mean, any of them is assigned randomly

5 Among the selectable devices; the least assigned device is selected for operation. If there is more than one device that served in equal number of times, any one among them is assigned randomly.

In the algorithm, the least assignment refers to the counted assignment number thus far, not associated with the total operation time of that device.

6 Among the selectable devices; any of them is assigned randomly.

That mode is generated just for a comparison with the others. The results of that kind of randomness may be interesting for any researcher.

III. ABOUT THE COMPUTER CODE

All of the work in this study has been held by complete implementation of the algorithm to a MATLAB script. Considering the use of the code by the beginners of reliability science, the interface of the code was tried to be constructed user-friendly. The inputs of the code only necessitates the determination of the simulation period, number of the stand-by devices and their average mechanic data that are expected to be supplied by the site engineer. Since the investigation of device assignment strategy may be the most advantageous utilization area of the suggested

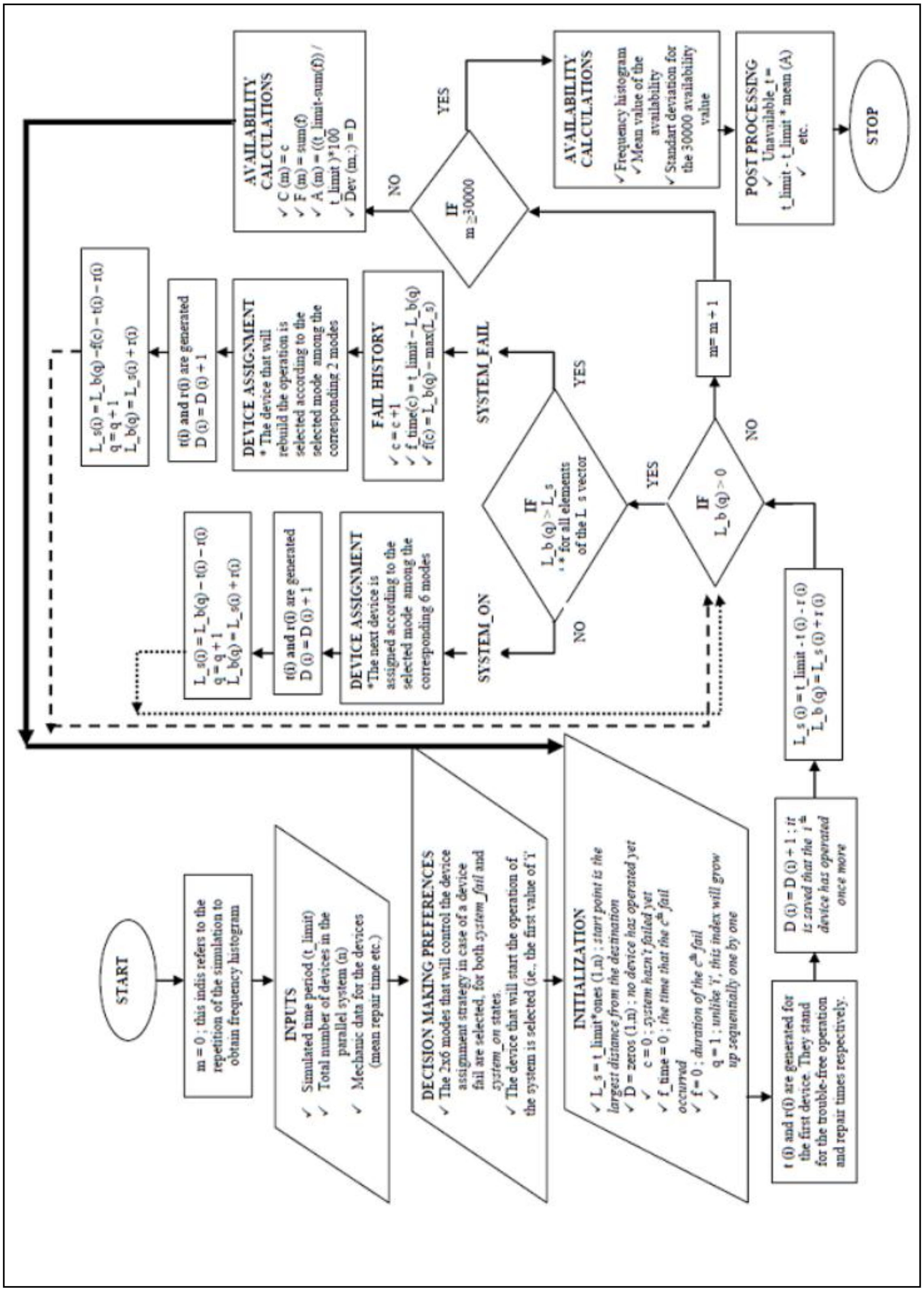

START
m = 0 ; this indis refers to the repetition of the simulation to obtain frequency histogram
INPUTS
✓ Simulated time period (t_limit)
✓ Total number of devices in the parallel system (n)
✓ Mechanic data for the devices (mean repair time etc.)
DECISION MAKING PREFERENCES
✓ The 2x6 modes that will control the device assignment strategy in case of a device fail are selected, for both system_fail and system_on states.
✓ The device that will start the operation of the system is selected (ie., the first value of 'i'
INITIALIZATION
✓ L_s = t_limit*ones (1,n) ; start point is the largest distance from the destination
✓ D = zeros (1,n) ; no device has operated yet
✓ c = 0 ; system hasn't failed yet
✓ f_time = 0 ; the time that the c^th fail occurred
✓ f = 0 ; duration of the c^th fail
✓ q = 1 ; unlike 'i', this index will grow up sequentially one by one
t (i) and r(i) are generated for the first device. They stand for the trouble-free operation and repair times respectively.
D (i) = D (i) + 1 : it is saved that the i th device has operated once more
L_s (i) = t_limit - t (i) - r (i)
L_b (q) = L_s (i) + r (i)
IF L_b (q) > 0
YES
NO
IF L_b (q) > L_s
* for all elements of the L_s vector
NO
YES
SYSTEM_ON
SYSTEM_FAIL
DEVICE ASSIGNMENT
*The next device is assigned according to the selected mode among the corresponding 6 modes
t(i) and r(i) are generated
D (i) = D (i) + 1
L_s(i) = L_b(q) – t(i) – r(i)
q = q + 1
L_b(q) = L_s(i) + r(i)
FAIL HISTORY
✓ c = c + 1
✓ f_time(c) = t_limit – L_b(q)
✓ f(c) = L_b(q) – max(L_s)
DEVICE ASSIGNMENT
* The device that will rebuild the operation is selected according to the selected mode among the corresponding 2 modes
t(i) and r(i) are generated
D (i) = D (i) + 1
L_s(i) = L_b(q) –f(c) – t(i) – r(i)
q = q + 1
L_b(q) = L_s(i) + r(i)
m= m + 1
IF m ≥30000
NO
YES
AVAILABILITY CALCULATIONS
✓ C (m) = c
✓ F (m) = sum(f)
✓ A (m) = ((t_limit-sum(f)) / t_limit)*100
✓ Dev (m,:) = D
AVAILABILITY CALCULATIONS
✓Frequency histogram
✓Mean value of the availability
✓Standart deviation for the 30000 availability value
POST PROCESSING
✓ Unavailable_t = t_limit - t_limit * mean (A)
✓ etc.
STOP

Figure 2. Flowchart of the LSLB based algorithm

algorithm, a decision menu is constructed as a button–type menu inside the code. For each simulation, at the beginning, the user can easily select any strategy for device assignment. Additionally, the code saves all the history within a simulation period. All the operational information (simulated) of a device and the system may be called by the user. For example, at each iteration (m), the system_fail times and their lengths, total time of the system_fails, how many times each device is assigned and the availability calculated are saved. That ability of the code is expected to be helpful when a modification for a further study is necessary.

IV. VERIFICATION OF THE MODEL

The consistency of the algorithm is verified through the simulation of a non-identical two-device cold redundant parallel system. However, prior to the discussion of the verification procedure it is important to highlight some key points about the model;

- In MATLAB, the default random number generator from exponential distribution requires the mean value of the distribution. That parameter is entered as hour based data in the model. That is in accordance with the simulation period which is entered as year data and converted to hour data by the code. One year is considered to be 8766 hours.
- The computer code calls for the mean and generates a distribution function. Then any number from that distribution is introduced as a random number to the simulation. So, mean time for trouble-free operation and mean time for repair are entered to the code to adjust the exponential distribution functions.
- The default uniform number generator of the MATLAB has been compared to the 'twister' algorithm which is also already implemented in MATLAB. A noticeable difference has not been observed.

Mechanic behavior of the devices are considered to not change within the present verification procedure as well as the presented base algorithm in this study. That is no wearing and the resulting variations in probability distribution functions or obsolescence of the device is considered during the simulation. So, selection of the first mode (i.e. if the number of the closest device is bigger than one, least assigned device is selected) in case of a system fail would not have a meaningful result in terms of a mechanistic investigation. If there are more than two stand-by devices, the first mode would randomly result in different availability values according to the device assigned. That is because the characteristics of the devices may be different. However, this randomness corresponds to the second mode as well. In other words, the first and the second mode of the 'system_fail' decision making algorithm become same when wearing effect is not considered in mechanistic investigation.

The six modes for the 'system_on' state would also show no difference in the results in a two-device cold redundant parallel system simulation. That is because; when there are two devices (and if the system is on) the algorithm looks for the available device which is always the other.

Consequently, 1x6 modes are held for the verification procedure which are expected to give the same results for two-device parallel system. No unexpected results have been arised. Same availability is achieved in all of the 6 modes. The two devices of the system are not necessarily identical. This result reveals that the algorithmic structure has no bugs and the model behaves as expected.

Simulation is traditionally repeated *'m'* times to obtain frequency histogram and mean value of the investigated parameter (s). The number that the simulation do not give significantly different outputs is considered to be the appropriate value of *'m'*, i.e. 30000 in this study. This number is not only determined for the two-device system but also for some other cold redundant parallel configurations with more devices. In the literature, some similar *'m'* values may be seen ; 10000 [2], 100000 [3], 30000[4], 10000 [5].

The execution of *'m'* iteration in the developed algorithm lasts only in a few minutes. That may be regarded as an advantageous property in complex system simulation.

V. CONCLUSION

An availability prediction tool for parallel systems is outlined through the use of new LSLB method in the paper. Introducing the engineers with the abilities of this new method is aimed. The suggested method enables the engineer simulate the systems with *'n'* stand-by units operating in cold redundancy state. Although the present simulation has been held under some assumptions, the decision making algorithm for device assignment is constructed in a generic manner and considers many scenerios as if more mechanistic investigation is of concern. The suggested algorithm may be a framework for future mechanistic / economic studies and some modifications on the algorithm may be easily done in LSLB method. As mentioned in the previous sections, the assumptions that simplify the simulation are not restrictive for a potential modification since the logic of the simulation does not rely on specific kinds of probability distribution functions or some functional interactions between the units. All the decisions are stemmed from a kind of geometric parameterization. Decision making algorithm which is consist of 2x6 modes is compatible with any further modifications in order to include more mechanic knowledge to the simulation. However, the presented version of the algorithmic structure is already appropriate for many practical cases.

The ability that is not only saving the operational history of the device but also of the system is one of the main advantages of the method for future mechanistic research. Short simulation times are advantageous in complex system simulation. Additionally, understanding the random behavior of the parallel systems (that may be considered as a common advantage of the simulation based prediction tools) and the ease of investigating device assignment strategy in automatical control loops may be valuable for the researchers.

REFERENCES

[1] S. P. Chambal, “Advancing Reliability, Maintainability And Availability Analysis Through A Robust Simulation Environment”, Dissertation, Arizona State University, USA., 1999.

[2] H. Cristina, ‘‘Reliability Modeling Of A Hydraulic System Using The Monte Carlo Method’’, Annals Of The Oradea University, Fascicle Of Management and Technological Engineering, vol. 7-17, 2008, pp. 328-334.

[3] M. Held and P. M. Nellen, ‘‘Availability Calculation and Simulation Of Optical Network Systems’’, Proc. Of SPIE, vol. 4940, pp. 163-173.

[4] A. A. Chowdhury and D. O. Koval, ‘‘Reliability Assessment Of A Backup Gas Turbine Generation System For A Critical Industry Load Using A Monte Carlo Simulation Model’’, IEEE Transactions On Industry Applications, vol. 45-1, 2009, pp. 310-316.

[5] Q. Miao, X. Zhang, D. Ling, Z. Chen and H. Z. Huang, ‘‘Reliability Assessment Of Ring-Standby Structure Based On Monte Carlo Simulation’’, International Conference On Reliability Maintainability and Safety (ICRMS 2009), July 20-24, Chengdu, China, pp. 1115-1118.

[6] A. C. Marquez, A. S. Heuguedas and B. Iung, “Monte Carlo-Based Assessment Of System Availability-A Case Study For Cogeneration Plants”, Reliability Engineering and System Safety , vol. 88, 2005, pp. 273-289.

[7] K. S. Barland, “Verification Of A Monte Carlo Simulation Method To Find Lower Confidence Limits For The Availability And Reliability Of Maintained Systems”, Thesis, Air Force Institute of Technology, Air University, 1985, Ohio,USA.

[8] R. Dekker and W. Groenendijk, “Availability Assessment Methods And Their Application In Practice”, Microelectron. Reliab., vol. 35, nos 9-10, 1995, pp. 1257-1274.

[9] C. Erdemir and C. Kadılar, “Benzetim Tekniklerine Giris”, Hacettepe University Press, 2003, pp. 59-62.

ISBN: 978-1-61208-234-9

Urban Energy Flow Microsimulation in a Heating Dominated Continental Climate

Diane Perez, Clémentine Vautey, Jérôme Kämpf
Solar Energy and Building Physics Laboratory
École Polytechnique Fédérale de Lausanne (EPFL)
1015 Lausanne, Switzerland
{*diane.perez; clementine.vautey; jerome.kaempf*}*@epfl.ch*

***Abstract*—A new energy management platform has been developed and tested on a 600 building urban zone of the Swiss city of la Chaux-de-Fonds. This simulation framework for energy-efficient systems simulates the buildings' energy demand, infers the production of the energy systems connected and simulate the complete supply and demand side energy flow picture. The platform is designed to use measured consumption data if available, and intended to produce correct estimates of the average energy demands of housing and administrative buildings if not available. This article discusses the first results obtained for the buildings' heating demands and their comparison with measured values. Two examples of energy efficiency scenarios are also presented. They show that the energy savings associated with building insulation can reach more than 12% of the total energy consumption in buildings in the zone studied.**

Keywords- energy flow simulation; simulation and measurement comparison; urban simulation; heating demand simulation; energy demand and supply.

I. Introduction

Despite the raising awareness of the problems linked to the widely unsustainable modern energy use, much improvement is still needed to reduce the resource consumption. Buildings in urban zones can be accounted for a large part of the energy use (20% to 40% of the total final energy consumption in developed countries according to [1]), and the urban population is increasing. It is thus a domain of interest for research to propose innovative energy efficiency measures and energy management tools.

Amongst the large amount of research domains concerned by urban energy simulation [2] [3], there is now a growing interest in the evaluation of the energy need of larger and/or pre-existing urban areas [4] [5], to evaluate the energy performance associated with alternative development or improvement scenarios. However, it remains a challenge today to simulate the detailed energy flow at the scale of a few hundred buildings, including the demand and supply sides [6]. The advantage of this simulation approach is that it allows for the test of scenarios of various detail level, covering in a large part the options available to local politicians and energy departments

As discussed in [7], the amount of data involved in this kind of study, as well as its quality and longevity become important concerns. However, few publications in this domain intend to tackle this problem. The MEU platform presented in this study was developed while keeping this concern in mind. It was designed to provide the simulation functionalities necessary for the energy management of urban zones, accounting for both demand and supply sides, whilst answering some of the data concerns related to this domain.

Section II gives an overview of the structure and simulation method of the MEU platform. Section III describes the model of the case-study neighbourhood, the default values and the origin of the data used. Section IV presents the results, first of the comparison between simulated and measured values, then about two energy efficient scenarios that were studied. Finally, Section V provides a concise conclusion of this paper.

II. The MEU platform

The MEU platform models the urban energy flow as an oriented graph, with (currently) the following rules: *source* nodes (in a broad sense) are linked to *energy system* nodes or *network* nodes. The network nodes themselves are linked to other network nodes or energy system nodes, the energy system nodes in turn linked to other energy system nodes or *building* nodes, seen as energy sink nodes. Measured energy consumption can be associated to any energy system node as its input flow. Energy demands for heating, hot water, electricity and cooling services are associated to each building node. An arbitrary number of connections link each of these demands with the energy system nodes supplying them. Each connection records information about which fraction of the demand is supplied through this connection.

The source nodes actually represent energy in any form: it can be natural gas as well as a pre-defined standard electricity mix. These nodes are defined along with their name by environmental factors, such as a *kWh primary energy per kWh* coefficient and a *kg* $CO_{2,eq}$ *per kWh* coefficient. Network nodes are mainly characterised by a loss factor, whereas energy system nodes refer to a dedicated simulation model including any number of modifiable parameters. Building nodes include data about their address, location (footprint), allocation and a physical model used for the estimation of their heating and cooling demand with the simulation program CitySim presented below. Their

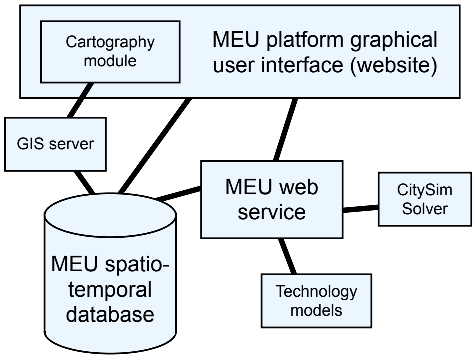

Cartography module
MEU platform graphical user interface (website)
GIS server
MEU web service
CitySim Solver
MEU spatio-temporal database
Technology models

Figure 2. Web-based structure of the MEU platform

electricity and hot water needs are estimated using Swiss norms [8], based on their allocation and treated floor area.

A GIS-based web interface provides access and editing functionalities to the model through a map representation (Figure 1). The data is stored in a spatio-temporal PostgreSQL (opensource) database and can also be accessed directly. The simulation process takes the form of a dedicated web service. It includes calls to a CitySim web service for the estimation of energy demands, and to a technology model web service that simulates the production of an energy system based on its consumption or vice-versa (Figure 2). These are combined in a simulation process consisting in:

- Estimating the whole annual energy flows based on the simulated demands;
- Scaling this first estimate to match the measured consumption values available, saving both original and scaled values for later analyses.

A. CitySim

The urban energy use simulator CitySim [9] was developed at EPFL based on multiple physical models coupled together. CitySim can compute an estimation of the on-site energy use for heating, cooling and lighting with an hourly time step. A radiation model first computes the irradiation incident on each surface of the scene, direct from the sun, diffuse from the sky and reflected by other surfaces. The results of this model, together with predictions of long-wave radiation exchange, are input to a thermal model. This model determinates the thermal exchange through buildings' envelopes and computes the heating and cooling energy needs to maintain predefined temperature conditions inside. Finally, energy systems providing heating, cooling and electricity can also be defined.

As input, a complete physical description of the scene as well as climatic data are needed for the simulation. The climatic data include hourly temperature, wind and irradiance values, together with the geographic coordinates and the definition of far field obstructions (which is used by the radiation model). The building models describe the envelope of each building (the thermal properties of each facade, the layered composition of the walls, the proportions of window and the physical properties of the glazing), as well as the infiltration rate and, when possible, the presence of occupants.

CitySim is used in the MEU platform to compute heating and cooling demands only. It was transformed for this purpose into a web service, called by the central MEU web service. The MEU web service prepares a simple input model based on the data available in the database and save the results simulated by the CitySim web service.

III. Model

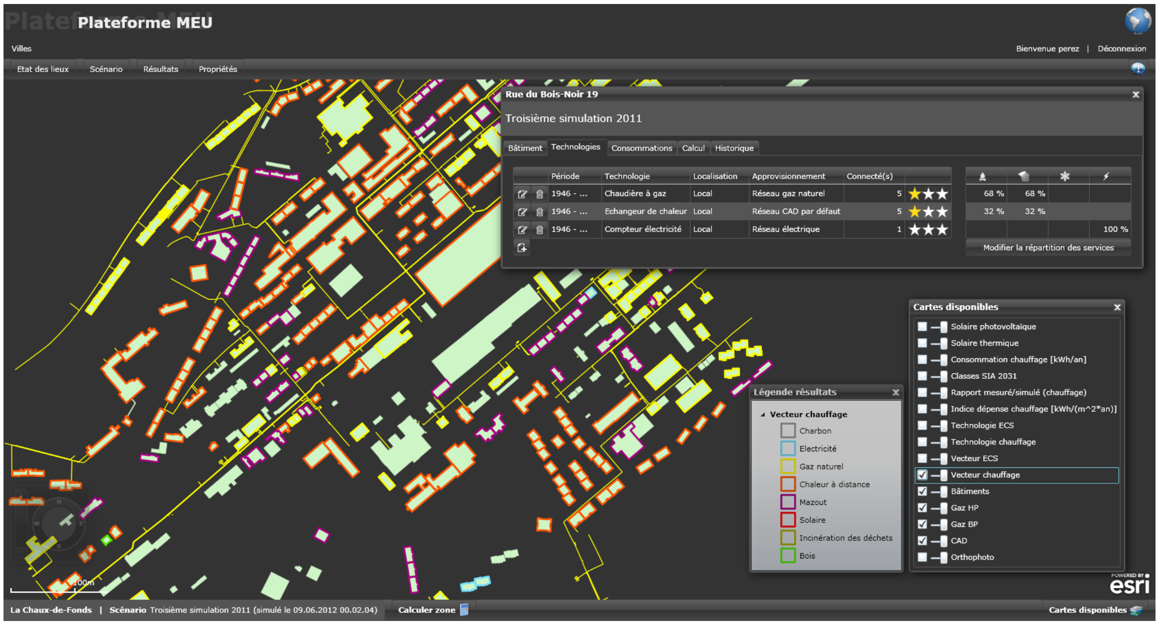
The input model used in this study is composed of both real data and default values, which are presented here.

A. Data sources

The chosen neighbourhood is located in la Chaux-de-Fonds in Switzerland (alt. 1000m), a UNESCO World Heritage Site for its watchmaking industry-driven urbanism mixing housing and workshop at the heart of the city. The model created for this project is based on cadastral data defining buildings' footprints. The footprints are combined with data from the national building register including the address, period of construction, allocation if housing, number of floors and optionally space heating and hot water supply systems. These were completed with a large amount of default data to form the physical model of the building (see Section III-B).

The energy consumption data of gas and heat from the district heating network (DHN) were supplied by the local energy provider for the years 2009, 2010 and 2011. Part of the fuel oil (supplied by various unknown companies) consumption values were provided by the city, based on contacts with building owners. Where measured consumptions of gas, fuel oil or district heating were available respectively gas boilers, fuel oil boilers and heat exchangers were defined in the corresponding buildings. It was assumed that these produce space heating, as well as hot water if the building register announced the same energy carrier for both services. The consumption values were then affected to these systems.

The lower confidence data of the building register was used to complete the supply picture with electrical boilers for domestic hot water in the buildings with measured consumptions but where the register announced electricity as the energy carrier for the production of domestic hot water. The register was also used to create other energy systems for both space heating and domestic hot water when information was available for buildings without consumption data. At this point, buildings without energy system for heating that are semi-detached from buildings with a high consumption value were considered to be heated by the same centralised energy system and thus connected to that energy system. Fuel oil boilers were eventually defined in buildings without



Rue du Bois-Noir 19

Troisième simulation 2011

	Période	Technologie	Localisation	Approvisionnement	Connecté(s)		🔥	🚿	❄	⚡
	1946 - ...	Chaudière à gaz	Local	Réseau gaz naturel	5	★★★	68 %	68 %		
	1946 - ...	Echangeur de chaleur	Local	Réseau CAD par défaut	5	★★★	32 %	32 %		
	1946 - ...	Compteur électricité	Local	Réseau électrique	1	★★★				100 %

Légende résultats

Vecteur chauffage
- □ Charbon
- □ Electricité
- □ Gaz naturel
- □ Chaleur à distance
- □ Mazout
- □ Solaire
- □ Incinération des déchets
- □ Bois

Cartes disponibles
- □ Solaire photovoltaïque
- □ Solaire thermique
- □ Consommation chauffage [kWh/an]
- □ Classes SIA 2031
- □ Rapport mesuré/simulé (chauffage)
- □ Indice dépense chauffage [kWh/(m^2*an)]
- □ Technologie ECS
- □ Technologie chauffage
- □ Vecteur ECS
- ☑ Vecteur chauffage
- ☑ Bâtiments
- ☑ Gaz HP
- ☑ Gaz BP
- ☑ CAD
- □ Orthophoto

La Chaux-de-Fonds | Scénario Troisième simulation 2011 (simulé le 09.06.2012 00.02.04)

Figure 1. The web interface of the MEU platform, here showing the main energy carrier used for heating in each building and the possibility to define any number of energy systems to provide the energy services.

any other energy system. An electricity meter providing electrical services was also defined in each building, and associated with the electricity consumption obtained through the energy provider. The electricity consumption of electrical boilers was assumed to be included in the total electricity consumption measure.

The model is completed with measured local meteorological data for the years 2009, 2010 and 2011, which are used for the CitySim simulation.

B. Default data

As mentioned above, in order to obtain a microsimulation model at this scale of a few hundred buildings, numerous default values where used. The objective of this first crude model is, therefore, not to obtain precise individual building energy demands, but statistically adequate results.

Most of the default physical parameters used for the buildings, shown in Table I, are independent of construction period or allocation, which is an admittedly rough hypothesis. However, the default wall types (Table II) are attributed based on the construction period, and the set point temperature depends on the building's allocation (Table III).

The technology models used for the energy systems are crude; the wood, gas, oil and electrical boilers have an efficiency of respectively 0.65, 0.85, 0.85 and 0.93. The electricity meters have an efficiency of 1 and the heat exchangers of 0.93. The heat pumps have a fixed COP of 3.4.

Facades' windows ratio	0.3
Windows U-value [W/m^{2}K]	1.4
Windows g-value	0.7
Windows openable ratio	0.5
Ground K-value [W/m^{2}K]	3
Roof U-value [W/m^{2}K]	0.3
Short wave reflectance of surfaces	0.4
Infiltration rate [h^{-1}]	0.4

Table I

DEFAULT PHYSICAL PROPERTIES OF THE BUILDINGS

Period	Description	U value [W/m^{2}K]
Before 1945	Rough-stone wall	1.64
Before 1945*	+ inner insulation	0.19
1946 - 1960	Rough-stone, air gap and brick	1.46
1946 - 1960*	+ gap and outer insulation	0.19
1961 - 1970	Double brick wall with air gap	1.14
1961 - 1970*	+ gap and outer insulation	0.19
1971 - 1980	Concrete, ins., reinforced concrete	0.57
1971 - 1980*	+ outer insulation	0.20
1981 - 1990	Insulation and armed concrete	0.42
1991 - 2000	Insulation and armed concrete	0.29
2001 - 2010	Insulation and armed concrete	0.21
1981 - 2010*	+ more outer insulation	0.19

Table II

DEFAULT WALL TYPES, DESCRIBED OUTSIDE TO INSIDE, AND THE INSULATED VERSION (WITH A STAR) USED FOR THE REFURBISHMENT SCENARIO

When two or more energy systems are defined to provide the same service, it is supposed that each meets the same share of the demand, an hypothesis that can be adapted during the simulation based on the measured data.

ISBN: 978-1-61208-234-9

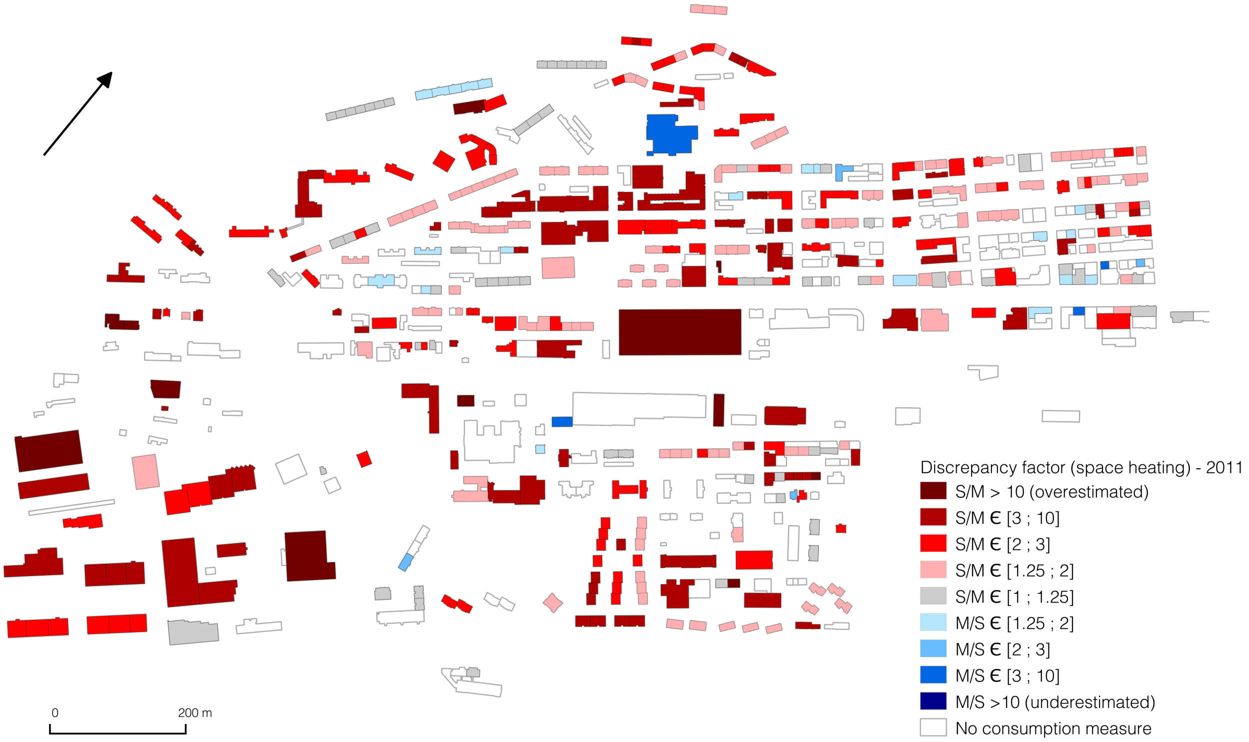


Figure 3. Discrepancy factor for the base case simulation. The ratio is defined with M for the measure and S for the simulated value in order to differentiate over- and underestimation.

Category	Electricity need MJ/(m^2·an)	Hot water need MJ/(m^2·an)	T min set point °C
apartment building	100	75	21
individual home	80	50	21
administrative	80	25	21
schools	40	25	21
sales	120	25	21
restauration	120	200	21
meeting venues	60	50	21
hospitals	100	100	23
industrie	60	25	18
stores	20	5	18
sports installations	20	300	18
indoor swimming pools	200	300	28

Table III
ALLOCATION-DEPENDENT DEFAULT VALUES

IV. RESULTS

The results cover two topics. First, a comparison of the simulated and measured values was performed. The second part of this section presents the results of two energy-efficiency scenarios.

A. Comparison between simulated and measured values

The first results confirmed that the approach described above could correctly represent urban energy flow. The scene includes several cases of centralised energy systems providing space heating and / or hot water in different buildings, buildings where space heating is produced by both the district heating network and a gas boiler (in order to free power on the DHN during heavy load periods), and electricity meters providing both the electricity demand and an electrical boiler, which were correctly simulated.

The map-based representation of the discrepancy between the measured and simulated values allowed for simplified error location and efficient corrections of the model. As the discrepancies might originate from a wrong allocation of the measured consumption as well as from the simulated model itself, a symmetric discrepancy factor $D = \max(\frac{\text{measured value}}{\text{simulated value}}; \frac{\text{simulated value}}{\text{measured value}})$ was used, with values close to one representing a good match. The discrepancy factor over the scene after the correction of the obvious error is shown in Figure 3.

We observe a global over estimation of the buildings' heating demand, with a median discrepancy factor of 2. The most important sources for this difference are probably the following :

- The internal gains have not been considered in the model for now, which definitely results in an over estimation of the demand for heating, although possibly not of that magnitude.

- The numerous default values used for the physical properties of the model might not be appropriate. Once the internal gains are accounted for, it will be possible to test for better adapted default values.
- Some of the measured consumption interpreted as covering hot water as well as space heating might actually correspond only to space heating. As the details about the energy systems installed in the buildings is not known and cannot be obtained without a large survey, this was defined with several assumptions.
- Finally, the default value approach adopted here is not expected to produce very close results for buildings which are not of housing or administrative allocation. The processes taking place in other allocation buildings are too various to be represented with this method, and the platform is intended to be used with real consumption data for these buildings. The discrepancy is also slightly better when excluding these ”non-standard” buildings: the median of the discrepancy factor is at 1.8 over the housing and administrative allocation buildings.

The inclusion of internal gains in the simulation and the improvement of the default values used are thus high priority developments for the future.

B. *Energy efficiency scenarios*

Using the global energy flow picture discussed above, two energy efficiency scenarios where studied. The base case simulation results are known to be of limited quality; however the relative results of two simulations remain valid. The energy flow have been scaled to match the consumption measurements, and this scaling of the simulation results is used to study first scenarios presented here. The future improvements of the model will evidently improve the quality of the scenarios’ results as well.

1) Refurbishment of high consumption building on the DHN: A known intention of the energy providers in la Chaux-de-Fonds and more generally in Switzerland, is to make the DHN denser. The objectives are to increase their efficiency as well as their profitability, while adapting to the decrease of heat demand following the recently prevailing refurbishment policies.

The MEU platform provides a map of the density of final energy used for each service (Figure 4), which allowed to easily spot a high consumption block of 5 buildings (over 400 kWh/m^2 per year). As the heating demand of these buildings was originally underestimated by the platform, the model was adapted for the base case to represent better the low quality of the envelope. The window U-value was set to 3 [W/m^{2}K], the roof U-value to 0.3 [W/m^{2}K] and the infiltration rate to 1.5, leading to a discrepancy factor lower than 1.25. In the scenario, the building was refurbished under the swiss standard Minergie so that the average U-value of the envelope approaches 0.2 [W/m^{2}K] :

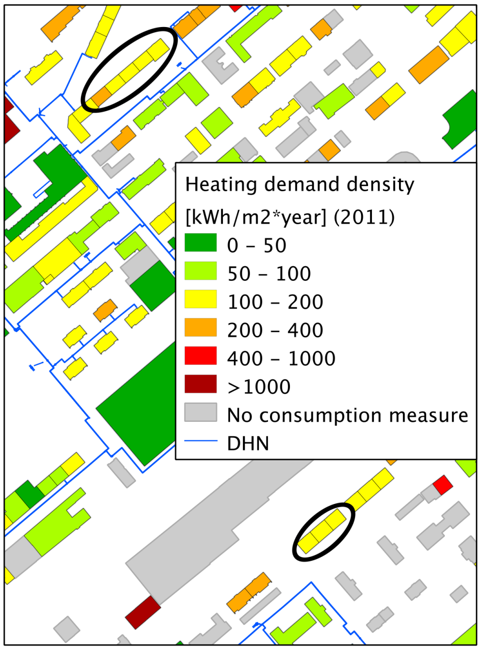


Figure 4. Heating energy use density and buildings that could be connected to the DHN with the energy saving of the first scenario

- The wall type was changed for the insulated version of 1946-1960 (see Table II).
- The window U-value was set to 1 [W/m^{2}K].
- The roof U-value was set to 0.2 [W/m^{2}K].

Applying the same discrepancy factor to scale the simulation results, the scenario leads to a 3.6 GWh decrease of the heating demand, or 65% of the original value. These buildings happen to be heated by the DHN for only 32% of their heat needs (interruptible DHN, completed by a gas boiler). The economy at the DHN level is still of 1.2 GWh per year, which would be sufficient to connect 7 other buildings close to the DHN (shown in Figure 4) and currently heated with oil furnaces.

2) Refurbishment of all buildings in the scene: A second scenario was defined to estimate the possible gains over the whole scene by refurbishing all buildings. The scenario was defined by setting the same window and roof U-values as in the first scenario in every building, and by insulating all facades according to Table II.

This scenario leads to the map of heating energy use density shown in Figure 5, and to an economy of 22.3 GWh of final energy, representing 12.2% of the total final energy used in the scene for the electricity, hot water, space heating and cooling services. All results have been scaled with the discrepancy factor where available. Moreover, the non-inclusion of the internal gains leads to an underestimation of the decrease of heating demand, as these gains become

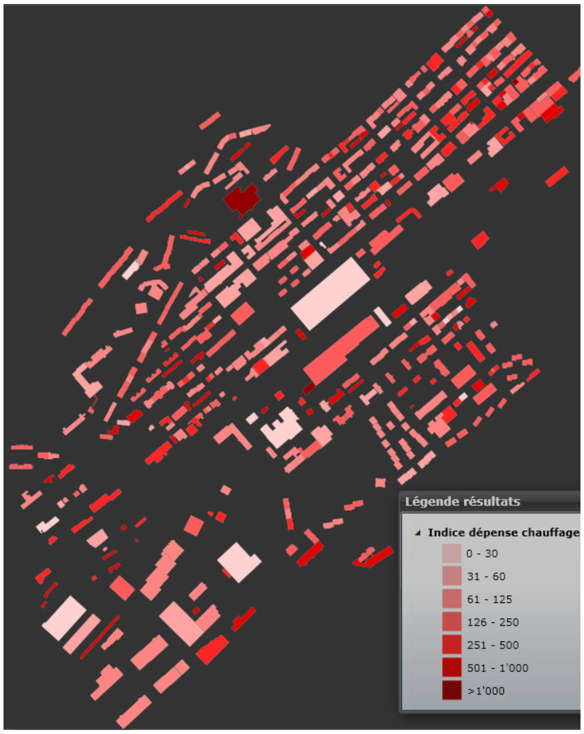


Figure 5. Heating energy use density in kWh/m^2 per year for the second scenario, as shown by the platform MEU.

proportionally more important when the buildings are better insulated.

V. Conclusion and Future Work

This first study performed with the MEU platform is first of all a proof of concept. It demonstrates the possibility to simulate demand and supply side urban energy flows at a large scale and on a building basis. This approach opens a wide panel of possibilities for more specific or detailed studies. The first results show that the current limitations of the MEU platform are not structural, but mostly concern shortcomings of the simulation input model. It requires some adjustments regarding the inclusion of internal gains for the heating demand simulation and the improvement of the default values used. Whereas the simulation of energy demands yet need improvements, a possibility to use them together with measured values was demonstrated.

The first scenario illustrates how the energy supply can be quite easily rationalised. The insulation of 5 low performance buildings liberating enough power on the DHN to connect and heat 7 other buildings close to the network and currently heated with oil furnaces. This is combined with an economy of more than 2.4 GWh of gas, the insulated buildings being heated by both gas and DHN.

The second scenario presented provides an estimate of the energy economy possible on a urban zone by insulating the buildings' envelope. It amounts to more than 12% of the final energy used in buildings (not including transportation). This result supports the suitability of policies encouraging energy efficient refurbishment of buildings as part of the general efforts to reduce the global energy consumption.

Acknowledgment

The authors would like to thank the Swiss Federal Office of Energy (project number 102775) and all funders and partners of the MEU project. The help of the city of La Chaux-de-Fonds and the energy provider Viteos, who provided most of the data used in this project, is gratefully acknowledged.

References

[1] L. Perez-Lombard, J. Ortiz, and C. Pout, "A review on buildings energy consumption information," *Energy and Buildings*, vol. 40, no. 3, pp. 394–398, 2008.

[2] L. Swan and V. Ugursal, "Modeling of end-use energy consumption in the residential sector: A review of modeling techniques," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 8, pp. 1819–1835, 2009.

[3] J. Keirstead, M. Jennings, and A. Sivakumar, "A review of urban energy system models: Approaches, challenges and opportunities," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 6, pp. 3847–3866, 2012.

[4] L. Shorrock and J. Dunster, "The physically-based model BREHOMES and its use in deriving scenarios for the energy use and carbon dioxide emissions of the UK housing stock," *Energy Policy*, vol. 25, no. 12, pp. 1027–1037, 1997.

[5] Y. Shimoda, T. Fujii, T. Morikawa, and M. Mizuno, "Residential end-use energy simulation at city scale," *Building and environment*, vol. 39, no. 8, pp. 959–967, 2004.

[6] J. Snäkin, "An engineering model for heating energy and emission assessment – The case of North Karelia, Finland," *Applied energy*, vol. 67, no. 4, pp. 353–381, 2000.

[7] D. Perez and D. Robinson, *Digital Urban Modeling and Simulation*, ser. Communications in Computer and Information Science. Springer, 2012, vol. 242, ch. Urban energy flow modelling: a data-aware approach.

[8] Société suisse des ingénieurs et des architectes, "Swiss norm 380/1 : L'énergie thermique dans le bâtiment," SIA Zurich, 2009.

[9] D. Robinson, A. Rasheed, J. H. Kämpf, M. Bruse, K. Axhausen, F. Flourentzou, M. Batty, F. Haldi, and D. Perez, *Computer Modelling for Sustainable Urban Design*, D. Robinson, Ed. Earthscan, 2011.

Development of a Neural Network-based Building Model and Application to Geothermal Heat Pumps Predictive Control

Tristan Salque[a,b], Peter Riederer[a]
[a] Energy-Health-Environment Dept.
CSTB (Scientific and Technical Centre for Building)
Sophia-Antipolis, France
tristan.salque@cstb.fr; peter.riederer@cstb.fr

Dominique Marchio[b]
[b] CEP - Centre énergétique et procédés,
CEP/Paris
Mines ParisTech
Paris, France
dominique.marchio@mines-paristech.fr

***Abstract* - The use of artificial neural networks in the field of building energy management has led to remarkable results over the recent years. In this study, the development of room temperature neural network models, to be used for predictive control of geothermal heat pump systems, is discussed. The training process, including the determination of optimal input data, algorithm and structure, is detailed. The prediction performance of the developed neural network is compared to linear ARX models. Simulated data used for training and validation is generated using the TRNSYS environment. The developed model is then implemented into a predictive controller for geothermal heat pumps systems. Simulation results showed that the predictive controller can provide up to 17% energy savings in comparison with conventional controllers.**

Keywords - Artificial neural networks; Room temperature prediction; Predictive control; Energy savings; Geothermal heat pump.

I. INTRODUCTION

This study focuses on the identification of the building thermal behavior with the aim of being used in a predictive control. Three main types of building modeling are usually distinguished. Direct modeling, which assumes that all the characteristics of the building are known a priori, is more appropriate at the design stage of a building. On the other hand, black-box modeling infers description of the building based on observed data only. Gray-box modeling is a combination of the two latter at various degrees. In this article, only black-box models for predictive control of the indoor temperature will be discussed.

Important research was conducted on predictive control strategies and especially on the building model. Linear autoregressive models were first used for prediction by Lute et al [1]. More recently, the use of artificial neural networks (ANN) has significantly increased the prediction performances of models. ANN clearly outperforms linear models for the prediction of room temperature [2, 3]. In our investigation, a number of parameters are different from the existing studies (input parameters, algorithm, performance criteria, prediction horizon, etc.). The main difference lies in the emitter type which is in our case a radiant floor heating. Its high thermal inertia creates a thermal lag that requires a longer prediction horizon. The ANN prediction performances are compared to linear ARX models, which are commonly used for the building model in predictive control.

The developed ANN model is then applied to ground source heat pumps (GSHP) predictive control. With this system, conventional controls often lead to overheating in the afternoon as they do not integrate a prediction of solar gains. On the contrary, predictive control can adjust the heat supply in advance in accordance with a prediction of room temperature and weather data. The operation of the controller is tested by simulation on a residential house. The predictive controller is compared to conventional controllers with respect to energy savings and overheating control.

II. SIMULATION TEST CASE

Simulation data are obtained with the graphically based TRNSYS software. A residential house equipped with a radiant floor heating connected to a geothermal heat pump is simulated during 3 months with a 15 minutes time step. The simulation includes the following components:

- The studied building (Type 56) is the “Mozart house”, which is a 99.8 m2 single-family house of single floor area (Figure 1). The building elements have been chosen to correspond to the current French regulation. The external wall is made of 20 cm of concrete and 8 cm of expanded polystyrene (U-value of 0.42 W.m-2.K-1). The glazing area (U-value of 2.43 W.m-2.K-1) covers 15% of the external surface. This multi-zone building is equipped with a centralized radiant floor heating made of 6 cm of concrete and 6 cm of insulation. The internal gains profile is based on a typical profile for a family of 4 persons.

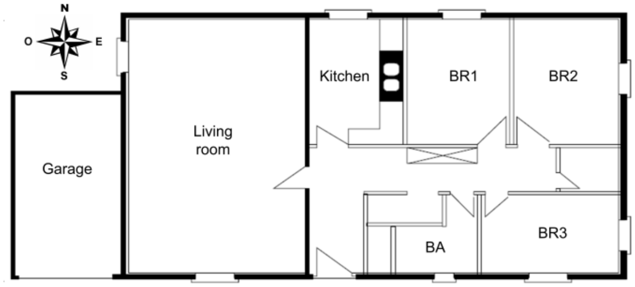


Figure 1: Plan of the "Mozart House" used in simulation

ISBN: 978-1-61208-234-9

- The vertical borehole heat exchanger (Type 557b – Tess Library) is made of one 75 m vertical borehole sized using a calculation spreadsheet developed by Philippe [4]. The soil has typical thermal characteristics (cp= 2000 J.kg-1.K-1 and λ= 2 W.m-1.K-1). The double-U pipes have a fluid to ground resistance of Rb= 0.1 K.m.W-1 ;
- Two single-speed pumps (Type 740 – Tess Library) of nominal mass flow rate 1000 kg/h and nominal electric power 80 W ;
- Outdoor conditions are given by a weather data reader (Type 109) for the city of Nancy in France provided by Meteonorm.

III. MODELING PROCESS

A. Models

In this study, ANN models for room temperature prediction over the next 6 hours are developed and compared to ARX models. Various sets of inputs are tested on both types of models.

For each set of inputs, the optimal ANN architecture (number of hidden layers, number of neurons per layer) is determined via a parametric study. In the present study, one hidden layer was always found to be the best solution. The number of neurons in the hidden layer was first chosen to be equal to 75% of the number of inputs [5] and then optimized by trial-and-error until no improvement could be seen. The hyperbolic tangent sigmoid function was used as the transfer function in the hidden layer.

B. Choice of inputs

Various input parameters influence the indoor environment: outdoor temperature, solar radiation, occupation (internal gains, windows opening, etc.), heating power, wind, humidity, etc. Taking into account all these parameters is not conceivable for two main reasons. First, regarding the application on a real controller, the number of sensors would be too high and some variables are difficult to measure. Second, a more complicated model is more likely to diverge as it is more sensitive to noise in the data. The model has to be as simple as possible while taking into account the most relevant inputs. Among all the meteorological variables, the global horizontal solar radiation and the outdoor temperature are accordingly the most influential parameters for the indoor environment.

For nonlinear models such as ANN, there is no systematic approach [6] and the risk of dismissing relevant inputs is high. Statistical methods like auto-correlation criterion or cross correlation give a good insight into the relevance and the lag effect of an input variable on the output.

C. Training process

The parameters of the ARX-models have been identified using the ordinary least squares method that minimizes the quadratic prediction error criteria. As regards ANN, the algorithm used for training was an optimized version of the Levenberg-Marquardt algorithm that included Bayesian regularization. This algorithm minimizes a combination of squared errors and weights, and then determines the correct combination so as to produce a network that generalizes well. The generalization capability is also improved with the early stopping feature. In this technique, the collected data that was first normalized to the range [-1; 1] is divided into three subsets: training, validation and test. Training stops when validation performance has increased more than 5 times since the last time it decreased. The test data set is used to estimate the generalization error of the ANN, but does not interfere during the training process.

IV. PREDICTION RESULTS

A. Performance criteria

To evaluate the prediction error of ANN and ARX models, the root mean square error (RMSE) and the mean error (ME) were used as performance criteria:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|\hat{T}_i - T_i\right|^2} \qquad (1)$$

$$ME = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{T}_i - T_i\right) \qquad (2)$$

The 95% confidence intervals for one-step-ahead forecasts are approximately equal to the point forecast plus or minus 2 times the RMSE (under the assumption of normality). The ME indicates whether the forecasts are biased, i.e. whether the prediction errors tend to be disproportionately positive or negative.

B. Results and discussion

Three months of simulation were used to train and test the models: January and February data are used for training and validation of ANN and ARX models, while March is used for test. A wide range of inputs were tested, but we chose to show only 4 representative set of inputs (TABLE 1), including the set that gave the best prediction performance. The models provide the room temperature $\hat{T}_i$ for the next time step from weather data (outdoor temperature T_o and global horizontal solar radiation I) as well as previous and current values of heating power P_h and room temperature T_i. Tests not presented here revealed that the mean value of outdoor temperature on the last 24 hours To_{24} contains enough information to describe the dynamic behavior of this specific building. For less insulated buildings, the impact of outdoor temperature is higher and the current value of T_o is likely to be more appropriate.

TABLE 1. INPUTS GIVEN TO ANN AND ARX MODELS FOR ROOM TEMPERATURE PREDICTION

Inputs	ANN1/ARX1	ANN2/ARX2	ANN3/ARX3	ANN4/ARX4
I(k)	x	x	x	x
I(k-1)	x			
$P_h(k)$	x	x	x	x
$P_h(k-1)$	x		x	
$To_{24}(k)$	x	x	x	x
$T_i(k-1)$	x	x	x	x
$T_i(k-2)$	x		x	x

TABLE 2. PREDICTION ERRORS OF ROOM TEMPERATURE OVER A 6 HOUR PREDICTION HORIZON

Prediction horizon	**1 h**		**2 h**		**3 h**		**4 h**		**5 h**		**6 h**		**Total**	
Error [°C]	*ME*	*RMSE*	*ME*	*RMSE*	*ME*	*RMSE*	*ME*	*RMSE*	*ME*	*RMSE*	*ME*	*RMSE*	*ME*	*RMSE*
ANN1	0.02	0.13	0.04	0.25	0.07	0.35	0.10	0.42	0.13	0.49	0.15	0.53	0.52	2.17
ANN2	0.04	0.20	0.08	0.32	0.12	0.42	0.16	0.52	0.20	0.61	0.25	0.69	0.86	2.75
ANN3	0.02	0.12	0.03	0.23	0.06	0.31	0.09	0.39	0.12	0.44	0.14	0.49	0.46	1.99
ANN4	0.02	0.12	0.04	0.31	0.07	0.42	0.10	0.46	0.12	0.48	0.14	0.50	0.48	2.30
ARX1	-0.05	0.41	-0.12	0.53	-0.19	0.66	-0.24	0.74	-0.27	0.79	-0.29	0.80	-1.17	3.92
ARX2	-0.11	0.35	-0.22	0.53	-0.32	0.70	-0.43	0.84	-0.53	0.96	-0.64	1.04	-2.25	4.42
ARX3	-0.05	0.40	-0.12	0.51	-0.19	0.64	-0.23	0.71	-0.26	0.75	-0.27	0.76	-1.11	3.76
ARX4	-0.06	0.40	-0.13	0.52	-0.19	0.64	-0.24	0.72	-0.27	0.76	-0.27	0.76	-1.16	3.80

As the models time step is one hour, the prediction is iterated several times to return up to a 6 hour forecast. TABLE 2 shows the prediction performances of the different models over a prediction horizon from 1 to 6 hours. The following comments can be made:

- ANN models clearly outperform ARX models in terms of ME and RMSE over the whole prediction horizon. The RMSE is in average 40% lower using non-linear ANN models. ANN forecasts are less biased as the ME is smaller in absolute value.
- Too complicated models, such as ANN1 and ARX1, do not give accurate results. On the contrary, ANN2 and ARX2 are too simple to describe the dynamic behavior of the building.
- ANN3 and ARX3 are the most accurate models given both criteria. Previous values of heating power $P_h(k-1)$ as well as room temperature $T_i(k-1)$ and $T_i(k-2)$ must be taken into account due to the high inertia of the building and the floor heating.
- Other tests not presented here showed that taking into account previous values further into the past does not improve the prediction performances of both types of models.

An example of 3 hour prediction results of ANN3 and ARX3 models on a representative week of March is given in Figure 2. ANN model reproduces more accurately the thermal behavior of the building in comparison to the linear ARX model. ANN is in particular much better when the building is subject to strong solar gains (first day of Figure 2).

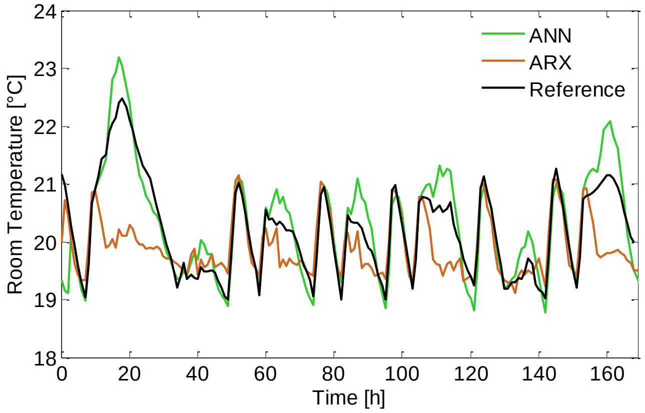


Figure 2. 3 hours prediction of room temperature with ANN3 and ARX3 (March 16-23).

V. APPLICATION TO PREDICTIVE CONTROL

Predictive control can adjust the heat supply in advance in accordance with a prediction of future room temperature and perturbations (solar radiation, outdoor temperature etc.). Important research was conducted on predictive control strategies during the 1980s and 1990s [7-9]. ANN models were successfully applied to the control of residential and small office buildings [10, 11]. The development of predictive controllers for radiant floor heating systems has also led to remarkable results [12-14].

The developed ANN model for room temperature prediction is applied here to single-speed ground source heat pumps (GSHP) predictive control. The concept of the predictive controller, developed by Salque [15], is first introduced. The operation of the controller is then tested by simulation on a residential house and compared to conventional controllers.

A. Concept of the controller

The objective of the controller is to minimize the energy consumption of the GSHP system and maintain a good temperature level anticipating future disturbances and room temperature. The controller is designed to be self learning and easily adaptable in practice. To be compatible with the developed controller, the GSHP system must fulfill the following conditions:

- The GSHP is single-speed (only one single-speed compressor);
- The GSHP only supplies heating and/or cooling (no domestic hot water supply);
- The GSHP is directly connected to the radiant floor heating, without any storage tank for hydraulic decoupling.

1) Controller strucutre

The modular structure of the controller is illustrated in Figure 3.The forecasting modules are all based on ANN. A weather module performs predictions of solar radiation (I) and outdoor temperature (T_o). The heating power produced (P_h) and the electric power consumed by the GSHP (P_{el}) are predicted by another module. The latter uses as inputs the supply and returns temperatures in the boreholes (T_b) and in the radiant floor (T_f), as well as all the possible trajectories of the GSHP on/off for the next 6 hours. The developed ANN model is used for room temperature prediction.

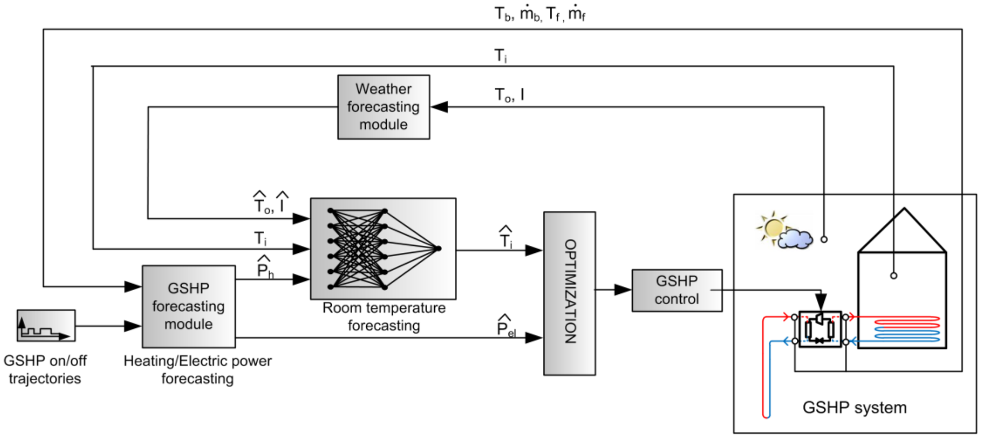


Figure 3. Flow chart of the ANN-based predictive controller. The symbol (^) is assigned to the predicted values.

2) Control strategy

The optimization block determines the optimal trajectory that minimizes the following cost function:

$$J = \sum_{k=1}^{N} \alpha^k \left[\delta(k) \left(\frac{\hat{T}_i(k) - T_r(k)}{\Delta T_{max}} \right)^2 + \frac{\widehat{P_{el}}(k)}{P_{max}} \right] \qquad (3)$$

$$\text{subject to} \quad T_{min} < \hat{T}_i(k) < T_{max} \qquad (4)$$

where $\hat{T}_i(k)$ and $T_r(k)$ are the predicted and the setpoint temperature, while $\widehat{P_{el}}(k)$ and P_{max} are the predicted and the maximum electric power consumed by the GSHP. The maximal distance to the setpoint ΔT_{max} can be adjusted whether the occupants give more importance to comfort or to energy savings ($\Delta T_{max} = 0.5K$ by default). When the building is not occupied, the condition (4) maintains T_i between T_{min} and T_{max}. For intermittent control strategy, $\delta(k)$ is set to one during the occupancy period and to zero otherwise. α is a value between zero and one (typically 0.8) that gives more weight to the first predictions in time, these being usually more accurate than the distant predictions.

3) Prediction horizon

The length of the prediction horizon depends on several factors. A large horizon is needed when large room temperature or electricity price changes are expected in the future [1]. It is the case in an intermittently occupied building. In practice, the horizon length is chosen as an equivalent of the room time constant corresponding to the first active layers of the walls. For the purpose of the present study, a 6 hour receding horizon is applied.

4) Algorithm

At each time step, the optimal on/off trajectory for the next 6 hours is determined. The discrete nature of the input makes it possible to compute all the possible trajectories and chose the one that minimizes the cost function (3) subject to constraint (4). Moreover, it allows the use of non-linear models, such as ANN, that usually limit the possibilities of analytical problem solving [16].

B. Test of the predictive controller

1) Reference controllers

Two conventional controllers are used as a reference: a closed loop controller (CL) on room temperature and a compensated open loop controller (COL) on floor heating supply temperature $T_{f,s.}$ These two control logic are the most frequently observed in single-speed GSHP installations.

The CL controller switches on/off the GSHP when room temperature is beyond the temperature setpoint Tr with a standard 1°C hysteresis. A smaller hysteresis loop can lead to relatively better temperature level but it reduces the compressor lifetime by increasing the number of on/off cycles.

The COL controller is based on the following heating curve that is adjusted with the actual value of room temperature:

$$T_{HC} = (-0.22 \times T_o + 24.5) - (T_i - T_r) \qquad (5)$$

where T_o is the outdoor temperature and $(T_i - T_r)$ the difference between the actual and the setpoint temperature. The COL controller switches on/off the GSHP when the water supply temperature $T_{f,s}$ is beyond T_{HC} ±2°C. The coefficients of the heating curve were finely tuned to optimize room temperature for this particular case. The compensated open loop control logic requires the pump on the building side to always be working to keep the fluid circulating. The two controllers are represented in Figure 4.

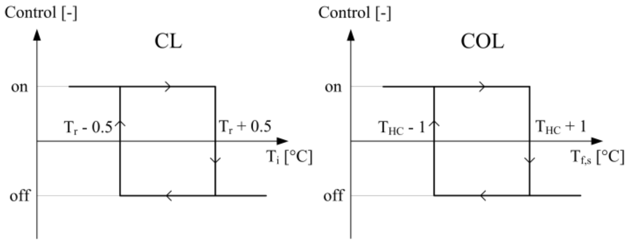


Figure 4. Control logics of the conventional controllers taken as references.

2) *Results and discussion*

For the purpose of the test, the TRNSYS environment that simulates the GSHP system is coupled to the Matlab software where the predictive controller is coded. January and February data are used to train and validate the ANN models while March is used to test the controller. Two conventional controllers presented above are used as reference: a closed loop controller (CL) on room temperature and a compensated open loop controller (COL) on floor heating supply temperature and outdoor temperature. The temperature setpoint is constant and set to 20°C.

A comparison of the controllers on the first day of March is depicted in Figure 5. CL and COL controllers both lead to overshoots in the afternoon. These conventional controllers actually face the same problem: when the GSHP is switched on in the morning of a sunny day, the indoor environment is likely to be overheated in the afternoon. This is of course due to the fact that both control logics do not integrate a prediction of solar gains. The ANN controller stays closer to the setpoint thanks to its prediction capability. A small undershoot is observed just before strong solar gains are expected so that to avoid overheating and benefit from free solar gains, leading to energy savings.

Over the whole month of March, overheating time (Ti>21°C) is reduced by at least 86% with the ANN controller. The COL controller is generally more efficient than the CL in terms of overheating, but the total electrical energy consumed with the COL logic is much higher with the pump at the building side working permanently. The ANN controller ensures a good temperature level with just 4 hours of overheating above 21°C. Total energy savings achieved are 6% and 17% in comparison with CL and COL controllers.

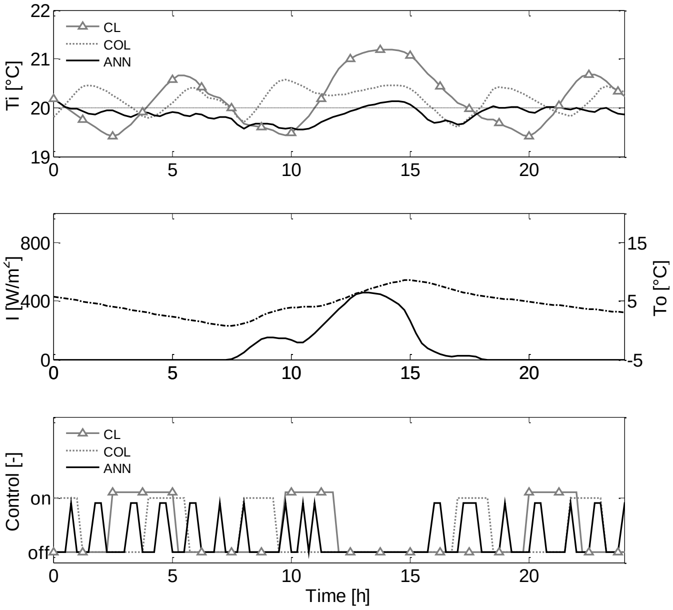


Figure 5. Comparison between predictive controller (ANN) and conventional controllers (CL and COL). From top to bottom : room temperature, weather data (global horizontal solar radiation and outdoor temperature) and GSHP on/off control. March 1st.

VI. Conclusion and future work

In this study, the identification of black-box models for predictive control of room temperature in buildings has been discussed. ANN models were developed and compared to ARX models. Various sets of inputs were tested on both types of models. It was shown that the results obtained with ANN are much better than those obtained with ARX models. RMSE on room temperature prediction over a 6 hour horizon is in average 40% lower with ANN. It was also demonstrated that too complicated models do not give accurate results. The model has to be as simple as possible while taking into account the most relevant inputs. Given the high inertia of the floor heating and the building, previous values of heating power $P_h(k-1)$ as well as room temperature $T_i(k-1)$ and $T_i(k-2)$ must be taken into account in the models.

The developed ANN model was then applied to ground source heat pumps predictive control. The operation of the controller was tested by simulation on a residential house and compared to conventional controllers. Numerical results showed that the predictive controller was able to increase comfort and save energy at the same time. Over the tested month, overheating time was reduced by at least 86% thanks to the prediction capability of the controller and energy savings ranged from 6% to 17% depending on the reference controller.

In a next step, this predictive control algorithm will be implemented as a prototype in a real heat pump system. The real heat pump system has already been monitored for one heating season. The performances using the predictive control algorithm can thus be compared to the classic control.

References

[1] P. Lute and D. van Paassen, "Optimal indoor temperature control using a predictor", IEEE Control Systems, pp. 4-9, 1995.

[2] B. Thomas and M. Soleimani-Mohseni, "Artificial neural network models for indoor temperature prediction: investigations in two buildings", Neural Computing & Applications, vol.16 (1), pp. 81-89, 2007.

[3] G. Mustafaraj, G. Lowry, and J. Chen, "Prediction of room temperature and relative humidity by autoregressive linear and nonlinear neural network models for an open office", Energy and Buildings, vol.43 (6), pp. 1452-1460, 2011.

[4] M. Philippe, M. Bernier, and D. Marchio, "Sizing Calculation Spreadsheet Vertical Geothermal Borefields", ASHRAE Journal, vol.52 (7), pp. 20-28, 2010.

[5] Q.Y. Tang and M.G. Feng, "DPS Data Processing System for Practical Statistics.", Beijing: Science Press, pp. 648, 2002.

[6] T. Chernichow, A. Piras, K. Imhof, P. Caire, Y. Jaccard, B. Dorizzi, and A. Germond, "Short term electric load forecasting with artificial neural networks.", Engineering Intelligent Systems, vol.2 pp. 85-99, 1996.

[7] P. Parent, "Optimal control theory applied to dwelling heating system", IRCOSE, Agence française pour la maitrise de l'énergie, 1987.

[8] J.-C. Visier, V. Paillassa, A. Marti, and M.-H. Foucard, "La commande optimale : un outil d'aide à la définition de stratégies de

gestion", Journées RCT 93 (Régulation - Commande - Télégestion), Sophia-Antipolis, pp. 97-107, 1993.

[9] T.B. Hartman, "Dynamic control : Fundamentals and considerations", ASHREA Transactions 94, vol.1 pp. 599-609, 1988.

[10] P.S. Curtiss, G. Shavit, and K. Kreider, "Neural networks applied to buildings - a tutorial and case studies in prediction and adaptive control", ASHRAE Transactions, vol.102 (1), 1996.

[11] N. Morel, M. Bauer, El-Khoury, and J. Krauss, "Neurobat, a predictive and adaptive heating control system using artificial neural networks", International Journal of Solar Energy, (21), pp. 161-201, 2000.

[12] H. Karlsson and C.-E. Hagentoft, "Application of model based predictive control for water-based floor heating in low energy residential buildings", Building and Environment, vol.46 (3), pp. 556-569, 2011.

[13] C. Verhelst, F. Logist, J. Van Impe, and L. Helsen, "Study of the optimal control problem formulation for modulating air-to-water heat pumps connected to a residential floor heating system", Energy and Buildings, vol.45 (0), pp. 43-53, 2012.

[14] A.A. Argiriou, I. Bellas-Velidis, M. Kummert, and P. André, "A neural network controller for hydronic heating systems of solar buildings", Neural Networks, vol.17 (3), pp. 427-440, 2004.

[15] T. Salque, P. Riederer, and D. Marchio, "Neural predictive control for single-speed ground source heat pumps connected to a floor heating system", Building Services Engineering Research and Technology, (unpublished), 2012.

[16] K.J. Aström and B. Wittenmark, "Computer controlled systems : theory and design", Prentice Hall, 1990.

Importance Sampling for Model Checking of Continuous Time Markov Chains

Benoît Barbot, Serge Haddad, Claudine Picaronny
LSV, ENS Cachan & CNRS & INRIA Cachan, France
$\{$*barbot,haddad,picaronny*$\}$*@lsv.ens-cachan.fr*

***Abstract*—Model checking real time properties on probabilistic systems requires computing transient probabilities on continuous time Markov chains. Beyond numerical analysis ability, a probabilistic framing can only be obtained using simulation. This statistical approach fails when directly applied to the estimation of very small probabilities. Here combining the uniformization technique and extending our previous results, we design a method which applies to continuous time Markov chains and formulas of a timed temporal logic. The corresponding algorithm has been implemented in our tool COSMOS. We present experimentations on a relevant system. Our method produces a reliable confidence interval with respect to classical statistical model checking on rare events.**

***Keywords*-statistical model checking; rare events; importance sampling; coupling; uniformization**

I. INTRODUCTION

Many complex systems exhibit probabilistic behaviour either in an inherent way or through interaction with unreliable environment (communication protocols, biological systems, etc.). Quantitative model checking is an efficient technique to verify properties of these systems. It consists in estimating the probability of a real time property, expressed by some temporal logic formula like in Continuous Stochastic Logic (CSL) [1] as “the probability that the airbag fails to deploy within 10ms is less than 10^{-3}”. This requires to compute transient probabilities on a probabilistic model of the system [2]. Whenever numerical methods cannot be used because of the inherent state explosion, statistical sampling techniques prove to be efficient as soon as it is possible to perform a Monte-Carlo simulation of the model. Simulation usually requires a very small amount of space comparatively, thus allows to deal with huge models [3]. In principle, it only requires to maintain a current state (and some numerical values in case of a non Markovian process). Furthermore no regenerative assumption is required and it is easier to parallelise the methods. Several tools include statistical model checking: COSMOS [4], GREATSPN [5], PRISM [6], UPPAAL [7], YMER [8].

The main drawback of statistical model checking is its inefficiency in dealing with very small probabilities. The size of the sample of simulations required to estimate these small probabilities exceeds achievable capacities. This difficulty is known as the *rare event* problem.

Several methods have been developed to cope with this problem. Of these, the principal is *importance sampling* [9]. Importance sampling method is based on a modification of the underlying probability distribution in such a way that a specific rare event occurs much more frequently. Theoretical results have been obtained for importance sampling but none of them includes any *true confidence interval*. Indeed, all previous works propose *asympotic confidence intervals* based on the central limit theorem. For rare event simulation, such an interval is inappropriate since to be close to a true confidence interval, it is necessary to generate a number of trajectories far beyond the current computational capabilities.

In [10], we proposed an efficient method based on importance sampling to estimate in a reliable way (the first one with a true confidence interval) tiny steady-state probabilities, required for logical formula using a standard “Until” property (aUb) [1], when the model operational semantic is a Discrete Time Markov Chain (DTMC).

Our contribution. We extend here our previous results in order to deal with simultaneous timed and probabilistic assessments: we improve our method to estimate transient probabilities of rare events on Continuous Time Markov Chains (CTMC). More precisely, given a bounded delay τ, we statistically estimate the (tiny) probability that a random path generated by the CTMC reaches a certain state before instant τ. In order to design and prove the correctness of the method we proceed in three stages:

- We show using uniformisation [11] that a confidence interval for the estimation can be computed from confidence intervals of several estimations in the embedded DTMC of the CTMC.
- Our importance sampling approach for time bounded reachability in DTMC is then developped by generalizing the method in [10], based on the mapping of the original model to a reduced one using coupling [12].
- However, contrary to the original approach, the memory requirements are no longer negligible and depend on the considered (discrete) time interval. Thus, we propose three algorithms with a different trade-off between time and space so that very large time intervals can be handled.

As far as we know, our method is the first importance sampling method for CTMC to provide a true confidence interval. Furthermore, we have implemented it in the statistical model checker COSMOS [4]. Experiments with our tool on a classical relevant model show impressive time and/or memory reductions.

Organisation. Section II recalls our previous results [10].

ISBN: 978-1-61208-234-9

Section III extends this method to the estimation of transient probabilities on continuous time Markov chains. Section IV develops algorithmic issues in order to overcome excessive memory consumption. Section V is devoted to the implementation in the tool COSMOS and presents an experimentation on a classical example. Section VI concludes and gives some perspectives to this work.

II. IMPORTANCE SAMPLING METHOD WITH GUARANTEED VARIANCE FOR UNBOUNDED REACHABILITY

To summarize the method developed in [10], first note that the modeller does not usually specify its system with a Markov chain. He rather defines a higher level model $\mathcal{M}$ (a queueing network, a stochastic Petri net, etc.), whose operational semantic is a Markov chain $\mathcal{C}$. If $\mathcal{C}$ is a DTMC with state space S, transition probability matrix $\mathbf{P}$ and two absorbing states s_+ and s_-, which are reached with probability 1, define $\mu(s)$ $s \in S$ as the probability to reach s_+ starting from s. Our goal is to estimate the probability $\mu(s_0)$ with s_0 being the initial state of $\mathcal{C}$.

By generating a large sample of trajectories, the Monte Carlo algorithm provides an estimation of $\mu(s_0)$ as the ratio of the trajectories reaching s_+ by the total number of generated trajectories. For a rare event this approach is not suitable as it is due to the size of the sample far too big when one wants a precise result. The variance of the underling random variable is in fact too big [9].

The importance sampling method uses a modified transition matrix $\mathbf{P}'$ during the generation of paths. $\mathbf{P}'$ must satisfy:

$$\mathbf{P}(s,s') > 0 \Rightarrow \mathbf{P}'(s,s') > 0 \vee s' = s_- \qquad (1)$$

It means that this modification cannot remove transitions that have not s_- as target, but can add new transitions. The method maintains a correction factor called L initialized to 1; this factor represents the *likelihood* of the path. When a path crosses a transition $s \rightarrow s'$ with $s' \neq s_-$, L is updated by $L \leftarrow L\frac{\mathbf{P}(s,s')}{\mathbf{P}'(s,s')}$. When a path reaches s_-, L is set to zero. If $\mathbf{P}' = \mathbf{P}$ (i.e., no modification of the chain), the value of L when the path reaches s^+ (resp. s^-) is 1 (resp. 0).

Let V_s (resp. W_s) be the random variable associated with the final value of L for a path starting in s in the original model (resp. in the modified one). By definition, $\mathbf{E}(V_{s_0}) = \mu(s_0)$. A classical result [9] p. 25, states that $\mathbf{E}(W_{s_0}) = \mathbf{E}(V_{s_0})$.

In the importance sampling method, the challenge is to find a suitable $\mathbf{P}'$. In [10], numerical analysis is performed on an approximation of the chain to produce a suitable matrix $\mathbf{P}'$ having in mind a variance reduction.

We associate with the model $\mathcal{M}$ a smaller one $\mathcal{M}^\bullet$ whose associated DTMC $\mathcal{C}^\bullet$ is a smaller Markov chain with similar attributes $(S^\bullet, \mathbf{P}^\bullet, \mu^\bullet, \ldots)$. The Markov chain $\mathcal{C}^\bullet$ is *reduced* from $\mathcal{C}$ if their exists a *reduction* f, that is a mapping from S to $S^\bullet$ such that $s_-^\bullet = f(s_-)$ and $s_+^\bullet = f(s_+)$. Note that this reduction is designed at the model level. Our method only uses a particular kind of reductions:

Definition 1: Let $\mathcal{C}$ be a DTMC and $\mathcal{C}^\bullet$ reduced from $\mathcal{C}$ by f. $\mathcal{C}^\bullet$ is a *reduction with guaranteed variance* if for all $s \in S$ such that $\mu^\bullet(f(s)) > 0$ one has :

$$\sum_{s' \in S} \mu^\bullet(f(s')) \cdot \mathbf{P}(s,s') \leq \mu^\bullet(f(s)) \qquad (2)$$

Fortunately, the function $\mu^\bullet$ does not have to be computed in order to check that $\mathcal{C}^\bullet$ is a reduction with guaranteed variance. In [10], a structural requirement using coupling theory is brought out to ensure that these hypotheses are fulfilled. This requirement to this context was extended in [13].

We can now construct an efficient important sampling based on a reduced chain with guaranteed variance.

Proposition 1: Let $\mathcal{C}$ be a DTMC and $\mathcal{C}^\bullet$ be a reduction with guaranteed variance by f. Let $\mathbf{P}'$ be defined by:

- if $\mu^\bullet(f(s)) = 0$ then for all $s' \in S$, $\mathbf{P}'(s,s') = \mathbf{P}(s,s')$
- if $\mu^\bullet(f(s)) > 0$ then for all $s' \in S \setminus \{s_-\}$,
  $\mathbf{P}'(s,s') = \frac{\mu^\bullet(f(s'))}{\mu^\bullet(f(s))}\mathbf{P}(s,s')$ and
  $\mathbf{P}'(s,s_-) = 1 - \sum_{s' \in S} \frac{\mu^\bullet(f(s'))}{\mu^\bullet(f(s))}\mathbf{P}(s,s')$.

The importance sampling based on matrix $\mathbf{P}'$ has the following properties:

- For all s such that $\mu(s) > 0$,
  W_s is a random variable taking values in $\{0, \mu^\bullet(f(s))\}$.
- $\mu(s) \leq \mu^\bullet(f(s))$ and $\mathbf{V}(W_s) = \mu(s)\mu^\bullet(f(s)) - \mu^2(s)$.
- One can compute a confidence interval for this importance sampling.

Let us now describe the full method:

1) Specify a model $\mathcal{M}^\bullet$ with associated DTMC $\mathcal{C}^\bullet$, and a reduction function f satisfying hypotheses of proposition 1.
2) Compute function $\mu^\bullet$ with a numerical model checker applied on $\mathcal{M}^\bullet$.
3) Compute $\mu(s_0)$ with a statistical model checker applied on $\mathcal{M}$ using the importance sampling of proposition 1.

III. EXTENSION TO BOUNDED REACHABILIY

We now want to apply the previously defined method to estimate bounded reachability probabilities. We extend it to bounded reachability in DTMC and then to CTMC.

A. Bounded Reachability in DTMC

Given a finite integer horizon u, $\mu_u(s)$ denote the probability to reach s_+ from s in u steps. The goal now is to estimate $\mu_u(s_0)$.

Adding a countdown timer, we define a new Markov chain $\mathcal{C}_u$ whose state space is $(S \backslash \{s_-, s_+\}) \times [1,u] \cup \{s_-, s_+\}$. The timer is initialized to u. Except from the two absorbing states s_+ and s_-, all transitions decrease this timer by one. All trajectories of length u not ending in s_+ are sent by means of their last transition into the sink state s_-. Therefore, the probability to reach s_+ in $\mathcal{C}$ in at most u steps is equal to the probability to reach s_+ in C_u.

Theoretically, this allows the use of the method described in the previous section in the bounded reachability context. In practice, the size of $\mathcal{C}_u$, which is u times the size of $\mathcal{C}$ often make the direct computation intractable. In the following, we describe several algorithms bypassing this problem.

B. Bounded Reachability in CTMC

In a continuous time Markov chain, each state s is equipped with an exit rate λ_s. The waiting time in each state s is then distributed according to an exponential law of parameter λ_s.

To apply our method in the continuous setting the standard method of uniformization can be used. Uniformization reduces the problem of bounded reachability in a CTMC to some problems of bounded reachability in the embedded DTMC.

A chain is said to be uniform when the rate $\lambda = \lambda_s$ is independent from s. Given a uniform chain, the probability $\mu_\tau(s)$ to reach the state s_+ in τ time is equal to:

$$\mu_\tau(s) = \sum_{n \geq 0} \frac{e^{-\lambda\tau}(\lambda\tau)^n}{n!} \mu_n(s)$$

Indeed using the uniform hypothesis, $\frac{e^{-\lambda\tau}(\lambda\tau)^n}{n!}$ is the probability that n transitions take place in interval $[0, \tau]$.

Given a non uniform chain with bounded rates, it is routine to transform it in a uniform chain with the same distribution [11] . It consists in selecting some upper bound of the rates (say λ), consider λ as the uniform transition rate and set a transition matrix $\mathbf{P}_u$ defined by:

$$\begin{aligned} \forall s \neq s' \in S \; \mathbf{P}_u(s, s') &= \tfrac{\lambda_s}{\lambda} \mathbf{P}_u(s, s') \\ \mathbf{P}_u(s, s) &= 1 - \textstyle\sum_{s' \neq s} \mathbf{P}_u(s, s') \end{aligned}$$

This value can be evaluated by truncating the infinite sum. The Fox-Glynn algorithm [14] allows the computation of left (n^-) and right (n^+) truncation points given an error threshold. The errors made by this truncation have to be added to the confidence interval. We obtain a precise formulation of a true confidence interval combining errors from the statistical simulation and from truncation in Fox-Glynn algorithm. For details, see the research report [13]. Then terms $\mu_n(s)$ are estimated using the previously defined method.

IV. Algorithmic Considerations

Based on the previous developments, we describe a methodology to perform statistical model checking using importance sampling to estimate the tiny probability $\mu_\tau(s_0)$ to reach the state s_+ in time less than τ in several steps.

1) Specify a reduced a model $\mathcal{M}^\bullet$ whose embedded DTMC $\mathcal{C}^\bullet$ is a reduction with guaranteed variance.
2) Fix some uniform rate λ for the uniformization of $\mathcal{C}$. Compute left and right truncation points n^-, n^+ for the desired error threshold. Then compute for each n between n^- and n^+ the coefficient $\frac{e^{-\lambda\tau}(\lambda\tau)^n}{n!}$.
3) Compute the distributions $\{\mu_n^\bullet\}_{0<n\leq n^+}$ (numerical computations of the iterated power of the transition matrix on $\mathcal{C}^\bullet$).
4) Use these distributions to perform importance sampling on the simulation of the initial model in order to estimate $\mu_u(s)$ for $n^- \leq u \leq n^+$. Generate a large sample of trajectories using the transition system corresponding to matrix P'_u obtained by applying proposition 1 to the DTMC $\mathcal{C}_u$; compute along each path the likelihood L in order to obtain an estimation with accurate confidence interval.
5) Deduce from these confidence intervals the final confidence interval.

The first step requires some understanding of the system to design an appropriate reduced chain. Steps 2 and 3 only require standard computations on finite Markov chains. Step 5 is obtained by weighting with the Poisson probabilities confidence intervals obtained in step 4 and combining them with the numerical error produced by the Fox-Glynn algorithm; see [13] for a precise formulation. We now detail step 4 since it rises algorithmic problems.

Let m denote the number of states of the Markov chain $\mathcal{C}^\bullet$ and d denote the maximum of outdegrees of vertices of $\mathcal{C}^\bullet$. Let us remark that in typical modellings, d is very small compared to m. A simulation takes at most u steps going through states $(s_u, u), \ldots, (s_1, 1), s_\pm$ where $s_u = s_0$ and $s_\pm \in \{s_+, s_-\}$. In state (s_v, v), we compute the distribution $P'_u((s_v, v), -)$ (cf. proposition 1), which requires the values of $\mu_v^\bullet(f(s))$ and $\mu_{v-1}^\bullet(f(s'))$, for each possible target state s' from s_v.

Vectors $\{\mu_v^\bullet\}_{0<v\leq u}$ may be computed iteratively one from the other with complexity $\Theta(mdu)$: Precisely, define $\widetilde{\mathbf{P}}^\bullet$ as the substochastic matrix obtained from $\mathbf{P}^\bullet$ by removing state s_- and $\mu_0^\bullet$ as the null vector except for $\mu_0^\bullet(s_+) = 1$; then $\mu_v^\bullet = \widetilde{\mathbf{P}}^\bullet \cdot \mu_{v-1}^\bullet$. But for large values of u, the space complexity to store them becomes intractable and the challenge is to obtain a space-time trade-off. So we propose three methods. The methods consist of a precomputation stage and a simulation stage. Their difference lies in the information stored during the first stage and the additional numerical computations during the second stage. In the precomputation, each method computes iteratively the u vectors $\mu_v^\bullet = (\tilde{\mathbf{P}}^\bullet)^v(\mu_0^\bullet)$ for v from 1 to u.

1) First method is the “natural” implementation. It consists in storing all these vectors during the precomputation stage and then proceeding to the simulation without any additional numerical computations. The storage of

vectors $\{\mu^\bullet_v\}_{v \leq u}$ is the main memory requirement.

2) Let $l(< u)$ be an integer. In the precomputation stage, the second method only stores the $\lfloor \frac{u}{l} \rfloor + 1$ vectors $\mu^\bullet_\tau$ with τ multiple of l in list Ls and $\mu^\bullet_{l\lfloor \frac{u}{l} \rfloor + 1}, \ldots, \mu^\bullet_u$ in list K (see the precomputation stage of algorithm 2). During the simulation stage, in a state (s, τ), with $\tau = ml$, the vector $\mu^\bullet_{\tau-1}$ is present neither in Ls nor in K. So the method uses the vector $\mu^\bullet_{l(m-1)}$ stored in Ls to compute iteratively all vectors $\mu^\bullet_{l(m-1)+i} = P^{\bullet i}(\mu^\bullet_{l(m-1)})$ for i from 1 to $l - 1$ and store them in K (see the step computation stage of algorithm 2). Then it proceeds to l consecutive steps of simulation without anymore computations. We choose l close to $\sqrt{u}$ in order to minimize the space complexity of such a factorization of steps.

3) Let $k = \lfloor \log_2(u) \rfloor + 1$. In the precomputation stage, the third method only stores $k+1$ vectors in Ls. More precisely, initially using the binary decomposition of u ($u = \sum_{i=0}^k a_{u,i} 2^i$), the list Ls of $k + 1$ vectors consists of $w_{i,v} = \mu^\bullet_{\sum_{j=i}^k a_{v,j} 2^j}$, for all $1 \leq i \leq k + 1$ (see the precomputation step of algorithm 3). During the simulation stage in a state (s, v), with the binary decomposition of v ($v = \sum_{i=0}^k a_{v,i} 2^i$), the list Ls consists of $w_{i,v} = \mu^\bullet_{\sum_{j=i}^k a_{v,j} 2^j}$, for all $1 \leq i \leq k + 1$. Observe that the first vector $w_{1,v}$ is equal to $\mu^\bullet_v$. We obtain $\mu^\bullet_{v-1}$ by updating Ls according to $v - 1$. Let us describe the updating of the list performed by the step-computation of algorithm 3. Let i_0 be the smallest index such that $a_{v,i_0} = 1$. Then for $i > i_0$, $a_{v-1,i} = a_{v,i}$, $a_{v-1,i_0} = 0$ and for $i < i_0$, $a_{v-1,i} = 1$. The new list Ls is then obtained as follows. For $i > i_0$ $w_{i,v-1} = w_{i,v}$, $w_{i_0,v-1} = w_{i_0-1,v}$. Then the vectors for $i_0 < i$, the vectors $w_{i,v-1}$ are stored along iterated $2^{i_0-1} - 1$ matrix-vector products starting from vector $w_{i_0,v-1}$: $w(j, v - 1) = P_0^{\bullet 2^j} w(j + 1, v - 1)$. The computation associated with v requires $1 + 2 + \cdots + 2^{i_0-1}$ products matrix-vector , i.e., $\Theta(md2^{i_0})$. Noting that the bit i is reset at most $m2^{-i}$ times, the complexity of the whole computation is $\sum_{i=1}^k 2^{k-i}\Theta(md2^i) = \Theta(mdu \log(u))$.

The three methods are numbered according to their decreasing space complexity. The corresponding space-time trade-off is summarized by Table I, where the space unit is the storage of a float.

V. Experimentation

A. Implementation

Tools. Our experiments have been performed on Cosmos, a statistical model checker whose input model is a stochastic Petri net [15] with general distributions and formulas are expressed by the logic HASL [4]. We have also used the model checker Prism for comparisons with our method. All the experiments have been performed on a computer with twelve 2.6Ghz processors and 48G of memory.

Algorithm 2:

```
Precomputation($u, \mu^\bullet_0, P^\bullet_0$)
Result: $Ls, K$
// List $Ls$ fulfills $Ls(i) = \mu^\bullet_{i \cdot l}$
$l \leftarrow \lfloor \sqrt{u} \rfloor$
$w \leftarrow \mu^\bullet_0$
for $i$ from 1 to $\lfloor \frac{u}{l} \rfloor l$ do
    $w \leftarrow P^\bullet_0 w$
    if $i \mod l = 0$ then
        $Ls(\frac{i}{l}) \leftarrow w$
// List $K$ contains $\mu^\bullet_{\lfloor \frac{u}{l} \rfloor l+1}, \ldots, \mu^\bullet_u$
for $i$ from $\lfloor \frac{u}{l} \rfloor l + 1$ to $u$ do
    $w \leftarrow P^\bullet_0 w$
    $K(i \mod l) \leftarrow w$

Stepcomputation($v, l, P^\bullet_0, K, Ls$)
// Updates $K$ when needed
if $v \mod l = 0$ then
    $w \leftarrow Ls(\frac{v}{l} - 1)$
    for $i$ from $(\frac{v}{l} - 1)l + 1$ to $v - 1$ do
        $w \leftarrow P^\bullet_0 w$
        $K(i \mod l) \leftarrow w$
```

Algorithm 3:

```
Precomputation($u, \mu^\bullet_0, P^\bullet_0$)
Result: Ls
// $Ls$ fulfills $Ls(i) = \mu^\bullet_{\sum_{j=i}^k a_{u,j} 2^j}$
$k \leftarrow \lfloor \log_2(u) \rfloor + 1$
$v \leftarrow \mu^\bullet_0$
$Ls(k + 1) \leftarrow v$
for $i$ from $k$ downto 0 do
    if $a_{u,i} = 1$ then
        for $j$ from 1 to $2^i$ do
            $w \leftarrow P^\bullet_0 w$
    $Ls(i) \leftarrow w$

Stepcomputation($v, l, P^\bullet_0, Ls$)
// Ls is updated accordingly to $v - 1$
$i_0 \leftarrow \min(i \mid a_{v,i} = 1)$
$w \leftarrow Ls(i_0 + 1)$
$Ls(i_0) \leftarrow v$
for $i$ from $i_0 - 1$ downto 0 do
    for $j = 1$ to $2^i$ do
        $w \leftarrow P^\bullet_0 w$
    $Ls(i) \leftarrow w$
```

Copyright (c) IARIA, 2012.   ISBN: 978-1-61208-234-9

Table I
COMPARED COMPLEXITIES

Complexity	Method 1	Method 2	Method 3
Space	mu	$2m\sqrt{u}$	$m \log u$
Time for the precomputation	$\Theta(mdu)$	$\Theta(mdu)$	$\Theta(mdu)$
Additional time for the simulation	0	$\Theta(mdu)$	$\Theta(mdu \log(u))$

Adaptation of COSMOS. In addition to the implementation of our algorithms, two main modifications on the tool had to be performed in order to integrate our method. First, a freely available implementation of the Fox-Glynn algorithm [16] was added in order to compute probabilities from Poisson distributions. Second, COSMOS sequentially generates a batch of trajectories. In our context, this is highly inefficient since the numerical computations of $\mu_n^\bullet$ required by algorithms 1 and 2 should be repeated for every trajectory. So, one generates a bunch of trajectories in parallel step by step. Different sizes of bunches are possible but they cannot exceed the size required for the numerical computations. Based on the asymptotic time and space cost of these computations, we handle m^2 trajectories.

B. Global Overflow in Tandem Queues

Let us present an experimentation on tandem queues. This example is a classical benchmark for importance sampling. It has also practical interest as a standard modeling of networks [17]. Such a modeling allows to accurately dimension a network for a given loadwork.

Specification. We consider a system of k queues in serie. A client arrives in the first queue with rate ρ_0. In queue i ($i < k$), a client is served with rate ρ_i and then go to the next queue. In the last queue, clients leave the system with rate ρ_k. For this model, we can construct a reduced one by bounding the number of clients except in the first queue by a parameter R. A suitable coupling relation can be established in order to ensure the hypotheses of definition 1 as described in [13]. We are interested in estimating the probability for the system to overflow i.e., there is more than $H = 50$ clients in the whole system before being empty in less than $\tau = 100$ time units.

Choice of parameters. We choose the parameters of the system as follows. $\rho_0 = 0.25$ and for all $1 \leq i$ $\rho_i = 0.375$. We study the behaviour of the methods for different values of k. We have chosen for the reduced model $R = 5$ as we experimentally found that this value of R yields a tight confidence interval. We generated 1000 simulations to estimate every $\mu_n(s_0)$ with a confidence level for the simulation of 10^{-6}.

Fox-Glynn algorithm. We plotted in figure 1 the curves $\mu_n(s)$, $\frac{e^{-\lambda}\lambda^n}{n!}$ and $\frac{e^{-\lambda}\lambda^n}{n!}\mu_n(s)$ for the tandem queues with two queues and $\lambda = 100$ with logarithmic scale. The quantity, which we estimate, is $\sum_{n=0}^{\infty} \frac{e^{-\lambda}\lambda^n}{n!}\mu_n(s)$. We observe that for $n < 50$, $\mu_n(s_0) = 0$ whereas the Poisson probability for such a n is not null. Therefore, a left truncation of $n^- = 50$ on the Fox-Glynn does not produce any error. On the right part of the Poisson distribution, after the maximum ($n = 100$), the curve decreases while the curve of μ_n increases. Thus the maximum of the product is shifted to the right compared to the maximum of the Poisson probabilities. In order to get a confidence interval of $10^{-1}\mu_\tau(s_0)$, a big enough right truncation index is required. We choose a right truncation on the index $n^+ = 206$ in order to bound the error by 10^{-10} in the Fox-Glynn algorithm.

Analysis of confidence interval. Results are collected with respective time and space consumption for the three algorithms and PRISM in table II. We also computed the value μ with a confidence level of 0.001 estimated with method described in [10]. The overall confidence level is then equal to $(206-50)\times10^{-6}+0.001 = 156{\cdot}10^{-6}+0.001 = 0.001156$ using formula (2) from [13]. In all experiments, the width of the confidence interval is ten times smaller than the estimated value. Moreover, when the numerical computation terminates, the result belongs to the confidence interval. With our choice of truncation indices, the contribution of the right truncation of the Poisson distribution to the length of the confidence interval is several magnitude orders less than the contribution associated with the statistical estimations. So in order to reduce this length, the number of simulations should be increased and not the truncation index n^+.

Analysis of numerical and statistical PRISM. Our method is compared to numerical and statistical model checking done by PRISM. Due to the rarity of the considered event, the statistical approach always fails returning 0. We observe that for small models ($k \leq 4$), PRISM numerical model checker is faster and uses less memory than COSMOS. For $k = 5$, our method is 10 times faster and uses up to 28 times less memory. For $k \geq 6$, PRISM crashes due to a lack of memory.

Comparison of the three methods. While the empirical storage behaviour of the three methods follows the theoretical study, memory does not constitute a bottleneck until $k = 8$. For this value, memory required by method 1 is too important. In order for method 2 to fail, farther time horizons must be chosen.

VI. CONCLUSION AND FUTURE WORK

We proposed a method of statistical model checking in order to compute with accuracy a tiny probability associated with a timed temporal formula on a CTMC. We obtain a true confidence interval bounding this value. We have developed a theoretical framework justifying the validity of a confidence interval and ensuring the reduction of the variance. As the memory requirements (which depend on the time horizon) put a curb on the efficiency of the method, we propose three algorithms with a different trade-off between time and space. We have implemented these algorithms in the statistical model checker COSMOS and we have done experiments on several examples. We detailed one of them in the paper.

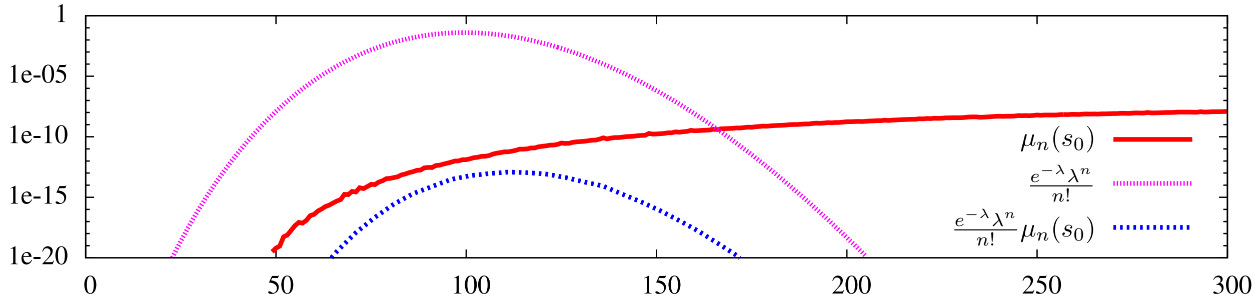


Figure 1. Repartition of Poisson and $\mu_n(s)$ probabilities

Table II
EXPERIMENTAL RESULTS FOR THE TANDEM QUEUES

k	Size of $\mathcal{C}$	numerical PRISM			Cosmos											
								Method 1			Method 2			Method 3		
		T (s)	Mem	$\mu_\tau(s_0)$	$\mu_\tau(s_0)$	$\mu_\infty(s_0)$	Conf. Int.	T_{pre}	T_{sim}	Mem	T_{pre}	T_{sim}	Mem	T_{pre}	T_{sim}	Mem
2	2601	0.021	156K	1.996e-13	1.993e-13	3.764e-8	1.732e-14	≈ 0	68	140M	≈ 0	69	140M	≈ 0	73	158M
3	132651	1.36	4.3M	1.694e-12	1.692e-12	9.196e-7	1.271e-13	≈ 0	144	202M	≈ 0	141	200M	≈ 0	137	200M
4	6765201	107	168M	9.381e-12	9.392e-12	1.524e-5	4.997e-13	1	243	259M	2	246	239M	1	250	237M
5	≈345e+6	5306	8400M	3.941e-11	3.941e-11	2.290e-4	1.725e-12	7	501	439M	7	538	310M	7	561	300M
6	≈17e+9	Out of Memory			1.355e-10	2.355e-3	4.031e-12	57	2577	1347M	57	2278	509M	54	2470	448M
7	≈897e+9				4.013e-10	8.391e-3	9.998e-12	415	33262	7039M	487	31942	1581M	387	33087	1213M
8	≈45e+12				1.051e-09	0.088	2.757e-11	Out of Memory			3030	261050	7502M	2896	267357	5157M

We plan to go further in several directions. Our first goal is to deal with infinite models whose reduction yields an infinite one and more expressive language logical formula. Finally we aim at defining formalisms on which the reduced model can be automatically produced.

REFERENCES

[1] C. Baier, B. R. Haverkort, H. Hermanns, and J.-P. Katoen, "Model checking continuous-time markov chains by transient analysis," in *CAV*, E. A. Emerson and A. P. Sistla, Eds., vol. 1855. Springer, 2000, pp. 358–372.

[2] M. Z. Kwiatkowska, G. Norman, and D. Parker, "Stochastic model checking," in *SFM*, M. Bernardo and J. Hillston, Eds., vol. 4486. Springer, 2007, pp. 220–270.

[3] A. Legay, B. Delahaye, and S. Bensalem, "Statistical model checking: An overview," in *RV*, H. Barringer, Y. Falcone, B. Finkbeiner, K. Havelund, I. Lee, G. J. Pace, G. Rosu, O. Sokolsky, and N. Tillmann, Eds., vol. 6418. Springer, 2010, pp. 122–135.

[4] P. Ballarini, H. Djafri, M. Duflot, S. Haddad, and N. Pekergin, "HASL: An expressive language for statistical verification of stochastic models," in *VALUETOOLS'11*, P. H. Samson Lasaulce, Dieter Fiems and L. Vandendorpe, Eds., Cachan, France, May 2011, pp. 306–315.

[5] G. Chiola, G. Franceschinis, R. Gaeta, and M. Ribaudo, "GreatSPN 1.7: Graphical editor and analyzer for timed and stochastic Petri nets," *Perform. Eval.*, vol. 24, no. 1-2, pp. 47–68, 1995.

[6] M. Z. Kwiatkowska, G. Norman, and D. Parker, "Prism: Probabilistic symbolic model checker," in *Computer Performance Evaluation / TOOLS*, T. Field, P. G. Harrison, J. T. Bradley, and U. Harder, Eds., vol. 2324. Springer, 2002, pp. 200–204.

[7] G. Behrmann, A. David, K. G. Larsen, P. Pettersson, and W. Yi, "Developing uppaal over 15 years," *Softw., Pract. Exper.*, vol. 41, no. 2, pp. 133–142, 2011.

[8] H. L. S. Younes, "Ymer: A statistical model checker," in *CAV*, K. Etessami and S. K. Rajamani, Eds., vol. 3576. Springer, 2005, pp. 429–433.

[9] G. Rubino and B. Tuffin, *Rare Event Simulation using Monte Carlo Methods*. Wiley, 2009.

[10] B. Barbot, S. Haddad, and C. Picaronny, "Coupling and importance sampling for statistical model checking," in *TACAS*, C. Flanagan and B. König, Eds., vol. 7214. Springer, 2012, pp. 331–346.

[11] A. Jensen, "Markoff chains as an aid in the study of markoff processes," *Scandinavian Actuarial Journal*, vol. 1953, no. sup1, pp. 87–91, 1953.

[12] T. Lindvall, *Lectures on the coupling method*. Dover, 2002.

[13] B. Barbot, S. Haddad, and C. Picaronny, "Importance sampling for model checking of continuous-time Markov chains," Laboratoire Spécification et Vérification, ENS Cachan, France, Research Report LSV-12-08, May 2012.

[14] B. L. Fox and P. W. Glynn, "Computing poisson probabilities," *Commun. ACM*, vol. 31, no. 4, pp. 440–445, 1988.

[15] M. Diaz, *Petri Nets: Fundamental models, verification and applications*. Wiley-ISTE, 2010.

[16] D. N. Jansen, "Understanding Fox and Glynn's "computing poisson probabilities"," Nijmegen: Radboud Universiteit, Tech. Rep. ICIS-R11001, 2011.

[17] L. Kleinrock, *Queueing Systems*. Wiley Interscience, 1976, vol. II: Computer Applications.

ISBN: 978-1-61208-234-9

Comparison of Lumped Simulation Models for Three Different Building Envelopes

Kyung-Soo Yoon
Home Appliance Air-Conditioning
LG Electronics
Seoul, South Korea
Kyungsoo.yoon@lge.com

Young-Jin Kim and Cheol-Soo Park
Dept. of Architectural Engineering
SungKyunKwan University
Suwon, South Korea
cheolspark@skku.ac.kr

Keon-Ho Lee
Dept. of Building Research
Korea Institute of Construction Technology
Goyang, South Korea
lee1ncdh@kict.re.kr

***Abstract*—Over the last several decades, shading devices such as blinds, louvers, roll shades, etc., have received much attention due to their roles as indoor environmental controllers (preventing glare, blocking shortwave radiation, and contributing to thermal comfort). In order to allow the aforementioned systems to act as true *optimal* controllers, a fast and accurate mathematical model that is able to predict the dynamic behavior of the system is necessary. This paper describes the development of a lumped simulation model of an indoor blind system adjacent to double glazing (6 mm clear+12 mm air + 6 mm low-e). This is typical of office building shading systems in Korea. Rather than attempting to develop the most detailed and accurate model of the system, the approach described in this paper is based on the postulated *minimalistic* model augmented with a parameter estimation technique. The lumped simulation model was validated with measurements obtained from an in-situ, full-scale experimental facility mounted on the south-facing façade. It was found that the calibration method delivers accurate results for the unknown parameters (convective heat transfer coefficients and air permeability of the shading device), allowing the calibrated lumped model to be used in ensuing optimal control and performance studies.**

Keywords-blind; lumped model; parameter estimation; calibration; validation.

I. Introduction

Saving energy through architectural design is an important issue due to climate change and high oil prices. In particular, interests in building envelope design, control, and performance assessment are increasing due to the impacts on building energy and comfort. The latest trend in envelope systems is to increase window area for transparency and aesthetics. However, this leads to undesired heat gain/loss, assymetric discomfort, and an increased energy consumption. Therefore, shading devices are often adopted to reduce the aforementioned problems. Shading devices installed indoors, outdoors, or in cavities have several effects on indoor environmental conditions (preventing glare, blocking shortwave radiation, and contributing to thermal comfort).

The development of simulation model, optimal control, and energy performance assessment studies for the following three envelope systems are now in progress by the authors. System I is a generic type used in a curtain wall system. Systems II and III are examples of the double-skin with different configuration (cavity depth, blind slat, etc.)

- System I: double glazing (low-e) + interior blind system (blind slat 50mm)
- System II: double-skin (50 mm cavity) system, blind slat 15mm
- System III: double-skin (200 mm cavity) system, blind slat 50mm

In order to assess the energy performances of the systems, it is necessary to develop a simulation model. Obviuosly, the developed simulation model can be used later in optimal design and control studies. In particular, the simulation model should be able to predict the behavior of the system quickly and accurately in order to apply optimal control.

There are three approaches to mathematically modeling a system: (1) the use of a 3D, full-blown model, (2) the use of a whole building simulation tool (Energy Plus, Esp-r, TRNSYS [1], IDA ICE [2], TAS [3], etc.), and (3) the use of a lumped simulation model.

The first approach divides the system into small nodes in the form of a grid, and then mathematically expresses the heat and mass transfers that appear in each node. While this approach has the advantage of precisely modeling the airflow dynamics and temperature distribution around the system, the mathematical modeling requires numerous assumptions as well as detailed information. For these reasons, the uncertainty of its simulation results might increase.

The second approach is to use a general-purpose tool developed for analyzing performance of a whole building. While this approach is advantageous in terms of assessing the influence of the envelope system on the performance of the entire building, it has a limited ability to express in detail the physical phenomena that involve transient convective and radiant heat transfer and airflow movement in and around the system. It is difficult to make accurate predictions about the airflow movement in a cavity [4], and it is not easy to apply modern control strategies (Pontryagin's minimum principle [5], the Hamilton-Jacobi-Bellman equation [5], a Riccati equation [5]), to any shading installed device (control of louver slat angles, ventilation dampers in the cavity, etc. because of high nonlinearity of the system.

The third approach is to express the fundamental heat transfer phenomenon in a system as one-dimensional (1D) in a lumped fashion. As a result, this model has the advantage of fast calculations. For real-time performance optimization of a system, this approach can be applied for optimal control and performance assessment. In other words, it is possible to optimize and assess system performance by determining the optimal variables in real-time [6].

The purpose of the paper is to present the initial development of the System I (Fig. 1) among the aforementioned three systems. In this study, the third approach was employed and the heat transfer and airflow movement are expressed in a 1D state-space equation. The lumped simulation model was calibrated using the parameter estimation technique. Next, it was experimentally validated with the test facility described in the following section.

This paper reports the following three processes for development of the lumped simulation model of System I.

- Step 1 (mathematical modeling): modeling complex heat transfer and airflow movement in the system
- Step 2 (calibration): estimating unknown parameters in the model
- Step 3 (validation): comparing simulated results with measurements.

II. Experiment Setup

An experimental test facility of the system was constructed in an actual building as shown in Fig. 2. The installed system faces true South and the window was double glazing (6 mm clear glazing + 12 mm air space + 6 mm low-e glazing), and the blind was placed adjacent to the indoor glazing. The blind was 10 cm from the surface of the interior pane, its height was 150 cm from the bottom to the top of the pane, and the width of the blind slat was 5 cm. The control of the slat angle was performed by an electric motor.

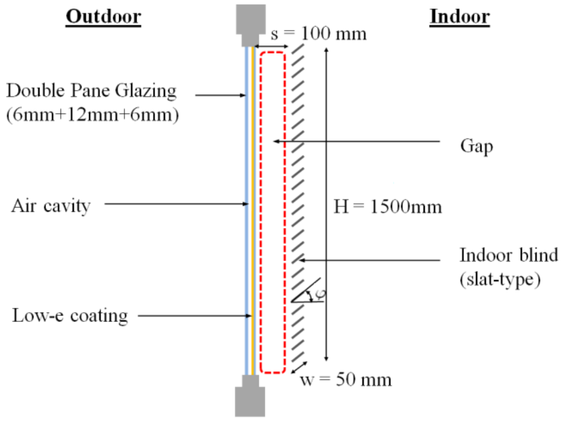
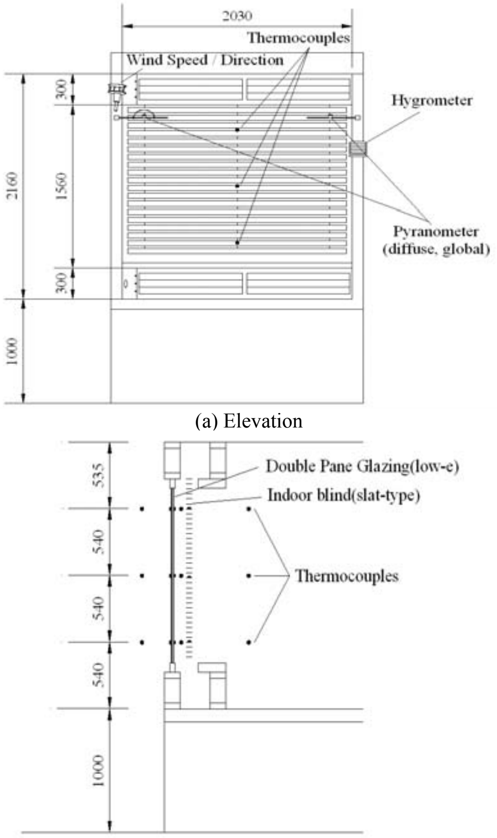


Figure 1. System I (24 mm double-glazing + interior blinds).

(a) Elevation (b) Interior

Figure 2. The test unit installed on Sungkyunkwan University campus, Korea.

Fig. 3 shows the elevation and section of the experiment unit, the locations of the sensors, and the measurement instruments used for study. Wind speed and wind direction were measured using a wind sensor (Wind Sonic, Gill inc.). Direct and diffuse solar radiation was measured using a pyranometer (S-LIB-M003, HOBO inc.). The outdoor humidity was measured using a hygrometer (M-RSA, HOBO inc.). T-type thermocouples were installed at three points vertically, as shown in Fig. 3, to measure glazing surface, gap air, and indoor temperatures. The data were collected using a National Instrument data logger.



(a) Elevation

(b) Section

Figure 3. Elevation and section of the experiment unit and the locations of the sensors (unit: mm).

III. Mathematical Model

In order to describe the dynamics of the indoor blind system, the governing heat and mass transfer phenomena were studied as follows: 1) direct, diffuse, and reflected solar radiation; 2) long wave radiation between surfaces; 3) convective heat transfer along the exterior and interior glazing surfaces and blind slats; and 4) air movement through the gap (between the glazing and the indoor blinds).

In this study, the heat transfer and airflow phenomena in the system were described in a lumped fashion. The essence of the lumped model is based on the assumption that the temperature of the solid is spatially uniform at any instant during the transient process [7]. With this in mind, a one-dimension model without a temperature gradient was assumed (Fig. 4) in order to describe the simplified dynamics of a three-dimensional (3D) system. $x1$-$x5$ are state variables which represent the temperature at each point in Fig. 4. Although this approach does not render explicit information about the vertical and horizontal temperature gradients, it is assumed to be sufficient to represent the overall thermal characteristics of any indoor blind system and, in particular, to determine the optimal control actions. This assumption has to be substantiated by experiments and will be described later in the paper.

For the details of the grey-box approach and the thermal model, see [8] [9].

This paper gives a detailed account of the airflow occurring in the gap shown in Fig. 4. The size of the space between the blind tip and the window (gap in Fig. 4) has an effect on the energy performance of the system [10]. Therefore, the gap is equated with a cavity in this study.

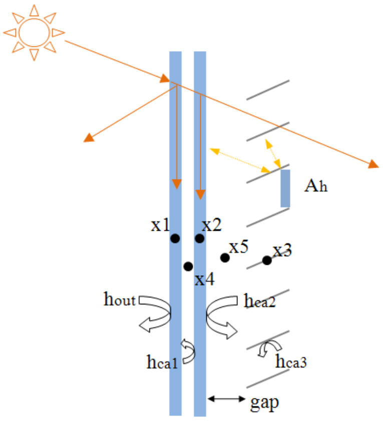


Figure 4. Simplified system (● = state variables, x1= outer glazing temperature of the double-pane, x2= inner glazing temperature of the double-pane, x3= louver slat temperature, x4= cavity air temperature in the double-pane, x5= air temperature in the gap).

Airflow in the gap is caused by a difference between the gap temperature (T_{gap}) and the indoor air temperature (T_{in}). The effects of the blinds (blind slat angle, distance from the glazing surface, etc.) also affect this airflow. The presence of the blind has a strong effect on the heat transfer from the indoor glazing. Moreover, when blind slats are fully closed (90°), the air velocity and convective transfer are promoted by the fully closed cavity effect [10].

In general, the airflow speed in the gap (Fig. 4) can be expressed in (1) [11] [12].

$$v_{gap} = \frac{\left[\left(\left(\frac{12\mu H}{s^2}\right)^2 + \frac{2\rho^2(1+z_{in}+z_{out})\rho_0 T_0 g H \sin\theta \left|T_{in}-T_{gap}\right|}{T_{in}T_{gap}}\right)^{\frac{1}{2}} - \frac{12\mu H}{s^2}\right]}{\rho(1+z_{in}+z_{out})} \tag{1}$$

where v_{gap} is the air velocity in the gap in m/s, μ is the viscosity of the gap air at temperature (T_{gap}) in NS/m^2, ρ is the air density in the gap in kg/m^3, Z_{in} is the inlet pressure drop factor, Z_{out} is the outlet pressure drop factor, T_0 is the reference temperature in K, ρ_0 is the density of air at temperature T_0 in kg/m^3, g is the acceleration due to gravity in m/s2, and θ is the tilt angle of the window in degrees (0° = horizontal, 90° = vertical).

Z_{in} and Z_{out} of (1) can be calculated using (2) and (3) [12].

$$z_{in} = \left[\frac{A_{gap}}{0.66\left(A_{bot}+\frac{A_{top}}{2(A_{bot}+A_{top})}(A_l+A_r+A_h)\right)}-1\right]^2 \tag{2}$$

$$z_{out} = \left[\frac{A_{gap}}{0.60\left(A_{top}+\frac{A_{bot}}{2(A_{bot}+A_{top})}(A_l+A_r+A_h)\right)}-1\right]^2 \tag{3}$$

where A_{gap} is the cross-sectional area of the gap in m^2, A_{top} is the area of the top opening in m^2, A_{bot} is the area of the bottom opening in m^2, A_l is the area of the left-side opening in m^2, A_r is the area of the right-side opening in m^2, and A_h is the air permeability of the shading device in m^2.

The air permeability of the shading device (A_h , Fig. 4) changes according to the configuration of the shading layer (roll shade, screen or blind, etc.). In the case of the blinds, A_h varies with the blind slat angle. Based on the descriptions given above, the mathematical model was expressed with a state-space equation, as shown in (4).

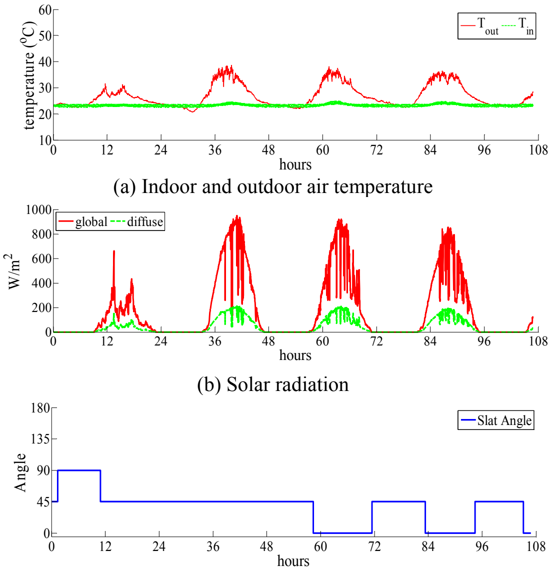
$$\dot{x} = A(u,t)x + b(u,t) \tag{4}$$

Copyright (c) IARIA, 2012.   ISBN: 978-1-61208-234-9

where x is the state variable vector, A is the state matrix, u is the input vector, and b is the load vector.

IV. Comparison between Simulated and Primary Measured Values

For a comparison of the simulated and measured values, the first experiment was conducted for about 108 hours (Jul 27, 2009 - Aug 30, 2009). The data were recorded with a sampling time of one minute, and the number of measured data points was 6,412 (6,412 minutes = four days, ten hours, and 52 minutes). During the experiment, the internal temperature of the laboratory was set to 24 °C, and the slat angle (0° = horizontal, 45° = facing the floor, 90° = vertical) was changed randomly, as shown in Fig. 5 (c). Fig. 5 (a) and (b) show the recorded indoor and outdoor air temperatures and the solar radiation.



(a) Indoor and outdoor air temperature

(b) Solar radiation

(c) Slat angle

Figure 5. Weather conditions and slat angle of the primary experiment.

Table 1 shows the differences between the measured temperatures and predictions from the un-calibrated simulation model that use literature values of the convective heat transfer coefficients (h) and air permeability of the shading device (A_h) [7] [11] [13]. It should be noted that the air temperature ($x4$) of the small cavity in the double-pane was not measured.

As shown in Table 1, the temperature difference between the simulation and measurement was 1.91 °C. Considering a degree of details of the lumped model, the un-calibrated model has an unexpected accuracy (the T-type thermocouples used in the experiments had an accuracy range of ±0.5 °C). The differences between the simulated and measured values result from the unknown parameters of the model, the assumptions used in the modeling process, and the simplifications of the physical phenomena in the system (3D→1D).

TABLE I. DIFFERENCES BETWEEN THE SIMULATED AND MEASURED VALUES (UN-CALIBRATED MODEL).

$\lvert x_{measured}-x_{simulated}\rvert$	**results (°C)**
x_1	2.81
x_2	2.60
x_3	0.86
x_5	1.38
Average	**1.91**

Table 2 shows the convective heat transfer coefficient values of the un-calibrated model. In the case of the convective heat transfer coefficients (Table 2), the literature values [7] [14] [15] [16] are derived empirically from experiments under specific conditions (vertical walls instead of windows). The convective heat transfer coefficients are influenced greatly by surface roughness and geometry, system geometry (height, width, etc.), the local environment and the nature of the air motion. For these reasons, they should be calibrated to fit to the test unit ($h_{ca,1}$, $h_{ca,2}$, $h_{ca,3}$ in Fig. 4).

TABLE II. CONVECTIVE HEAT TRANSFER COEFFICIENTS OF THE UN-CALIBRATED (W/M^2).

	Literature values	**Literature**
h_{out}	7.44	Reference [14]: smooth surface
	8.00	Reference [11]: summer conditions
	22.70	Reference [15] : summer conditions
$h_{ca,1}$	0.60	Reference [7]: the convective heat transfer coefficient when both ends of the wall were insulated, allowing only horizontal heat flow conditions
$h_{ca,2}$	2.42	Reference [12]: the convective heat transfer coefficient for the vertical cavity with a shade layer (shade, screen, blind)
$h_{ca,3}$	4.16	Reference [7]: the convective heat transfer coefficient for an isothermal horizontal cylinder

The air permeability of the shading device (A_h) depends on the blind slat angle, and it is difficult to measure accurate air permeability . Even in the fully closed position (vertical, 90°), there is air permeability through openings between blind slats. Such air permeability has an influence on airflow in the gap.

Thus, the unknown parameters related to the convective heat transfer coefficients and the airflow must be identified with a suitable parameter estimation technique based on extensive data points obtained from experiments. This will be discussed in the following sections.

V. Calibration

The parameter estimation technique is used to determine unknown parameters that minimize the differences between the actual measurements and the simulation predictions [8]. This approach is used to estimate values that cannot be calculated analytically or measured directly. The parameter estimation technique can be expressed as the minimization of the objective function (S), as in (5).

$$\begin{aligned} &\min\ \mathrm{S} = \sum_{k=1}^{z} \left[Y_k - \psi_k\left(\xi_i\right) \right]^T \left[Y_k - \psi_k\left(\xi_i\right) \right] \\ &s.t.:\ lb \le \xi \le ub \end{aligned} \tag{5}$$

where Y_k is the observation vector, ψ_k is the discrete state vector in discrete state space, z is the number of observations, ξ is a vector of the unknown parameters, lb is the lower bound of the unknown parameters, and ub is the upper bound of the unknown parameters.

In this study, the aforementioned convective heat transfer coefficients (Fig. 4) and the air permeability of the shading device, given by (2) and (3), were selected as the unknown parameters.

Equation (1) is orginally developed for the shade layer (e.g., roll shades) and needs calibration to be used for the indoor blind system installed in the experimental unit. In other words, air permeability of the shading device should be estimated because it cannot be calculated exactly, and it should be determined based on the blind slat angle. The selected unknown parameters are expressed in the following equations.

$$h_{out} = 5.678\left[\xi_1 + \xi_2 \left(\frac{u_{lsv}}{0.3048} \right)^{\xi_3} \right] \tag{6}$$

$$h_{ca,1} = \frac{N \cdot k_f}{D_s} + \xi_4 \tag{7}$$

$$h_{ca,2} = \xi_5 + \xi_6 \left(\Delta x\right)^{\xi_7} + \xi_8 \left(u_{ca}\right)^{\xi_9} \tag{8}$$

$$h_{ca,3} = \xi_{10} + \xi_{11} \left(\Delta x\right)^{\xi_{12}} + \xi_{13} \left(u_{ca}\right)^{\xi_{14}} \tag{9}$$

$$\begin{aligned} &A_h = a_1\xi_{15} + a_2\xi_{16} + a_3\xi_{17} \\ &If \quad \varphi_{slat} = 0^\circ,\ a_2 = a_3 = 0,\ a_1 = 1 \\ &If \quad \varphi_{slat} = 45^\circ,\ a_1 = a_3 = 0,\ a_2 = 1 \\ &If \quad \varphi_{slat} = 90^\circ,\ a_1 = a_2 = 0,\ a_3 = 1 \end{aligned} \tag{10}$$

where u_{lsv} is the local surface velocity in m/s, N is the Nusselt number [dimensionless], k_f is the thermal conductivity of the glazing in W/mK, D_s is the cavity width of the double-pane in m, ξ represents the unknown parameters, Δx is the temperature difference in K, and a is a constant.

Equation (6) is empirically driven for the convective heat transfer coefficient of exterior surfaces (h_{out}) [14]. The purpose of ξ_4 in (7) is to account for the end effect.

Equations (8) and (9) are mathematical representations of the convective heat transfer coefficients in the gap, and they are expressed as functions of airflow velocity and temperature difference (the pane surface and gap air temperatures). Simply put, Equations (8) and (9) can be read as a consideration of the convective effects according to airflow velocity and temperature difference.

The air permeability of the shading device (A_h) from (2) and (3) was expressed in (10) and also reflects the changes due to the blind slat angle (φ_{slat}).

The function LSQNONLIN in the MATLAB optimization toolbox was used to solve (5). LSQNONLIN is specially suited for this kind of constrained nonlinear optimization problem. The values of the unknown parameters were numerically estimated using LSQNONLIN. Table 3 shows the estimated convective heat transfer coefficients. There are considerable differences in convective heat transfer coefficients between the un-calibrated (Table 2) and calibrated models (Table 3). The calibrated value (h_{out}) in Table 3 is similar to the summer conditions in [15] (Table 2).

For $h_{ca,1}$, the estimated values were greater than those in the literature because the literature values are derived empirically from experiments under specific conditions (both ends of the wall are insulated, allowing for only horizontal heat flow). In other words, the literature values do not take into consideration the lateral heat loss that occurs in the cavity, and they are usually valid for solid walls, but not for transparent glazing.

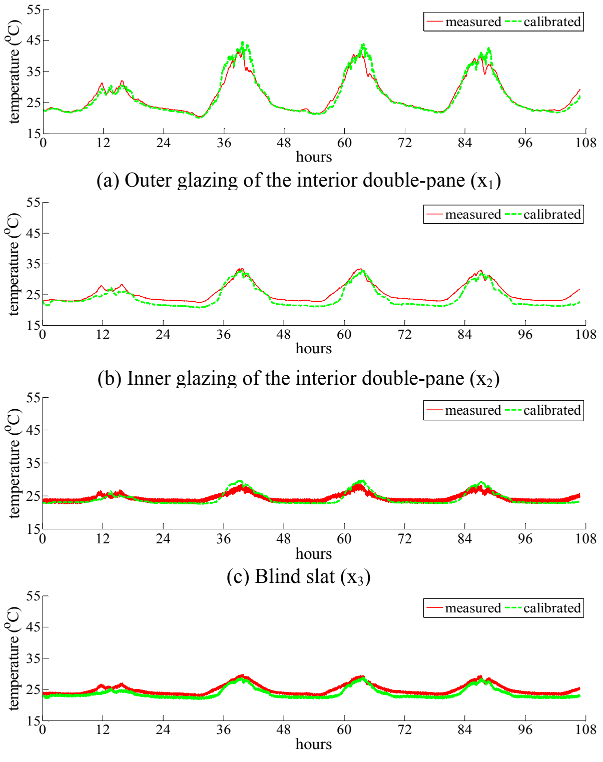
For $h_{ca,2}$, the estimated values were also greater than those calculated from the literature.

In counterpoint to the literature value for the shading layer, our study took into account variations in the airflow based on the blind slats. The differences in $h_{ca,2}$ were due to uncertain air movement in the gap (Tables 2, 3). $h_{ca,3}$ was close to that of the literature.

TABLE III. CONVECTIVE HEAT TRANSFER COEFFICIENTS OF THE CALIBRATED MODEL (W/M^2).

	Estimated values
h_{out}	24.52
$h_{ca,1}$	4.51
$h_{ca,2}$	8.23
$h_{ca,3}$	4.04

The estimated unknown parameters (ξ_1 - ξ_{15}) were applied to the simulation model and compared with the measured values (Fig. 6, Table 4). After calibration, the average difference between the simulated and measured temperatures was 1.13 °C (Table 4). It is clear from the results that the accuracy of the calibrated model was improved compared to that of the un-calibrated model. Furthermore, the estimated air permeability of the shading device depends on the blind slat angle (Table 5). The A_h for slat angles of $\varphi = 45°$ (facing the floor) was greater than the A_h for slat angles of $\varphi = 0°$ (horizontal). The A_h calculated using the calibrated results implies explicit conditions about the influence of the air permeability and the airflow configuration based on the slat angles and the impact of the convective heat transfer at the indoor glazing surface. That is, an ascending airflow is promoted by buoyancy (compared with the horizontal conditions) when φ =45°. This effect increases the convective heat transfer adjacent to the internal glazing surface.



(a) Outer glazing of the interior double-pane (x_1)

(b) Inner glazing of the interior double-pane (x_2)

(c) Blind slat (x_3)

(d) Gap between glazing and blind (x_5)

Figure 6. Comparison of the simulated and measured values for the calibrated model.

TABLE IV. DIFFERENCES BETWEEN THE SIMULATED AND MEASURED VALUES FOR THE CALIBRATED MODEL.

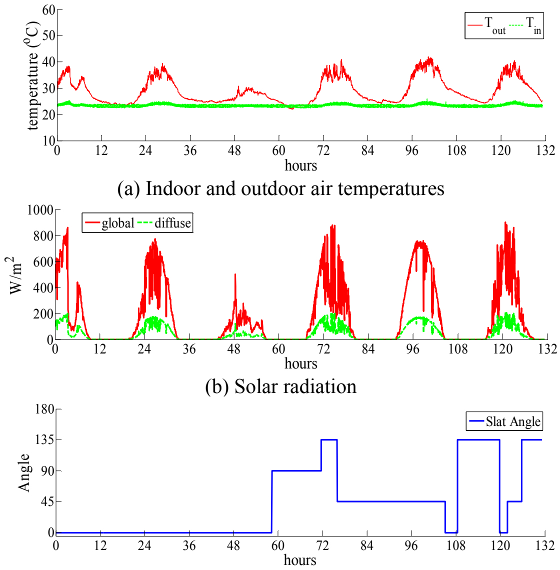
$\|x_{measured}-x_{simulated}\|$	**results (°C)**
x_1	0.88
x_2	1.50
x_3	0.90
x_5	1.23
Average	**1.13**

TABLE V. AIR PERMEABILITIES OF THE SHADING DEVICE.

	results (m2)
0°	2.29
45°	3.90
90°	0.82

VI. VALIDATION

Validation processes were performed to determine whether the calibrated model was capable of accurately predicting the system’s response. To validate the model, the second experiment was conducted for about 132 hours (Aug 1, 2009 - Aug 6, 2009). The data were recorded with a sampling time of one minute, and the number of measured data points was 7,874 (7,874 minutes = five days, 11 hours, 14 minutes). The measured values were compared with the predicted values from the calibrated model. Fig. 7 shows the recorded indoor and outdoor air temperatures and the solar radiation.



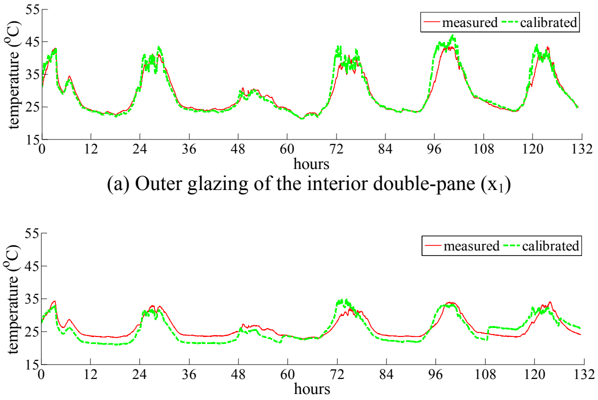
(a) Indoor and outdoor air temperatures

(b) Solar radiation

(c) Slat angle

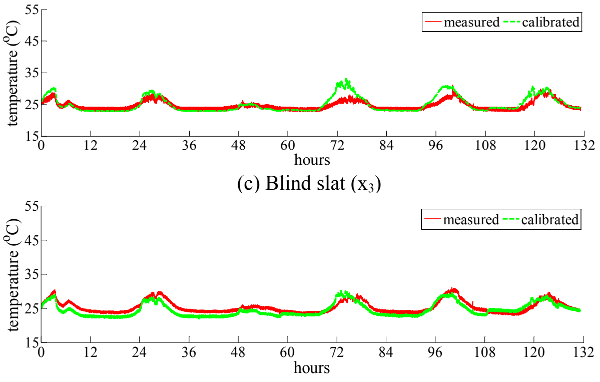
Figure 7. Weather conditions and slat angles of the second experiment.

Fig. 8 shows a comparison of the simulated and measured values, and Table 6 shows the average differences in temperature between the simulated and measured state variables. The overall average temperature differences were 1.27 °C. Considering the accuracy range (±0.5 °C) of the thermocouples (Omega T-type), the calibrated model proved surprisingly accurate in the prediction of the most relevant state variables.



(a) Outer glazing of the interior double-pane (x_1)

(b) Inner glazing of the interior double-pane (x_2)



(c) Blind slat (x_3)

(d) Gap between the glazing and the blind (x_5)

Figure 8. Validations of the state variables.

TABLE VI. RESULTS OF VALIDATION FOR THE CALIBRATED MODEL (STATE VARIABLES).

$\lvert x_{measured} - x_{simulated} \rvert$	**results (°C)**
x_1	1.02
x_2	1.79
x_3	1.02
x_5	1.24
Average	**1.27**

Table 7 shows the calculated convective heat transfer coefficients using the estimated unknown parameters. As mentioned above, there are many differences from the literature values (Table 2).

TABLE VII. RESULTS OF VALIDATION FOR THE CALIBRATED MODEL (CONVECTIVE HEAT TRANSFER COEFFICIENTS).

	Estimated values
h_{out}	24.25
$h_{ca,1}$	4.52
$h_{ca,2}$	9.60
$h_{ca,3}$	4.56

Fig. 9 shows the air velocity (v_{gap}) in the gap calculated using (1). The average air velocity was 15 cm/s, with a maximum value of 36 cm/s.

Considering the air velocity in the gap, the convective heat transfer phenomenon depends on airflow movements in the gap. In other words, air velocity is influenced by gap size and configuration. This indicates the need of model calibration based on the system configuration and the components in envelope system.

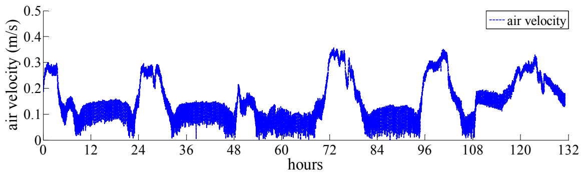


Figure 9. *Air velocity between the glazing and the blind (gap).*

VII. CONCLUSION AND FUTURE WORK

This study attempted to develop a simplified model for real-time optimal control and performance assessment of an indoor blind system. The unknown parameters in the mathematical model were estimated using a parameter estimation technique. The model was validated, implying that the mathematical model developed for this study is capable of accurately predicting system response. Based on the results of this study, the following conclusions can be made.

- Use of the lumped model: 3D modeling of heat transfer and airflow movement in a system is complicated, but the lumped model, expressed in one-dimension (1D), is a practical approach for predicting system behavior. Evidently, the 1D lumped model is also able to express the behavior of a system using a calibration technique.
- The unknown parameters: the un-calibrated model (using values from the literature) can be improved into a more accurate calibrated model using the parameter estimation technique. Namely, it was shown that there are limitations in developing a simulation model based solely on the parameters from the literature.
- Performance assessment and real-time optimal control: the simulation run-time was as short as several seconds using the calibrated model. The lumped model has the advantages of fast calculation, flexibility, etc., for emulating optimal control. The lumped simulation model can be applied to performance assessment and real-time optimal control.
- The airflow movement in the gap: it is necessary to consider airflow movement in the gap if a slat-type of indoor blinds are installed. Air permeability of the shading device (A_h) was estimated using the parameter estimation technique and was consequently adjusted according to the blind slat angle. Thus, the size and component of the gap are always of concern for modeling indoor blind systems since the airflow movement in the gap also changes with the blind slat angle.

Based on the results of this study, following studies are on-going.

- Optimal control and performance assessment: optimal control and performance assessment in cooling, heating, and intermediate modes, under differentweather conditions (clear, overcast), and orientations.
- Integrating of the lumped model with a whole building simulation model: the lumped model is applied to optimal control study, and the whole building simulation model is used to confirm the effect of the optimal control on the whole building energy performance.

- Applicability of the lumped model: the mathematical model of the double-skin system should be calibrated according to system configuration, local environment, components such as cavity width, cavity depth, cavity height, louver materials (reflectance, color, thickness, width, and geometric size), glazing type, etc. [9]. Subsequently, it is necessary to investigate the applicable range of the lumped model.

ACKNOWLEDGMENT

This research is supported by a grant (11 High-tech Urban G02) from High-Tech Urban Development Program funded by the Ministry of Land, Transport and Maritime Affairs of the Korean Government.

REFERENCES

[1] http://www.trnsys.com/

[2] http://www.equa.se/ice/intro.html

[3] http://www.edsl.net/main/

[4] O. Kalyanova, P. Heiselberg, C. Felsmann, H. Poirazis, P. Strachan, and A. Wijsman, “An empirical validation of building simulation software for modelling of double skin facade(DSF),” Proceedings of the 11th IBPSA Conference (International Building Performance Simulation Association), July 27-30, Glasgow, Scotland, 2009, pp. 1107-1114.

[5] F. L. Lewis, and V. L. Syrmos, “Optimal Control,” John Wiley & Sons, 2nd ed. 1995.

[6] C.S. Park, G. Augenbroe, N. Sadegh, T. Messadi, and T. Messadi, “Occupant Responsive Optimal Control of Smart Facade Systems,” 8th International IBPSA Conference, Eindhoven, Netherlands, 2003, pp. 1009-1016.

[7] F.P. Incropera and D.P. DeWitt, “Fundamentals of heat and mass transfer,” John Wiley & Sons, 6th ed., 2007

[8] C.S. Park, G. Augenbroe, T. Messadi, and M. Thitisawat, “Calibration of a lumped simulation model for double-skin facade systems,” Energy and Buildings. vol. 36. 2004, pp. 1117-1130.

[9] S.H. Yoon, and C.S. Park, “State-space Approach for Lumped Modeling of Double-skin System,” Proceedings of the Sustainable Building Conference 2007 Hong Kong (SB07 HK), December 3-5, Hong Kong, China, 2007

[10] D. Naylor, and H. Shahid, “Energy performance of a window with a horizontal venetian blind,” Energy and Buildings, vol. 37. 2005, pp. 836-843.

[11] ISO, “Thermal performance of windows, doors, and shading devices - Detailed calculations,” ISO 15099, 2003

[12] DOE, “Energy Plus Engineering Reference: The Reference to EnergyPlus Calculations,” US Department Of Energy, 2009

[13] S. Rheault, and E. Bilgen, “Heat Transfer Analysis in an Automated Venetian Blind Window System,” Journal of Solar Energy Engineering, vol. 111. no. 1. 1989, pp. 89-95.

[14] J.A. Clarke, “Energy Simulation in Building Design,” Butterworth-Heinemann, 2nd ed., 2001

[15] ASHRAE, “ASHRAE handbook fundamentals”. ASHRAE. Atlanta. USA, 2005

[16] W.H. McAdams, “Heat Transmission,” New York: McGraw-Hill, 1954.

Cost Optimization of a Nearly Net Zero Energy Building
A case Study

Narghes Doust, Marco Imperadori, Gabriele. Masera
Dept. of Building Environment Science & Technology (BEST)
Politecnico di Milano
Milan, Italy
narghes.doust@gmail.com,
marco.imperadori@polimi.it,
gabriele.masera@polimi.it

Francesco Frontini
Department for Environment Construction and Design (DACD)
Institute for Applied Sustainability to the Built Environment (ISAAC)
SUPSI
Canobbio, Switzerland
francesco.frontini@supsi.ch

Abstract— **The Net Zero Energy Building concept has received increasing attention in recent years, until becoming part of the EU policy on energy efficiency in buildings. Recently, a very important focus on cost-effectiveness has also been introduced. In particular, an EU regulation of 16 January 2012 establishes a comparative methodology framework for calculating cost-optimal levels of minimum energy performance requirements for buildings and building elements. The big challenge is to understand how much designers should rely on energy efficiency measures and when instead they should start to apply renewable energy technologies. The study presented in the paper is focused on the synergy between energy–efficiency in terms of envelope and renewable energy utilization to achieve a balanced energy budget over an annual cycle, minimizing at the same time the investment costs. An analysis adopting the cost optimality methodology on a residential case study has been carried out. Coupling TRNSYS 16, a transient system simulation tool, and GenOpt®, an optimization program, an optimization analysis has been performed in order to find a cost-optimal energy performance and to detect the best balance in terms of investment costs in envelope and in energy generation.**

Keywords- Zero energy building; Net zero energy buildings; Cost optimality analysis.

I. INTRODUCTION

The Nearly Zero Energy Building concept is a key issue for the next decade in Europe and not only. This is clearly pointed out in the Directive 2010/31/EU, which is the main EU-wide legislative instrument to improve energy performance in buildings [1]. Under this Directive, the Member States must apply minimum requirements as regards the energy performance of new and existing buildings and ensure the certification of their energy performance. In particular, the Nearly Zero Energy Building standard will become mandatory in 2019 for public buildings and in 2021 for private ones. Moreover, Lombardy Region (Italy) anticipated this deadline to the end of 2015 with a regional law issued on 18 April 2012 [2].

The EU directive requires nearly zero energy buildings, but since it does not give minimum or maximum harmonized requirements as well as details of energy performance calculation framework, it will be up to the Member States to define what the concept of Nearly Zero Energy Building stands for [3].

As highlighted by Marszal et al. [4,5] in the literature review of ZEB definitions, only few out of the reviewed definitions emphasize the importance of employing energy efficiency measures before using renewable energy sources. Therefore, Marszal et al. conclude that, in order to ensure that Net ZEBs are also very energy efficient buildings, a good solution could be to include a fixed value of maximum allowed energy use in the Net ZEB definition. However, when considering the Net ZEB concept, a new problem arises, i.e., to what level should we decrease the energy use by means of energy efficiency measures before the implementation of renewable energy sources [6]?

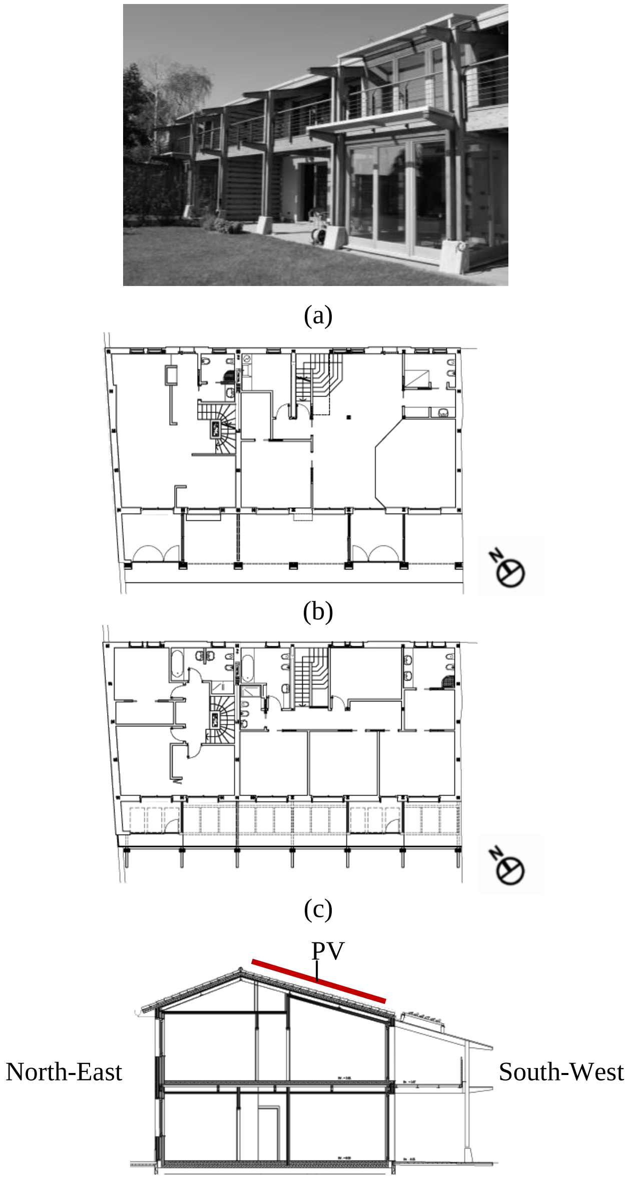
This paper treats the Net Zero Energy Building concept focusing in particular on the balance between envelope energy performance and energy production by Photovoltaic (PV) in terms of cost optimality. The aim is to investigate the existence of a compromise between a good envelope performance and investment cost while ensuring the Net Zero Energy Building target. This target is here treated according to the definition proposed by K. Voss and al. for which in Net ZEB total primary energy use, including building energy use, on a yearly basis is covered by energy produced on-site and building-connected renewable energy sources [7].

Currently, the cost optimality concept has been poorly investigated from this point of view and focused to obtain directions for residential building design. In the literature just a few national examples have been proposed by Aalborg University and Aalto University up to now; no data are available on residential building in the north weather [8,9, 10].

II. REFERENCE BUILDING

The present paper refers to a specific case study that is an existing residential building sited in Colognola, a small town near to Bergamo in the northern part of Italy, consisting of two independent homes sharing a party wall [11].

The northern façade, fully integrated with the historical context, is opposed to the South side towards the garden, where sunscreens, loggias and conservatories act as thermal collectors. The West side has no openings and is characterized by a ventilated skin of timber slats to avoid summer overheating of the envelope surface. Figure 1 shows picture, plans and cross section of the building.



(a)

(b)

(c)

(d)

Figure 1. The figure shows the picture (a), plan of the first (b) and second floors (c) and cross section of the building (d).

The envelope is based on a lightweight, stratified, dry-assembled construction system. This delivers a very high thermal performance, with very good behavior both in winter and summer. This building has been rated as "A - Gold" according to KlimaHaus protocol.

The heating system is based on a high-efficiency natural gas condensing boiler (efficiency at 30% partial load = 109%) combined to a radiant floor system working at low temperature. This system is characterized by flow temperatures of 28°C and 40°C, modulated by external probe and local temperature regulation in each room. In order to minimize the energy consumption and to ensure the necessary hygienic conditions inside the rooms, a mechanical ventilation system is provided to each of the two independent flats. Each unit is equipped with a cross-flow heat exchanger with 90% efficiency. Solar collectors provide more than 50% of the required domestic hot water.

The areas and U-values for both opaque and glazed parts of the building envelope are summarized in Table 1.

TABLE I. *GEOMETRICAL AND PHYSICAL PROPERTIES OF THE ENVELOPE "AS BUILT"*

	U-value	***Area***
	[W/m^{2}K]	**[m^2]**
External wall:		
South-West	0.12	70.8
South-East	0.12	74.0
North-West	0.12	77.5
North-East	0.12	128.0
Floor	0.16	220.0
Roof	0.09	245.7
Window:		
South-West	0.95	55.2
North-East	0.95	20.4

Sensible heat gains from equipments, including also heat gains due to artificial light, have been evaluated supposing the building occupancy. In particular, during the weekend a continuous building occupancy has been supposed. 100% occupancy corresponds to four people.

A typical working day occupancy is summarized in Figure 2.

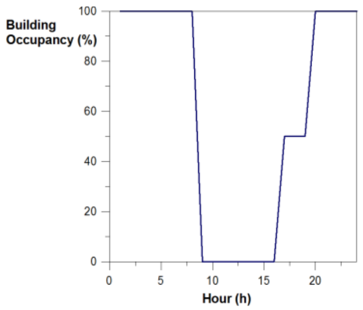


Figure 2. Occupancy schedule of the simulated building – week day.

III. NUMERICAL MODELLING AND BUILDING ENERGY PERFORMANCE

The TRNSYS 16 software [12] was used in order to perform a transient simulation of the energy behavior of the building. Trnsys model validation has been widely discussed in the literature [13].

The weather data of Bergamo was adopted for the simulation. The city lies 249 m above sea level, with a latitude of 45.70°N and a longitude of 9.67°E. The annual total solar radiation in Bergamo is 1,398 kWh/m^2 with approximately 1,900 hours of sunshine.

The heating system is operating between November and March; the cooling system is set to activate summer months only when indoor temperature is higher than 26°C, while the threshold for humidity control is 60% independently from the indoor temperature. In the remaining months, the building is in free-running condition.

The results related to the "as built" configuration show that the heating and cooling demand is equal to 13 kWh/m^{2}y.

In order to define the best envelope performance in terms of cost for the specific case study, TRNSYS was coupled with GenOpt, an optimization program for the minimization of a cost function.

GenOpt is designed for finding the values of user-selected design parameters that minimize a so-called objective function. The objective function is calculated by an external simulation program, in this case TRNSYS.

The Hybrid Generalized Pattern Search Algorithm with Particle Swarm Optimization Algorithm implemented in GenOpt was used [14]. Such an algorithm is a hybrid global optimization algorithm that starts by doing a Particle Swarm Optimization (PSO) on a mesh for a user specified number of generations $n_G \in N$. Afterwards, it initializes the Hooke-Jeeves Generalized Pattern Search (GPS) algorithm using the continuous independent variables of the particle with the lowest cost function value. The optimization problem has continuous and discrete independent variables, then the discrete independent variables are fixed at the value with the lowest cost function value by the GPS algorithm [14].

This approach is summarized in Figure 3.

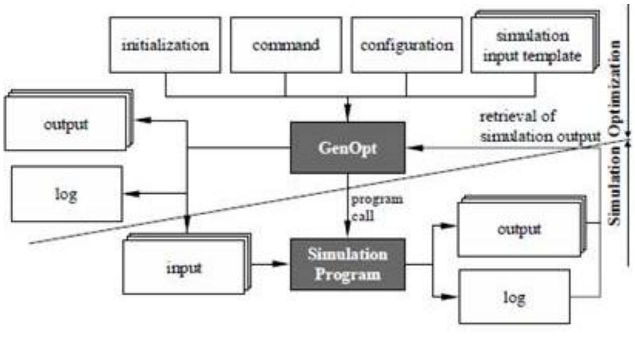


Figure 3. Interface between GenOpt and the simulation program (TRNSYS 16) [13]

GenOpt automatically rewrites the input files for TRNSYS at each iteration changing the variables taken into account. After this, it runs the simulation program, reads the output value of the function to be minimized from the simulation result file and then determines the new set of input parameters for the next run. The whole process is repeated iteratively until a pre-defined criterion of convergence is fulfilled or a maximum number of iterations is reached [14].

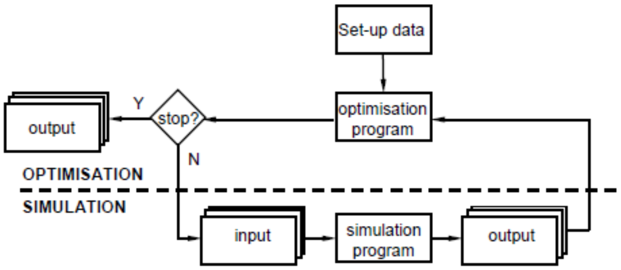


Figure 4. Combined simulation-optimisation [15]

The parameter taken into account to achieve low energy loads is the U-value of the envelope. Other parameters, like window to wall ratio or building orientation, have not been considered not only to ensure the necessary design freedom, but also because they scantily affect the energy response of the building. This consideration is the result of sensitivity analysis done before to assess which are the parameters that mainly govern the building energy behavior [11].

In the sensitivity analyses done, have been considered the U-value of the roof and walls, the building orientation, the glazing fraction of the wall and the wall concrete thickness. Starting from the "as built" situation each of the mentioned parameters were changed keeping all the others at the initial values.

Looking at the results represented in Figure 5, it is possible to observe how a change from 30% to 80% of the glazing fraction causes a change of less than 2 kWh/m^{2}y in the annual combined consumption (H+C), a variation from 3 to 8 cm in concrete thickness and a 360° rotation of the building both bring to a variation of about 1 kWh/m^{2}y and finally a change from 0.1 W/m^{2}K to 0.4 W/m^{2}K in the U value of the roof and wall brings an annual combined load variation larger than 7 kWh/m^{2}y.

In this way, it is possible to conclude that the U-value of the envelope is at the same time the easiest parameter to control independently of architectural design choices and the most decisive one. Considering this value as a variable, an optimization process has been run by considering the global cost of the construction as the function to minimize. The global costs here considered take into account envelope investment costs and the energy costs depending on the primary energy demand. The energy costs were calculated according to the EN 15459 standard in order to consider market movements with respect to energy price increases for the analyzed period (20 years) [16]. EN 15459 regards the economic evaluation procedure for energy systems in buildings and is also used by the Commission delegated regulation (EU) N° 244/2012 of 16 January 2012 which establishes a comparative methodology framework for calculating cost-optimal levels of minimum energy performance requirements for buildings and building elements [17].

Keeping a fixed thermal transmittance value for windows, equal to 0.95 W/m^{2}K, the thickness of the façade insulation was changed in order to vary its U-value from 0.09 W/m^{2}K to 0.35 W/m^{2}K.

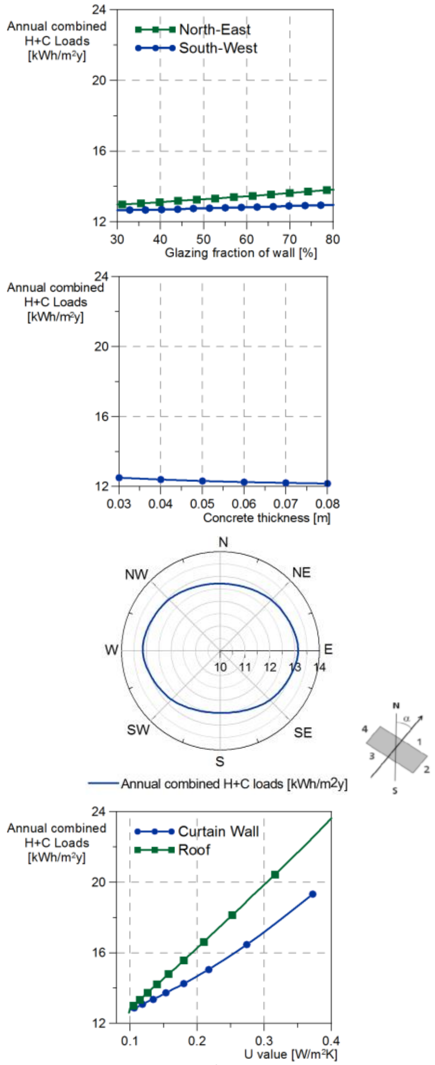


Figure 5. Results of sensitivity analyses.

The optimization performed considering the global cost as the cost function allowed us to define the curve that represents the relationship between global costs and primary energy demand (Figure 6).

The graph shows that the minimum point of the Cost/Energy demand curve does not coincide with the minimum point in terms of primary energy demand. This means that to invest overly on the envelope performance is not the best choice if considering only the cost optimality point of view [18].

It is worth noting that only the energy used for the building operation has been taken into account; the energy embedded in the building construction was not considered.

Since the European regulation requires nearly zero energy building starting 2020 but does not provide any limitation in terms of energy performance related to cost analyses, further investigations have been done to find the best investment balance between envelope and energy production by renewable sources.

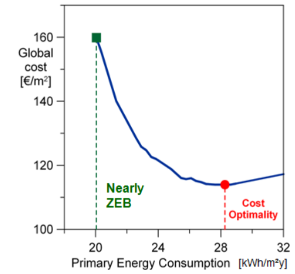


Figure 6. *Cost curve in function of envelope performance accounting a study period of 20 years.*

IV. TOWARDS NET ZERO ENERGY BUILDINGS

The next step of the study was to assess the effect of PV technology to achieve nearly ZEB status. To predict the behavior of PV panels was added an additional component (type 194) to TRNSYS building model.

TRNSYS PV model used is a five-parameter model based on an equivalent circuit of a one diode-model (Figure 7). This approach is useful to predict the energy production of monocrystalline PV power plants and requires very few parameters [19,20,21,22].

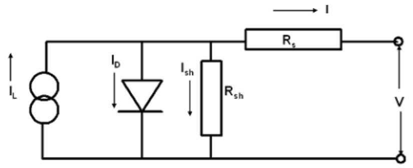


Figure 7. Equivalent circuit for the 5-Parameter model [21].

A new optimization process has been run considering both the envelope (U value) and the PV surface on the roof as variables. The global costs during the 20 year period have been again adopted as the function to minimize.

The PV panel here considered is a high performance panel characterized by cells efficiency higher then 20% and a nominal power of 330 W [23].

In the global cost evaluation, also the costs of the PV panels have been considered as well as the PV energy production that is subtracted from the primary energy demand if production and demand happen at the same time, while when production is larger than demand the remaining energy is sold to the grid.

Figure 8 shows that also in this case the minimum point in terms of cost does not coincide with the minimum point of the primary energy demand.

In this case, the optimization takes into account the actual situation of Italian government incentives (IV Conto energia [24]), which imply a considerable reduction of the global cost. Also in Figure 8, we present a comparison with the cost-energy demand curve previously discussed (Figure 6) .

A set of optimizations was run to study the different results obtained by considering different values of government incentives for each kWh of produced sold to the grid. This price was considered variable from 0 to 0.25 €/kWh.

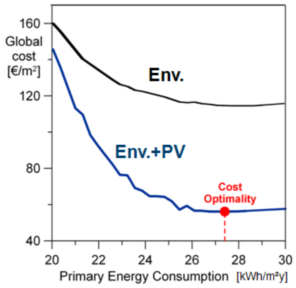


Figure 8. Cost curve in function of envelope performance and energy production by BIPV accounting a study period of 20 years.

Figure 9 summarizes the results focusing on the envelope, PV and global costs of the cost optimal solutions while Table 2 highlights, for the same solutions, the annual heating (EP_H) and energy (H+C) demands together with the PV energy production.

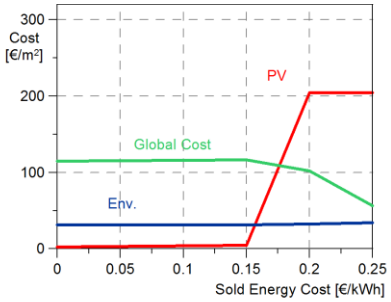


Figure 9. Results obtained by considering different values of government incentives.

TABLE II. ANNUAL HEATING (EP_H) AND ENERGY(H+C) DEMANDS WITH THE ENERGY PV PRODUCTION

Sold Energy cost	***EP_H of optimal solution***	***Heating+ Cooling demand of optimal solution***	***PV energy product of optimal solution***
[€]	**[kWh/m^{2}y]**	**[kWh/m^{2}y]**	**[kWh/m^{2}y]**
0	16.21	28.26	0.64
0.15	16.21	28.26	0.64
0.20	15.85	27.84	57.35
0.25	15.42	27.40	57.35
0.30	15.02	26.99	57.35

The graph shows that the envelope solution optimizing the global cost is not dependent from the government incentives: as a matter of fact, the line representing the envelope investments is quite horizontal.

This envelope solution corresponds to a building that is able to guarantee an annual energy demand (H+C) of about 27 kWh/m^{2}y and an energy heating demand (EP_H) of about 15 kWh/m^{2}y.

Increasing the price of sold energy, the optimized envelope is still the same, but the global cost optimal solution tends to have increased PV investment thus decreasing the global costs.

Considering the case in which the customer can not sell his overproduction of energy providing it for free to the grid (sold energy cost equal to 0), it is possible to observe that the cost optimal solution is not a net zero energy solution.

In this case, in order to ensure the net zero energy performance to the building it is necessary to provide energy by means of further PV panels thus moving away from the cost optimal solution.

Looking at the primary energy demand for space heating, Figure 10 shows a comparison of the situations for different types of envelope solutions. The 45° line represents a NZEB situation since the EP_H is equal to the energy production. Moving from the horizontal axis that represents the situation in which no PV panels are available, the figures represents which is the PV panel investment aimed to reach the NZEB solution for different building which envelope is rated in different classes according to Lombardy standard. For each of those solutions also the PV and envelope global costs are reported.

It is worth noting that a building rated in class A according to the energy performance scheme of Lombardy is the choice which is able to guarantee the lowest global cost.

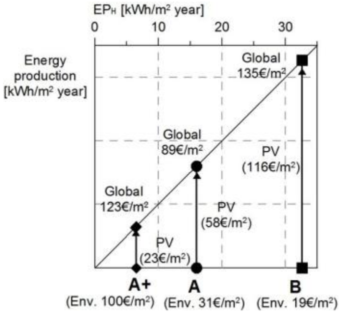


Figure 10. Cost comparison for different type of envelope solutions.

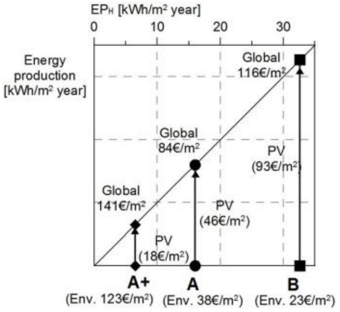


Figure 11. Cost comparison for different type of envelope solutions projecting the data in 20 years.

The same results can be achieved by projecting the data in 20 years (Figure 11). A price reduction of 20% of the PV modules price has been suggested and an increase of the envelope investment costs was calculated according to EN 15459 [16], that takes into account the trends.

This case also shows that a class A envelope is a reasonable trade off in terms of global costs.

CONCLUSION AND FUTURE WORK

The numerical investigation here presented allows to draw some conclusions about the design strategies that may be adopted to achieve cost optimality and NZEB performances for the investigated case study, which represents a large part of the building stock of Northern Italy:

1. The U value of the roof and walls is the design parameter that mostly affects the energy response of the building in terms of annual energy demand.
2. In case in which any PV renewable energy is adopted, the cost optimal solution is the one with a primary energy demand of 28 kWh/m^{2}y (H+C).
3. In the case of renewable energy supply by means of PV panels, the envelope solution that guarantees, in any case, the cost optimality is characterized by an EP$_H$ equal to 16 kWh/m^{2}y, typical of a class A according to the energy rating of Lombardy Region.
4. Class A according to the energy rating of Lombardy Region is the most promising solution to reach net zero energy performance. As a matter of fact both more (A$^+$) and less (B) performing envelopes require higher global costs to reach the NZEB standard.

Further analysis are in progress in order to extend the results to different building aspect ratios (Surface/Volume) and to investigate the role of the ratio between envelope surface and available surface for PV panels in defining the cost optimal solution.

ACKNOWLEDGMENT

The authors want to acknowledge Atelier2, designers of the building, and Vanoncini Spa that financed and built the investigated case.

REFERENCES

[1] The Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings. Official Journal of the European Union, 53, 2010.

[2] Regional law number 7 of 18th April 2012. “Misure per la crescita, lo sviluppo e l’occupazione”.Official Journal of Lombardy Region (Italy).

[3] J. Kurnitski, F. Allard, D. Braham, G. Goeders, P. Heiselberg, L. Jagemar, R. Kosonen, J. Lebrun, L. Mazzarella, J. Railio, O. Seppänen, M. Schmidt and M. Virta, “How to define nearly net zero energy buildings nZEB, REHVA proposal for uniformed national implementation of EPBD recast”, REHVA Journal, May 2011, pp. 6 – 12.

[4] A.J. Marszal, P. Heiselberg, J.S. Bourrelle, E. Musall, K. Voss, I. Sartori and A. Napolitano, “Zero energy building - A review of definitions and calculation methodologies”, Energy and Buildings 43, 2011, pp. 971 – 979.

[5] IEA SHC Task 40 / ECBCS Annex 52 ‘Towards Net Zero Energy Solar Buildings (NZEBs), http://www.iea-shc.org/task40/index.html (retrieved: September, 2012)

[6] A.J. Marszal and P. Heiselberg, “Life Cycle Cost analysis of a multi-storey residential Net Zero Energy Building in Denmark”, Energy 36, 2011,pp. 5600 – 5609.

[7] K. Voss, E. Musall and M. Lichtmeß, “From low-energy to net zero-energy buildings: status and perspectives”, Journal of Green Building, Volume 6, Number 1, 2011, pp. 46 – 57.

[8] A.J. Marszal, P. Heiselberg, R. Lund Jensen, J. Nørgaard, “On-site or off-site renewable energy supply options? Life cycle cost analysis of a Net Zero Energy Building in Denmark”, Renewable Energy, 44, 2012, pp.154– 165.

[9] J.Kurnitski, A. Saarib, T. Kalameesc, M. Vuolled, J. Niemeläd, T. Tarke, “Cost optimal and nearly zero (nZEB) energy performance calculations for residential buildings with REHVA definition for nZEB national implementation”, Energy and Buildings 43, 2011, pp. 3279–3288.

[10] A. Hasan, M. Vuolle and K. Sirén, “Minimisation of life cycle cost of a detached house using combined simulation and optimisation”, Building and Environment 43, 2008, pp. 2022 – 2034.

[11] N. Doust, G. Masera and M. Imperadori, “A study on Design Optimization of a Near Net-Zero Energy House”, Proceedings of the XVIII IAHS World Congress on Housing Science, 2012, Turkey.

[12] TRNSYS 16. A Transient Simulation Program. University of Wisconsin, Madison, WI.

[13] C. Adam and P. Andre. "Validation exercise applied to some TRNSYS components in the context of IEA 34/43." Paper presented at EPIC 2006 AIVC. Lyon, France, novembre 2006.

[14] M. Wetter, “GenOpt, generic optimisation program, user manual”, version 2.0.0. Technical report LBNL-54199, Lawrence Berkeley National Laboratory, 2004.Manual and program downloadable free of charge from http://gundog.lbl.gov/GO (retrieved: September, 2012)

[15] A. Hasan, “Optimal design of Net Zero Energy Buildings”, Low-Energy Architecture (LEA), World Renewable Energy Congress 2011, Sweden.

[16] EN 15459, Energy performance of buildings – economic evaluation procedure for energy systems in buildings, November 2007.

[17] Commission Delegated Regulation No 244/2012 of 16 January 2012 supplementing Directive 2010/31/EU) on the energy performance of buildings.

[18] The Buildings Performance Institute Europe (BPIE), “Cost optimality: Discussing methodology and challenges within the recast Energy Performance of Buildings Directive”. http://dl.dropbox.com/u/4399528/BPIE/BPIE_costoptimality_publication2010.pdf (retrieved: September, 2012).

[19] A.N. Celik and N. Acikgoz, “Modelling and experimental verification of the operating current of mono-crystalline photovoltaic modules using four- and five-parameter models”, Appl Energy 84, 2007, pp.1–15.

[20] J.A. Duffie and W.A. Beckman, “Solar Engineering of Thermal Processes”, 2nd Edition, John Wiley & Sons, Inc., New York. 1991.

[21] J. Nelson,The Physics of Solar Cells, Imperial College Press, 2003, London.

[22] W. De Soto, S.A. Klein and W.A. Beckman, “Improvement and validation of a model for photovoltaic array performance”, Solar Energy, 80, 2006, pp.78–88.

[23] http://www.sunpowercorp.co.uk/commercial/products-services/solar-panels/e20/ (retrieved: September, 2012)

[24] http://www.conto-energia-online.it/(retrieved: September,2012).

An Automatic Approach for Parameter Optimization of Material Flow Simulation Models based on Particle Swarm Optimization

Christoph Laroque
Institute of Applied Computer Science
Technical University of Dresden
Dresden, Germany
christoph.laroque@tu-dresden.de

Jan-Patrick Pater
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany
ppater@hni.upb.de

***Abstract* - The analysis of production systems by the use of discrete, event-based simulation is widely used and accepted as decision support method. It aims either at the comparison of competitive designs or the identification of a "best possible" configuration of the simulation model. Here, combinatorial techniques of simulation and optimization methods support the user in finding optimal solutions, which typically result in long computation times and though often prohibit a practical application in today's industry. This paper presents a fast converging procedure as a combination of a swarm heuristic, namely the particle swarm optimization, and the material flow simulation to close this gap. Faster convergence is realized by a specific extension of classic PSO implementations. First results show the applicability with a simulation reference model.**

*Keywords-**Design of Experiments; DES; meta-heuristics; particle swarm optimization; parameter optimization.***

I. MOTIVATION

Modern business computing, especially in the area of operations research, offers a wide variety of methods for complex problem solving for planning, scheduling and control of production and logistic processes. Those processes, which are to be designed or improved, are typically projected to mathematical models and then optimized by the use of simulation and/or optimization technologies. In both disciplines, models as an abstraction of the real-world system are used to improve decision variables and resulting key performance indicators under a given set of restrictions, e.g., the identification of the maximum throughput of a production site or network. In simulation, this improvement is usually achieved by the iterative evaluation of multiple scenarios and their subsequent simulation results. In the case of optimization, the optimal configuration is achieved by mathematical optimization algorithms or (meta-) heuristic approaches [12].

Due to the high computational demand of these methods, specific procedures as a combination of both simulation and optimization were derived, to combine both advantages: an optimization algorithm can be used to automatically generate a specific model configuration, which can be evaluated by simulation runs [9]. Especially for simulation models with stochastic influence factors, which need a high amount of simulation runs, these procedures can lead to faster identification of improving model configurations than standard methods for the design of simulation experiments [9].

This paper presents a feasibility study for a specific combination of simulation and optimization, where material flow simulation for production processes is combined with the meta-heuristic approach of particle swarm optimization. The goal is the development of a fast converging procedure model, which can be applied in a practical, industrial environment. Especially in this area, the given complexity of the underlying production system, and thereby the simulation model, is very high, so that the application of standard combinatorial approaches of mathematical optimization and material flow simulation is prohibited, since it needs an excessive amount of computational power.

The paper presents in the following sections in short the necessary state-of-the-art in discrete, event-based simulation, methods for the design of experiments as well as particle swarm optimization. The conceptual approach of the procedure is presented in Section 3, followed by the prototypical implementation in the material flow simulation tool d³FACT. The first evaluation results of the procedure are shown in Section 5. The paper closes with an outlook on future work in this area.

II. STATE-OF-THE-ART

A. Discrete, event-based material flow simulation

Simulation, especially material flow simulation is a methodology in operations research, which uses a model to describe a real world system, which in turn is a collection of entities, interacting together [9]. It has some distinct advantages, since it enables the observation of a system's behavior even before it exists in reality and furthermore for very complex, dynamic models. Based on the generated insights, users are able to validate the design of the system, to develop strategies for its operation or to determine optimal operating configurations. It is particularly suitable to determine how the system can be configured in the best way, especially when it is difficult or very costly to change an actual system design or to test control rules for a specific material flow in real. Time periods can be reviewed much quicker, sensitivity analysis and a graphical representation of the simulation model and its dynamics are further

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-234-9

advantages. Consequently, this will apply, where analytical methods fail due to system's complexity and if there is no other way to analyze an existing or designed system [6],[13].

B. *Design of Experiments*

Design of experiments (also DOE [11] or experimental design) refers to the use of statistical techniques to create an efficient, systematic set of controlled experiments for collecting data efficiently in order to estimate relationships between independent and dependent variables through measurement. DOE is used in engineering to design physical experiments to determine physical relationships (e.g., effect of pressure and temperature on yield in a manufacturing process). In the area of simulation, DOE is used for the systematic evaluation of simulation models in order to identify a set of model parameters, which leads to the desired simulation results. Each simulation run hereby evaluates a concrete set of parameters. Typically, the simulation models include stochastic influence factors, so that a single simulation run is not sufficient for the evaluation of the parameter set and multiple simulation runs for each of the configuration sets are to be performed. Efficient procedures like 2^k-factorial-Design, fractional designs, Plackett-Burman-experiments as well as response-surface method (RSM) or evolutionary optimization (EVOP) are used [11].

As mentioned above, the combination of simulation and optimization methods is also used and known, but today leads typically to high computational demands, which the practical application. Key factor for a successful application of such combinatorial approach is the fast convergence of the designed procedure.

C. *Particle swarm optimization*

Optimization algorithms can be distinguished in two areas, namely exact and heuristic methods [12]. Exact methods like linear or dynamic programming are based on a mathematical model and deliver an optimal solution. A great drawback is, however, that these procedures cannot solve some problems in an acceptable time period. Furthermore, they may not deliver solutions at all. Heuristics or meta-heuristic approaches are able to find a feasible solution in a shorter period of time, but cannot guarantee optimality. The GAP, the difference between the solution and an optimal solution, may not be reached [12]. Heuristics are problem specific procedures and can be divided into heuristics for generating intial solutions and heuristics, improving a given solution. Meta-heuristics are more generalized procedures, which can be applied to a broader range of problem instances. Typical examples in this domain are Taboo-Search, Simulated Annealing, Iterative Locale Search as well as Evolutionary Algorithms, Ant Colony Optimization, Swarm algorithms and Neuronal Networks [12].

The particle swarm optimization approach (PSO) was initially formulated by Kennedy and Eberhart in 1995 [8]. It is an evolutionary algorithm with a fixed population out of the group of meta-heuristics. It utilizes a population of individuals which create a set of solutions. A member of this population is called particle. Each particle has a specific position within the solution space, spanned by the sum of restrictions. Its basic idea is based on the Boid model of Reynolds, Heppner and Grenander [7]. Here, a boid is an individual, that moves in the same direction as its neighbor (alignment), but also tries to move to the middle of the group (cohesion). At the same time the individual tries to maintain a minimum distance to the others (separation). This behavior is inspired by natural phenomena, well known e.g. by bird or fish flocks. The information exchange between the individuals implements a social behavior. Kennedy and Eberhart developed collision-free particles, that search an optimal fitness value, the so called 'cornfield vector' [8]. Each particle stores the best positions found by the entire group as well as its personal best position

III. IDEA

A. *Fundamental approach*

The combined procedure of simulation and meta-heuristic presented in this section differs in an essential way from existing PSO-approaches, since the evaluation of the fitness of a particle is not possible in a direct way, but has to be derived through at least one, in most cases multiple simulation runs. Nevertheless, the core issue remains, that the simulation runs claims time and computing capacity and therefore a major objective to study is the determination of appropriate parameters for the algorithm, so that the number of required fitness evaluations is minimized.

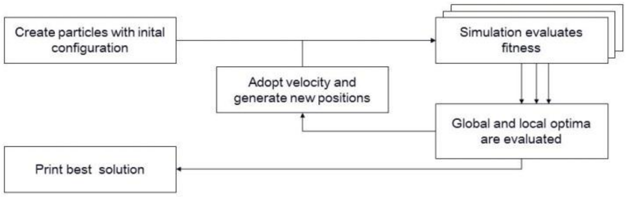

Create particles with inital configuration
Simulation evaluates fitness
Adopt velocity and generate new positions
Global and local optima are evaluated
Print best solution

Figure 1. Principle procedure for parameter optimization.

Figure 1 shows the designed process of an automatic experimental design procedure as a combination of simulation and the PSO. Based on an initialization of the simulation model the input factors of the simulation model and their modeled limitations are derived. The PSO-algorithm then creates particles for a given number of configurations, which are to be evaluated in a first iteration. The simulation runs are fed by this configuration and evaluate the configuration by simulating and generating key performance indicators that allow the determination of the fitness value of each particle. The best solution found is stored and distributed to each particle and its position in the solution space, meaning the configuration of the simulation parameters, is adjusted. Then, the next iteration for the simulation-based evaluation of the adjusted particles starts. The procedure repeats until a given termination criteria is reached (e.g., number of iterations, no improvement of the KPIs).

An important point of interest by using this approach is the application to complex systems with stochastic influences. This is due to their stochastic nature which always has a different result - and therefore a different fitness. A problem which arises here is the emergence of noise. One way of addressing this problem is evaluating a location several times to form an average (multisampling), or to exploit the history for sequential scanning. Here, the last values are smoothed to calculate a trend. The noise also causes particles not to reach the global optimum, as this cannot be precisely identified [2].

Additional improvements have been studied and introduced [3]. As the number of iterations of the algorithm is crucial for the computational effort, in conjunction with simulation, some modifications were made, that proved to be suitable in preliminary investigations. These were in detail:

- Intelligent positioning - The particles are specifically placed in certain areas of the solution space.
- Subdivision in three phases - exploration, exploitation and intensification. (further referred to as 3-phase PSO, cp. Section 3.2)
- Adaptive velocity - In the first phase (3 phase-PSO) the velocity of the particles is reduced to contain the explosion of the particles in the solution space.
- Selection - The particles of supposedly poor regions of the search space are relocated in good areas.
- Multisampling - The positions of the particles are evaluated several times and averages are calculated to reduce stochastic effects.
- Threshold - An additional stop criterion

B. Construction of the 3-Phase-PSO algorithm

The extensions to the PSO in the previous chapter provided some starting points. A compromise between exploring the search space and the convergence is to be found.

After an intelligent initialization the solution space can be subsequently investigated with a limited number of iterations and the adapted behavior of the particles. The next step is to select particles by their fitness values which are better than the rest. A further potential is obtained by exploiting a history for the previous positions as a particle could be pulled out by the global best position in the course of iterations from a good range. When only the current information is used, good areas might be neglected under certain circumstances. History allows that at the end of the first phase of the algorithm, the particles are repositioned in a well-defined number of top positions throughout the history. This selection has another positive side effect. The starting position of the particles has already been evaluated and is known. This again saves computational effort for the first iteration of the second phase. In total this adjustable first phase increases the diversification and the probability to find the global optimum [2].

In a second phase, the identified good areas have to be examined more closely, which is known as intensification in the domain of meta-heuristics. Here, the behavior of the particles is adapted to this stage. They are now trying to find the best possible solution in the subspace of the whole solution space that is being reinforced by an increased concentration of particles in this area. Furthermore, through formation of subpopulations reinforcement of behavior is achieved. The basic idea here is to examine defined regions of the search space in more detail and therefore not to be affected by disturbing influences of other areas. After a further, well chosen number of iterations, the areas can now be compared. Is the range on average substantially worse, it can be neglected; the parameterized essentials can be set by the decision maker.

As a part of the particles was filtered, the solution quality can be increased, with more iteration at the same computational effort as before. Because of the convergence properties of meta-heuristics few iteration at the beginning lead to some significant improvement in the objective value, while the last iterations achieve only marginal improvement. In the third and final phase, with a reduced number of particles, an improved result is searched as long as a predetermined number of iterations is not exceeded or improvement of a certain percentage level can be achieved.

IV. Implementation

A. Standard PSO-algorithm

The implementation of the PSO-algorithm was on a pre-implementation by Cingolani [5]. This is only an implementation of the initially described PSO algorithm and will be referred to as the ‘Standard PSO’. The basic algorithm consists of a swarm *S* of particles x_i, and a fitness function $f(x_i)$. The swarm itself is a set of particles with a position vector

$$\vec{x}_i = (x_{1i}, x_{2i}, ..., x_{ni}),$$

where *i* denotes particle x_i and x_1 the index of the vector component. The velocity

$$\vec{v}_i = (v_{1i}, v_{2i}, ..., v_{ni})$$

is defined equally. The particles also consist of a memory of their personal best found position p_i so far as well as the current fitness value. The bound for every component of the position form the search space and all positions within are valid solutions, thus the solution space. For the swarm however, the global best found position p_g as well as the global best found fitness are stored together. The position represents an allocation of numbers, which serves as input variables for the fitness function (here: the specific simulation model to be evaluated). The result of the fitness function is the fitness of the particle (here: the resulting simulation models KPIs). At the beginning of the optimization run, the positions are assigned randomly within the solution space. Then all positions are evaluated. The difference between the actual and the best position within the swarm results in a direction – the velocity. With this velocity a new position is calculated and the first iteration finishes. The next iterations form a cycle of position evaluation, velocity calculation and repositioning the particles. This is executed until stopping criteria is fulfilled.

In every iteration the global best position is evaluated which is formally defined as follows (here for a minimization, for maximization it would be *arg max*$\{\ldots\}$):

$$p_g^k = \left\{\left\{p_1^k,\ldots,p_n^k\right\} \middle| p_i^k = \arg\min\left\{f(p_1^k),\ldots,f(p_n^k)\right\}\right\}$$

The velocity is calculated per vector component:

$$v_{id}^{k+1} = v_{id}^k + c_1 r_1 (p_{id} - x_{id}^k) + c_2 r_2 (p_{gd} - x_{id}^k),$$

where r_1 and r_2 are random numbers within an interval [0 ; 1], which results in a randomness of the particles movement. They can be calculated once and be applied as a scalar to the vector or calculated per component. c_1 and c_2 are called acceleration coefficients and as a result from their concrete assignment, it is determined whether the particle is attracted stronger to his personal or to the global best position.

$$c_1 r_1 (p_{id} - x_{id}^k)$$

is also called the *cognitive* and

$$c_2 r_2 (p_{gd} - x_{id}^k)$$

the *social* component. If the velocity is also set at initialization it can be restricted to keep within the solution space after the first iteration.
With a new velocity vector the positions can be refreshed as follows:

$$\vec{x}_i^{k+1} = \vec{x}_i^k + \vec{v}_i^{k+1}.$$

Since the particles move free through the solution space it must be taken into consideration, that recalculated positions could be out of range. Most optimization problems are defined by restrictions, which are predetermined. A possibility to resolve these cases are so called walls (dimension limits). When the particles encounter the dimension limits, multiple methods exist, to set the position of the particle back within the solution space. Three methods are common and have been implemented [14]:

- Absorbing Walls: When a particle hits one of the dimensions limits, the speed in this dimension is set to 0. This creates the possibility that particles are set back into the solution space.
- Reflecting Walls: When a particle comes up against one of the limits, the algebraic sign of the velocity component in this dimension is inverted. The particle is set back in the direction of the solution space.
- Invisible walls: In this method, the particles can exceed the dimension limits. Particles that are outside these limits will not be evaluated. The motivation of this method is to save computational effort

Another common extension is *inertia weight* formulated by Shi and Eberhart [15]. Inertia weight is a factor and multiplied with previous speed of a particle v_{id}^k to contain explosion of the swarm, meaning that the velocity rises without limit. Inertia weight is by implication meant to accelerate the convergence of the swarm. An extended velocity formula results as follows:

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 r_1 (p_{id} - x_{id}^k) + c_2 r_2 (p_{gd} - x_{id}^k)$$

Shi and Eberhart investigated ω within the interval [0 ; 1.4]. They revealed that within [0.8 ; 1.2] a quicker convergence can be achieved, while higher values (>1.2) result in a failure to converge. To small values can cause the swarm to get stuck in local optima.

Angeline [1] noticed that the classic PSO algorithm had an implicit weak *selection* of particles regarding the personal best fitness. The purpose of selection aims at placing particles in supposedly good regions of the solution space discovered earlier. Angeline proposed a method to perform an explicit selection:

- For each particle in the swarm: compare the fitness value with those of the others – each time the current particle has a better fitness it is rewarded.
- Sort particles ascending by number of rewards.
- Choose upper half of particles and copy their position to those from the lower half, while preserving the personal best position of each particle copied.

This procedure has to be performed before calculating new velocity vectors. Evangeline revealed that solution quality within local optima increased, while weakening the ability to find global optima. Consequently this extension improves the convergence of the PSO

Finding global optima requires the procedure to maintain diversity. Suresh et al. [16] proposed two modifications. One is about adjusting the inertia weight depending of the distance current global best position. They formulated it as follows:

$$\omega = \omega_0 \left(1 - \frac{\delta_i}{\delta_{\max}}\right) \text{ with } \omega_0 = rand(0{,}5\,;1),$$

with δ_i denoting the Euclidian distance in a d-dimensional space and $\delta_{\max}$ the greatest possible distance:

$$\delta_i = \sqrt{\sum_{d=1}^{D} \left(p_{gd} - x_{id}^k\right)^2}\ .$$

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-234-9

As a result Suresh et al. revealed that this adaptive adjustment substantially improved performance. As being intensively investigated, these extensions, modifications and improvements have been proposed to the standard PSO.

To address the issue of random influences in the fitness function we used multisampling as an further extension. This means that the same position is evaluated multiple times and an average fitness value is returned. The work of Bartz-Beielstein et al. [2] was a foundation. Using history data and smoothening of fitness values over a specified amount of iterations, inspired to abuse this feature in combination with the selection extension described earlier.

B. 3-Phase PSO implementation

According to the conceptual design, the implementation of the PSO algorithm was refined by the realization of the 3-Phase PSO according to the following 3 phases:

Phase I:

1. Initialization of *n* particles using intelligent positioning,
2. Initialization of the velocity vectors as *null*-vectors,
3. Evaluate fitness of each particle using distributed simulation runs and update velocity vectors with a weak attraction towards the global best position,
4. Save fitness values in history data,
5. Generate new position for each particle,
6. Repeat step 3-5 till termination criteria achieved.

Phase II:

1. Select *n* best found fitness values from history data and set particles to these positions,
2. Initialize velocity vector, aiming at the best found fitness for each group,
3. Evaluate fitness and update history data,
4. Update velocity vector,
5. Repeat 2-4 as long as improvement is gained.

Phase III:

1. Identify best group of particles,
2. Delete other particles,
3. Initialize velocity vector, aiming at the overall best found position,
4. Evaluate fitness and update velocity vectors,
5. Repeat, until criteria for termination are realized.

The selection feature presented in [1] was investigated with the result that the 3-Phase-PSO was stuck in a local optimum. The intelligent initialization with the greatest possible dispersion strongly reduced this risk. Here, the particles are scattered evenly in the feasible solution space so that the swarm of particles is provided with a rough overview. An original random initialization of the starting positions could lead to the fact that all particles concentrate only in a certain region of the solution space. The remaining solution space would not be evaluated. Selection in conjunction with an intelligent initialization and an adaptive swarm behavior can strongly reduce the number of fitness evaluations. The objective of the fast converging procedure needs to quickly identify good areas and to concentrate the particles in these regions and to achieve good improvements in fitness in as few iterations as possible, without limiting the global search.

C. Material flow simulation with d³fact

d³fact is a discrete, event–based material flow simulation framework, designed and implemented at the Heinz Nixdorf Institute of the University of Paderborn, Germany. Designed as a multi-user environment, it allows simultaneous, collaborative modeling and simulation of a model by multiple simulation experts. d³fact consists of a modeling tool, a simulation server, that runs the simulation and few visualization options from 2D to 3D. The freeware software is based on the Eclipse Rich Client Platform (RCP) and is implemented in Java [6],[13].

For a first evaluation of the feasibility study a rather simple simulation model was selected, which is presented in Figure 2. Reason for such a selection was to crosscheck the conformability of the generated solutions by the designed procedure with a simple reference model. The resulting parameter configurations could easily be evaluated by just logical thinking.

Another advantage of this simple reference simulation model was a significant lower amount of necessary simulation time for each evaluation. Since all developed improvements of the standard PSO-algorithm, the 3-Phase-PSO and overall procedure had to be evaluated by multiple simulation runs, the evaluation could be limited in time.

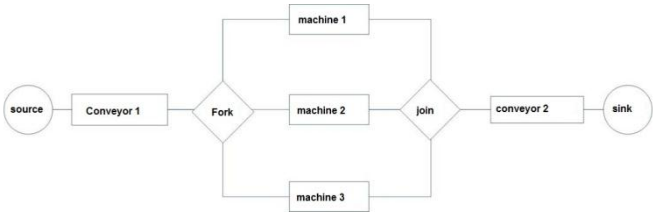
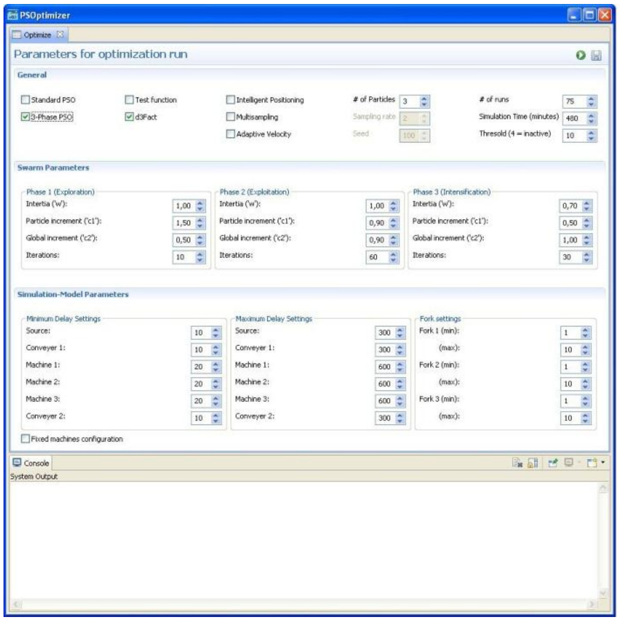


Figure 2. Simple material flow reference model in d³fact.

D. Graphical user interface

For a more user-friendly evaluation of the different configuration parameters of the PSO as well as the 3-Phase-PSO algorithm, a simple, graphical user interface was developed. Based on the initial simulation model, the parameter limits for the simulation configuration set as well as the PSO algorithms can be configured. Based on that, out of the GUI, the overall procedure process is started. The application starts the simulation runs with each particle's simulation configuration and shows the simulation results. Figure 3 shows a screenshot of the graphical user interface for the procedure testing.



Parameters for optimization run

General

☐ Standard PSO
☑ 3-Phase PSO
☐ Test function
☑ d3Fact
☐ Intelligent Positioning
☐ Multisampling
☐ Adaptive Velocity
# of Particles: 3
Sampling rate: 2
Seed: 100
# of runs: 75
Simulation Time (minutes): 480
Thresold (4 = inactive): 10

Swarm Parameters

Phase 1 (Exploration)
Intertia ('w'): 1,00
Particle increment ('c1'): 1,50
Global increment ('c2'): 0,50
Iterations: 10

Phase 2 (Exploitation)
Intertia ('w'): 1,00
Particle increment ('c1'): 0,90
Global increment ('c2'): 0,90
Iterations: 60

Phase 3 (Intensification)
Intertia ('W'): 0,70
Particle increment ('c1'): 0,50
Global increment ('c2'): 1,00
Iterations: 30

Simulation-Model Parameters

Minimum Delay Settings
Source: 10
Conveyer 1: 10
Machine 1: 20
Machine 2: 20
Machine 3: 20
Conveyer 2: 10

Maximum Delay Settings
Source: 300
Conveyer 1: 300
Machine 1: 600
Machine 2: 600
Machine 3: 600
Conveyer 2: 300

Fork settings
Fork 1 (min): 1
(max): 10
Fork 2 (min): 1
(max): 10
Fork 3 (min): 1
(max): 10

☐ Fixed machines configuration

System Output

Figure 3. Screenshot of the user interface PSOptmizer.

V. RESULTS

To ensure the correct operation of the PSO, a mathematical test function has been used initially. The functions global optimum is predictable by analytical methods. It also has multiple local optima. The evaluation of the simple simulation model was realized in d³FACT. The material flow network is built at runtime with a model configuration, which corresponds to the starting position of the particle in the solution space. The solution space is defined on the GUI in the ranges of model parameters. After establishing the model, it is simulated for the set duration and the material flow is returned as fitness to the PSO.

The test runs were conducted on several computers with two core processors and four gigabytes of memory. There were as many instances of the PSOptimizer started as cores available. These were assigned to a core to reduce losses through the operating system scheduler. Note that it could still come to computational losses due to simultaneous access to the working main memory. Run time played a minor role in these first investigations and therefore such losses have been neglected. Early trials showed, that there can be significant variations in the best objective value test run. This was caused by the stochastic factors in the model as well as in the PSO. So, each configuration was calculated 200-times to get a sample, which then was evaluated by statistical methods. The arithmetic mean for the iterations of the algorithm was calculated as the best achieved objective function values. For this, the GAP could be determined.

A. *Mathematical test function*

Besides demonstrating the feasibility of the designed procedure, a first investigated objective was to evaluate the PSO-extensions according to the reduction of necessary iterations, especially in conjunction with the simulation environment. Here, the focus was set on the PSO algorithm so that a minimum possible number of iterations was necessary for the best possible solution. In order to allow a fast evaluation of the PSO-algorithm configuration sets, a simple mathematical function was used:

$$f(x, y) = \sin(x) + \sin(y) + 0.1x + 0.2y$$

Table 1 shows some resulting GAP-values of the implemented PSO-algorithms according to the test function:

TABLE I. RESULTING GAP FOR MATHEMATICAL TEST FUNCTION

Error for test function				
particle	**standard PSO**	**standard PSO (int. position.)**	**3-Phase PSO**	**3-Phase PSO (int. position)**
6	4,64%	0,65%	2,39%	0,25%
9	2,25%	0,42%	1,44%	0,12%
11	1,03%	0,35%	0,42%	0,07%
22	0,30%	0,10%	0,10%	0,03%
33	0,26%	0,06%	0,01%	0,01%
GAP (mean)	**1,70%**	**0,32%**	**0,87%**	**0,10%**

The most important and up to date new extension was the intelligent positioning of a set of particles in the solution space. This option significantly increased solution quality and nearly zeroed the possibility of getting stuck in a local optimum. Figure 4 displays the impact of this feature in iteration counts.

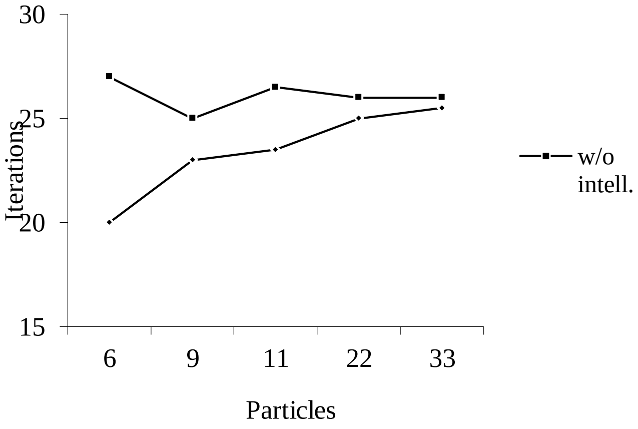


Figure 4. Iteration count for Standard PSO on test function.

B. *Reference simulation model*

The model was deliberately kept simple to determine the theoretical maximum throughput by analytical means. This eased an investigation of GAPs. The design of the experiment and the data obtained were analyzed, processed and are presented below. In addition to the preset maximum iteration number, a threshold as a criterion was selected. For the analysis presented below the threshold was set at ten.

Table 2 shows some resulting GAP-values of the implemented PSO-algorithms according to the simulation model in d³fact.

TABLE II. RESULTING GAP FOR D³FACT SIMULATION MODEL

Error for simulation				
particle	**standard PSO**	**standard PSO (int. position.)**	**3-Phase PSO**	**3-Phase PSO (int. position)**
6	10,47%	9,00%	8,80%	4,79%
9	7,27%	3,51%	5,75%	0,77%
11	6,85%	0,60%	3,86%	0,75%
22	2,61%	0,26%	1,42%	0,07%
33	5,70%	2,82%	4,11%	1,29%
GAP (mean)	**6,58%**	**3,24%**	**4,79%**	**1,54%**

C. Conclusion of results

Figure 4 demonstrates that the swarm parameters chosen for every phase are far from best in term of iteration count. Considering the GAP values listed in table 1 and table 2 the goal has been fulfilled. In terms of solution quality, it has to be considered that the random input in the swarm results in random behavior of the particles. In one run the particles may have more “luck” than the next. In conjunction with simulation based optimization such a case is inconceivable. Since the possibility of such a result is nearly zeroed and result quality can be raised at least without increase in total runtime the presented options still perform quite well.

However, not all proposed extensions lead to the desired reduction in iteration count. The adaptive velocity option seemed to slow down the algorithm especially in when the number of particles is low. It turns out that the PSO can manage the velocity of the particles better by itself than when interfering with a bound.

Since the swarm parameters were chosen with respect to the quality of the solution and a parameter study for those was not conducted at this point, potentials for improvement may left open. Another reason for this result is that the swarm parameters in the distinct phases were set to values that are boundaries of internals that have been investigated by researchers earlier [15].

Due to model characteristics the difference between both algorithms is not big as depicted in Figure 5. The models optimal solution lies at the fringe of the solution space. The implemented restriction violation procedure fosters the identification of solutions in these regions.

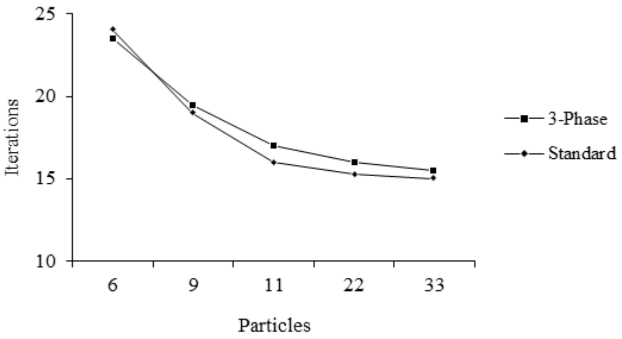


Figure 5. Iteration count for 3-Phase vs. Standard PSO with d³fact.

Not all modifications introduced had the desired success. The test runs, however, show a first conclusion: the configuration of the swarm is not arbitrary. Parameters such as population size and the choice of the threshold have to be done carefully. Population size is a key success factor for the necessary effort and solution quality. A higher threshold does not significantly increase the solution on average, but the effort is considerably. The investigated threshold was 10 and 20. The difference between the GAPs for all standard test runs of these two values is only 0.17%, while the average duration almost doubled.

The solution quality can be improved significantly by activation of intelligent positioning (cp. Figure 6). The studies show that the particle number and dimensionality may cause weaknesses. This plays a particular role as the last two particles are set. As a possible improvement, those two positions should always be occupied.

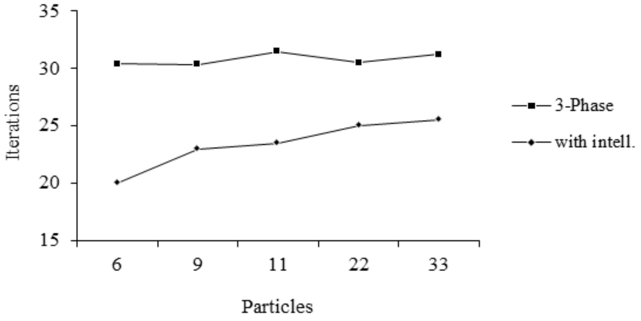


Figure 6. Iteration count for 3-Phase vs. Standard PSO with the test function.

The studies have also shown that the implemented selection in the 3-phase algorithm in conjunction with a history of positions leads to an improvement of GAP. It can also be shown that the question whether the selection should be performed several times remains open. Potentials for improvement exist. An investigation regarding the swarm parameters was impossible till today due to the complex combination possibilities. There is need for further experimentation as the swarm parameters implemented in each respective phase are adapted according to the behavior of the swarm, but their quality remains unclear at this stage. Overall, the studies show that the 3-phase PSO-algorithm construction is a solid basis for further developments.

VI. OUTLOOK

With the implemented toolkit, it has been successfully shown that simulation models, in particular the configuration of parameters, can be optimized by a combination of simulation and the meta-heuristic particle swarm optimization. The test model differs in its complexity still far from real-life simulation models, since simulation itself is only useful for such complex models that deny an exact mathematical analysis. Nevertheless, as a proof of concept, the feasibility was shown in principle with the existing prototype.

Initial findings are collected on the behavior and the duration of the 3-phase PSO. It turns out that this first draft can be furthermore improved. Potential still exists in a more intensive use of selection and a better adaptation of the swarm parameters itself in the various phases; this would again improve the needed computational time. The solution's quality has already been significantly improved by the introduced modifications, described and presented in this paper. Further calculations and tests on the PSO-algorithms parameter setting are to be made in near future.

Another desirable future work was the evaluation of other heuristics as the optimization component in this approach, e.g., genetic algorithms. Even the combined approach of multiple heuristics for the optimization of the parameters could be taken into consideration.

ACKNOWLEDGEMENTS

This work was significantly influenced by the student work of Barthold Urban as well as the strong support of Markus Eberling, PhD-student at the chair of knowledge-based systems at the University of Paderborn.

REFERENCES

[1] P. Angeline, Evolutionary optimization versus particle swarm optimization: Philosophy and performance differences. In V. W. Porto, N. Saravanan, D. Waagen, & A. E. Eiben (Eds.), Proceedings of evolutionary programming VII, 1998, pp. 601–610, Berlin, Springer.

[2] T. Bartz-Beielstein, D. Blum and, J. Branke, Particle swarm optimization and sequential sampling in noisy environments. In Metaheuristics International Conference, edited by R. Hartl and K. Doerner. 2005, pp. 89-94. University of Vienna.

[3] F. van den Bergh, An Analysis of Particle Swarm Optimizers. Ph.D. thesis, University of Pretoria, Pretoria, South Africa, 2002

[4] M. Clerc and J. Kennedy. The particle swarm - explosion, stability and convergence in multi-dimensional complex space. In IEEE Transactions on Evolutionary Computation. Volume 6. 2002, pp. 58-73, Washington, IEEE Computer Society.

[5] P. Cingolani, Open source particle swarm optimization library written in Java. Available via <http://jswarm-pso.sourceforge.net/> [accessed November 8, 2012].

[6] W. Dangelmaier 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. 2007, DUV Verlage.

[7] F. Heppner and U. Grenander, A stochastic nonlinear Model for coordinated Bird Flocks. In The Ubiquity of Chaos, edited by E. Krasner, 1990 pp. 233-238. AAAS Publications.

[8] J. Kennedy and R.C. Eberhart, Swarm Intelligence, San Francisco, Morgan Kaufmann Publishers Inc, 1995

[9] A.M. Law and W. D. Kelton, Simulation modeling & analysis. 3rd ed, New York: McGraw-Hill, Inc., 2000

[10] Leandro dos Santos Coelho, A quantum particle swarm optimizer with chaotic mutation operator, In: Chaos, Solitons & Fractals: Volume 37, Issue 5, 2006 pp. 1409-1418.

[11] D.C. Montgomery, Design and Analysis of Experiments, 7E International Student Version, John Wiley & Sons, Limited, 2008

[12] R.L. Rardin, Optimization in Operations Research, Prentice Hall, 1997

[13] H. Renken, C. Laroque and, M. Fischer, An Easy Extendable Modeling Framework for Discrete Event Simulation Models and their Visualization. In: Proceedings of The 25th European Simulation and Modelling Conference (ESM'2011), 2011

[14] J Robinson and Y. Rahmat-Samii, Particle Swarm Optimization in Electromagnetics. In Journal of Artificial Evolution and Applications. 2008, doi:10.1109/TAP.2004.823969

[15] Y. Shi and R. C. Eberhart, A Modified Particle Swarm Optimizer. In Proceedings of World Congress on Computational Intelligence, 1998, pp. 69-73. Anchorage, IEEE Computer Society.

[16] K. Suresh, S. Ghosh, D. Kundu, A. Sen, S. Das, and A. Abraham, Inertia-Adaptive Particle Swarm Optimizer for Improved Global Search, In Proceedings of the 2008 Eighth International Conference on Intelligent Systems Design and Applications. Volume 2, 2008, pp. 253-258. Washington, IEEE Computer Society.

Agent-Based Simulation and Cooperation in Business Organizational Settings

Claudia Ribeiro, José Borbinha, José Tribolet, João Pereira
INESC-ID
Rua Alves Redol 9 1000-029 Lisbon, Portugal
Department of Information Systems and Computer Science, Instituto Superior Técnico
claudia.sofia.ribeiro, jlb, jose.tribolet, joao.madeiras.pereira@ist.utl.pt

***Abstract*—The object of this paper is to use Agent-Based Simulation (ABS) to study the effects of cooperation in business organizational settings. To model the functioning of a business organization we have used an Enterprise Engineering approach named Design & Engineering Methodology for Organizations (DEMO). DEMO is based on the Ψ-theory which has the overall goal to extract the essence of an organization from its actual appearance. This theory assumes that an organization is a system of actors and incorporates four axioms. The operation axiom tells us that the implementation independent essence of an organization consists of actor roles and that the acts performed by the actor roles can be divided into two kinds: production acts and coordination acts. Another important axiom is the transaction axiom which states that coordination acts are performed as steps in universal patterns. Based on these assumptions and Game Theory principles of cooperation, a simple ABS was develop focused on studying the conditions that allow cooperation to emerge. By understanding these conditions, appropriate actions can be taken to foster the development of cooperation in such settings.**

***Keywords*-Agent-Based Simulation; Cooperation; Enterprise Engineering; Game Theory.**

I. Introduction

There is an increasing level of dynamics and uncertainty characterizing organizations and their environments. Consequently, contemporary organizational thinking has evolved to embrace paradigms supported by Complexity Theory and its principles. Complexity-based paradigms replace deterministic perspectives of the internal and external workings of organizations by perspectives based on emergence, self- organization and evolution [1]. In these paradigms, organizations are regarded as Complex Adaptive Systems that emerge from the interactions among human and non-human agents.

Complexity Theory involves the study of many actors and their interactions. The actors may be atoms, fish, people, organizations, or nations. Their interactions may consist of attraction, combat, mating, communication, trade, partnership, or rivalry. One of the central topics regarding interaction between self-interested agents is cooperation. Cooperation is crucial for societies and organizations, since it allows the creation of common goods that no single individual could establish alone. However, this situation itself presents a dilemma, because as the creation of these goods requires an individual effort and the result is shared by everyone, there is the temptation to make an individual contribution as little as possible and receive as much of the result as one can. The problem of how can cooperation emerge in a organization of self-interested individuals is one of the central questions addressed by Game Theory, Political Science and Behavioural and Evolutionary Economics.

The study of large number of actors with changing patterns of interaction often gets too difficult for a mathematical solution, therefore other type of solutions need to be used. A primary research tool of Complexity Theory is computer simulation. The basic underlying function of this tool is to specify how the agents interact, and then observe properties that occur at the level of the whole organization.

The simulation of agents and their interactions is known as Agent-Based Simulation (ABS). The goal of ABS is to enrich our understanding of fundamental processes that may appear in a variety of applications. This is the assumptions underlying the proposal described in this paper. To represent the functioning of an organization DEMO's Ψ-theory [2], [3], [4] was used. This theory combines concepts from the Language/Action Perspective (LAP) [5] and Speech Act Theory [6]. The Ψ-theory explains how and why people cooperate and communicate. It postulates that the operation of an organization can be expressed by a specification of the commitments that the organizational subjects enter into and comply with [7]. Based on this theory and concepts developed in Game Theory, this paper proposes an ABS with an underlying conceptual model that allows to experiment and analyse the different patterns that emerge when organizational subjects use different kind of strategies to handle commitments to produce organizational output.

We start by describing (Section 2) two of the most influential games studied as the best representation of the problem of social cooperation. These are respectively the Prisoner's Dilemma and the Stag Hunt. Also, in this section, some conclusions are drawn about what are the central concepts and conditions that promote cooperation to emerge. Next (Section 3), we explain and analyse how organizations are modelled using DEMO's Ψ-theory. Based on what was presented in Section 2 and 3, a very simple case study is presented (Section 4). Also, in this section, some of the potentials and limitations of the presented solution are discussed. Finally, (Section 5) conclusions are drawn pointing to the future scope for development that lies ahead on this vast and interesting field.

II. Cooperation

The evolution of cooperation has been largely studied in the research field of Game Theory [8], [9]. Game Theory studies what happens when self-interested agents interact. Self-interested agents are agents that have their own beliefs, preferences and actions as opposed to teams where some of these characteristics are shared among the group [10]. The assumption of self-interested therefore allows an examination of the difficult case in which cooperation is not completely based upon the concern for others or upon the welfare of the group as a whole. So, the assumption of self-interest is really just an assumption that concern for other does not

ISBN: 978-1-61208-234-9

completely solve the problem of when to cooperate and when not to. Two of the most studied games in this context is the Prisoner's Dilemma [8] and the Stag Hunt [11] game.

The Prisoner's Dilemma is a game that shows why two individuals might not cooperate, even if it appears that it is in their best interest. Axelrod [8] states that a similar problem occurs in many similar situations where the pursuit of self-interest by each agent leads to a poor outcome for all. The Prisoner's Dilemma game is a general representation of such situation. Axelrod has explored the conditions in which cooperation would emerged by promoting a computer tournament where people could submit their favourite strategy to play the iterative version of the prisoner's dilemma game. The winner for the rounds of the tournament was Anatol Rapoport [12] that submitted the very well know TIT FOR TAT strategy that was based on reciprocity. This very simple strategy that consisted in cooperating in the first move and then doing whatever the other player did in the previous move overcome complex strategies based on Markov processes and Bayesian inference. After analysing the data that resulted from this tournament Axelrod concluded the following. The evolution of cooperation requires that individuals have a sufficiently large chance to meet again so that they have a stake in their future interaction and also that cooperation be based in reciprocity.

The Stag Hunt [11] is a prototype of the social contract [13]. Like in the Prisoner's Dilemma game, each player must choose an action without knowing the choice of the other. If an individual hunts a stag, he must have the cooperation of his partner in order to succeed. An individual can get a hare by himself, but a hare is worth less than a stag.

Skyrms argues that the Stag Hunt does not have the same melodramatic quality as the Prisoner's Dilemma but instead raises its own set of issues. When comparing a two-person Stag Hunt with a two-person Prisoner's Dilemma he noticed the following. If two people cooperate in the Prisoner's Dilemma, each is choosing less rather than more. Specifically, there is a conflict between individual rationality and mutual benefit. In the Stag Hunt, what is rational for one player to choose depends on his beliefs about what the other will choose. The Stag Hunt differs from the Prisoner's Dilemma in that there are two Nash equilibria: when both players cooperate and both players defect. In the Prisoner's Dilemma, in contrast, despite the fact that both players cooperating is Pareto efficient [14], the only Nash equilibrium is when both players choose to defect.

The existence of those two Nash equilibria is just to say that it is best to hunt stag if the other player hunts stag, and it is best to hunt hare if the other player hunts hare. Therefore, it is clear that a pessimist, who always expects the worst, would hunt hare. But it is also true with these pay-offs that a cautions player, who was so uncertain that he though the other player was as likely to do one thing as another, would also hunt hare. Hunting hare is said to be the risk-dominant equilibrium. That is not to say that rational players could not coordinate on the stag hunt equilibrium that gives them both a better pay-off, but it is to say that they need a measure of trust to do so.

A. *Conclusion*

When observing cooperation from a prisoner's dilemma point of view it was concluded that for cooperation to emerge certain conditions have to hold true. Specifically, a player will be more likely to cooperate with another player if there is a high probability of interacting with that player in the future. The same applies to the stag hunt since a repeated prisoner's dilemma is equivalent to a two-person stag hunt. Another important conclusion is that a strategy based on direct reciprocity can overcome even complex strategies. Therefore, if cooperation is established on the basis of direct reciprocity it will endure even if non-cooperative strategies exist. Indirect reciprocity should also be consider and has been identified has an equal important factor that promotes the emergence of cooperation. Indirect reciprocity, also denominated reputation, states that if a player is known as being cooperative than other players more likely will cooperate with him. Basically, what this means is that cooperation can emerge either by inducing a certain behaviour in direct relationships but also through indirect relationships. From the stag hunt game several important conclusion could also be extracted. Skirms argued that the viability of cooperation depends on mutual beliefs, and rest on trust. Therefore, trust is a central concept when studying cooperation and was the property chosen in the case study. We believe that even if mutual beliefs don't exist or if the players are unaware of each other beliefs, cooperation will nevertheless emerge if the level of trust is higher enough.

III. Design and Engineering Methodology for Organizations (DEMO)

DEMO is the approach that supports our proposal for modelling the functioning of an organization. Its roots are in the Ψ-theory, which provides an explanation of the construction and operation of organizations based on four axioms thus contributing to the strong theoretical ground that ensures the formal correctness of its models [2], [3], [4]. DEMO's methodology has already been use successfully in a large number of projects [15], [16], [17].

The operation axiom states that the operation of an organization is constituted by the activities of actor roles, which are elementary chunks of authority and responsibility, fulfilled by subjects. In doing so, these subjects perform two kinds of acts: production acts (P-acts) and coordination acts (C-acts). By performing P-acts, the actors contribute to bringing about the goods or services delivered to the environment of the organization. However, by performing C-acts, actors enter into, and comply with, commitments and agreements towards each other regarding the performance of P-acts.

In this paper, our focus is to provide a systematic way to study both the effects of cooperation in organizational performance and also what are the conditions that foster or promote cooperation. In this manner, we are not so concerned about how a P-act is performed but we put more stress in the C-acts since these are the ones that are focused on the social interactions of the intervening actors.

A coordination act is an act performed by one actor, called the performer, and directed to another actor, called the addressee (Figure 1). It consists of two concurrent acts, the intention act and the proposition act. In the intention act, the performer proclaims its ”social attitude” with respect to the proposition. This term was defined by Searle [6] and Dietz [3] used it to distinguish coordination acts from communicative acts in general.

The actors are the active elements of an organization and they are the only ones Dietz considers when modelling the functioning of an organization. Therefore, machines or any other artificial systems

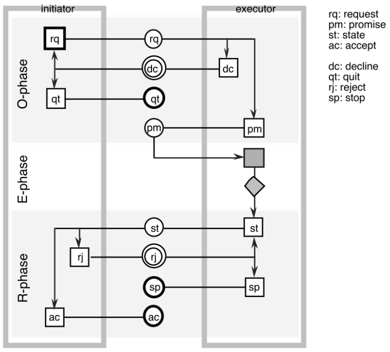


Figure 1. DEMO Transaction Axiom [3]

are consider as supporting actors an never replacing them. Actors are not triggered by events, instead they constantly loop through the actor cycle, in which they deal with their agenda. An *agendum* is a C-fact with a proposed time for dealing with it, to which actor is committed to respond. Action rules guide the actors in dealing with their agenda; there is an action rule for every type of *agendum*. The addressees of these C-acts are other actors, which means that the resulting C-facts are added to the agenda of these other actors. In this way, actors keep supplying each other with work. Dietz described the behaviour of actors as described in common organizational theories where three main factors are taken into account: responsibility, authority and, competence. In this paper we broaden this notion to include other factors borrowed from Game Theory, namely reciprocity, reputation and trust. This is further explained in Section IV-A.

Another important notion we have retrieve from the Ψ-theory is how the C-act are performed and this is explained by the the transaction axiom. This axiom states that C-acts are performed as steps in universal patterns. These patterns, also called transactions, always involve two actor roles (initiator and executor) and are aimed at achieving a particular result. A transaction is developed in three phases: the order phase (O-phase), the execution phase (E-phase), and the result phase (R-phase). In the O-phase the two actors agree on the expected result of the transaction; in the E-phase the executor executes the production act needed to create the anticipated result; and in the R-phase the two actors discuss if the transaction result is equal to the expected one. The general transaction pattern is shown in Figure 1.

In this general pattern, the course that is taken is when the initiator and the executor keep consenting to each other's acts. However, they may also dissent. There are two states where this may happen, namely "requested" and "stated". Instead of promising, one may respond to a request by declining it, and, instead of accepting one may respond to a statement by rejecting it. The reason for declining a request by the executor of a transaction or for rejecting a statement by the initiator is in principle a mixture of the three validity claims. These validity claims where defined by Habermmas [18] and are respectively, claim to truth, claim to justice and claim to sincerity.

This is central to our proposed ABS because since the transaction pattern is fixed, then, we can change how the behaviour of the actors involved is characterized and see the impact of such changes when performing this general transaction pattern.

IV. Agent-Based Simulation (ABS)

Formally, ABS is a computational method that enables a researcher to create, analyse, and experiment with models composed of diverse and heterogeneous agents that interact within an environment [19].

The underlying assumption for using ABS to model reality in organizations is to view organizations as Complex Adaptive Systems that emerge from the interactions among human agents. In this context, to take an agent-based approach means not having to assign an objective to an organization and instead model the agents that comprise it with explicit attention to their individual behaviours and how they interact with each other and the environment.

Agent-based models are characterized by the following [20], [21], [22]:

- A set of agents, their attributes and behaviours. The behaviour can be either according to rational models, behavioural models or rule-based models.
- A set of agent relationships and methods of interaction: An underlying topology of connectedness defines how and with whom agents interact.
- The agents environment: Agents interact with their environment in addition to other agents.
- Model outcomes: simulating a set of agents interacting in an environment provides insights into phenomena related to the part of reality being simulated.

Next section its describe how an ABS can be built using the assumption of DEMO's Ψ-theory and the conclusions drawn from the Prisoner's Dilemma and Stag Hunt game. Also, a simple instantiation of the model was implemented in Netlogo [23].

A. ABM and Cooperation in Business Organizational Settings

The basic units of the model proposed to systematically study the effects of cooperation in organizational performance are a set of agents that can be of two particular types: initiators and executors. These two types correspond to the two actor roles described in DEMO Transaction axiom. These two roles can either represent individual people, groups of people or even different organizations. As described in Section III each of these roles has a particular set of actions that define their behaviour in the context of a transaction as expressed in Figure 1. Combinations of these actions represent different possible paths and results that can happen while the two actor roles interact.

Dietz has a lot more subtleties concerning the Transaction axiom than what is explained here. But, one of our concerns was to keep the model as simple as possible, because the essence of modelling is to simplify things and also, as long as no vital conceptual features are lost, simplicity is the best modelling strategy. Also, the complexity of agent-based modelling should be in the simulated results, not is the assumptions of the model. In this manner when surprising results occur, it will be possible to understand everything that went into the model.

Dietz does not explicitly integrate in DEMO's models how cooperation factors impact the execution of a transaction. Namely, nothing is said about how trust affects or influences what is the actions the initiator or executor chooses while performing a

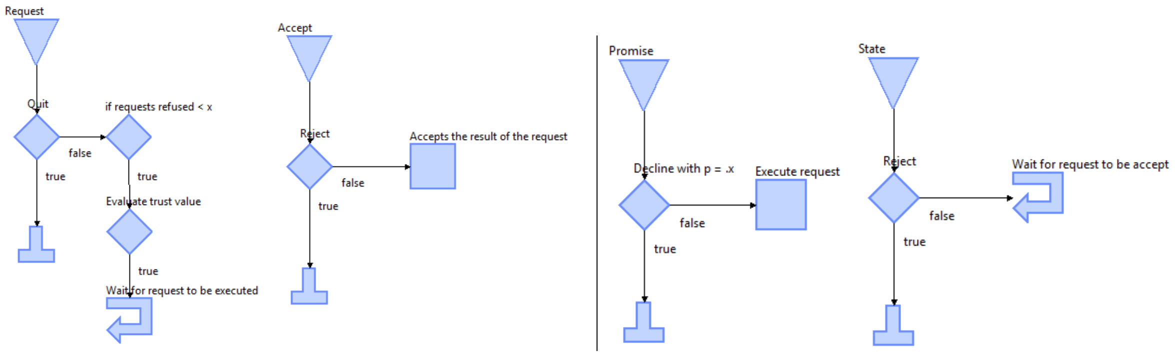


Figure 2. Initiator actions and behaviour (left). Executor actions and behaviour (right)

transaction. Instead, is assumed that the actors involved in an interaction do a ”best effort” while performing the transaction. In reality this is not always the case. The individuals that fulfil actor roles in business contexts are self-interest in the sense that they may or might not share the same goals. Also, fostering cooperation among individuals, groups, or even nations is something very much related to culture and context. What could help foster cooperation in some situations might not work across all similar situations. This means that although Axelrod and Skyrms argues that cooperation can emerge when there’s reciprocity, attention to reputation and trust, it’s not straightforward how this could be integrate in work practices nor what are the underlying conditions to make reciprocity, attention to reputation and trust exist among interacting individuals. Therefore, it would be very interesting and important to understand how we can add this kind of factors to DEMO’s transactions and see if this could help shed some light on questions such as: ”Does the level of trust between two actors affect the time a transaction takes to be concluded?”; Does the level of trust between two actors affect the probability of futures interactions? If both actors used a reciprocity based strategy will they become more tolerant to faulty situations, like the level of quality of the executor output?

We believe that bringing together DEMO’s extensive work in modelling organizations and Axelrod’s and Skyrms wonderful work on cooperation could help answer those and other important questions. In order to do an initial test to understand the potential of bringing this body of knowledge together, we have implemented an ABS in NetLogo based on the diagram in Figure 2.

It was assumed that both an initiator and an executor can only have a maximum number of simultaneous transactions. If a initiator has reached this maximum number he will only be able to do a new request after one of the current transactions is finished. Also, we have assumed that if an executor is inactive during a maximum period of time he would be eliminated from the environment. This is a consequence of all the initiators refusing to interact with them due to low level of trust.

In this model, we have used trust to test how this would impact the number of commitments between initiators and executors and also the number of executors that would die out for being inactive too long. We have devised three simple scenarios in order to be able to compare the differences between the results. Among the different scenarios we have change the probability with which an executor would refuse a request. This probability basically traduces the level of trust between an initiator and an executor. Therefore, the higher the level of trust the higher chances of an executor to fulfil a certain commitment. The details of each scenario and the results after several simulation runs are described in Table I and Figure 3.

Scenario	Description	Results
Scenario1	executors never refuse a request $p = 1$	Equilibrium is reach very fast (number of commitments constant $\simeq 490$) and no executors die out for lack of requests ($\simeq 97.9$).
Scenario2	executors refuse a request with a probability $p = .5$	Equilibrium takes longer to be achieve, number of commitments decreases to less than a half ($\simeq 218$) executors die out for refusing requests ($\simeq 82.5$).
Scenario3	executors refuse a request with a probability $p = .2$	Equilibrium takes longer to be achieve, number of commitments is very low ($\simeq 40.7$) and most executors die out($\simeq 33$).

Table I
SCENARIOS AND RESULTS

Although the assumptions of the presented scenarios are simple, the statements describing the scenarios and the respective conclusions show the potential of using Game Theory, DEMO’s Ψ-theory and ABS to study cooperation in organizational settings.

V. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we have addressed the problem related to understanding what are the effects of cooperation in business organizational settings. We have started by describing what is cooperation and how it has been studied from a Game Theory point of view. In this context we have described two of the most studied games related to cooperation, namely the Prisoner’s Dilemma and the Stag Hunt. These games have been extensively used to understand what are the necessary factors and conditions for cooperation to emerge in a environment populated by self-interested agents. To model the social interactions between two actors we have used DEMO’s Ψ-theory. Finally, we have described how a conceptual model based on ABS which included concepts from DEMO and Game Theory could be use to study the effects of cooperation in business organizational settings. A very simple instantiation of this

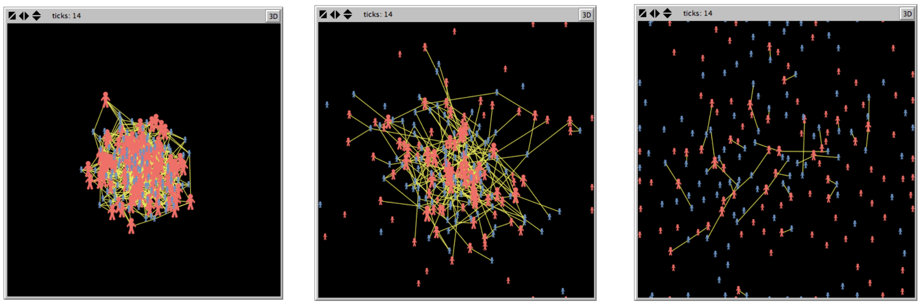


Figure 3. Scenario 1 (left). Scenario 2 (center) Scenario 3 (right)

model was used to simulate three different scenarios from which some conclusions were drawn. These conclusion were merely representative of the potential of bringing these concepts together. To be able to reach conclusive or even breakthrough insights it would be necessary to include more details and more though-out assumptions. Namely, extend the interaction to n-person instead of just 2-person transactions.

Axelrod, a part from studying the two-person Prisoner's Dilemma, he was also concerned in understanding how cooperation could emerge when many people interact with each other in groups rather that in pairs. In a n-person Prisoner's Dilemma, the dynamics that evolve to sustain cooperation are different from the two-person version. This is due to the fact that in a n-person Prisoner's Dilemma the players have no way of focusing their punishment on someone in the group who has failed to cooperate. From this realization, Axelrod developed a new game, the ”norms game”, that allowed players to punish individuals who do not cooperate. Another interesting experiment would be to add to the simulation the ability for executors to create groups among them to compete with other groups of executors. Finally, failing to cooperate sometimes is not intentional but instead the result of a misunderstanding, for example, the previous action was not understood or the current action failed to be correctly implemented. In order to also incorporate this possibility, it would be necessary to add some kind of ”noise” to the model and the ability for executors to show contrition and generosity.

Extending the interaction to n-person transactions involves considering a lot more factors that is no doubt a challenge but it could potentially provide very valuable insights to the questions raised in this paper. Also, in the possibility of these experiments reach conclusive statements about how cooperation could be promoted in such settings it could help to enhance DEMO's Ψ-theory models in order to better accommodate the social side of business and therefore bridge the gap that still exists in many of the approaches of Enterprise Engineering.

VI. Acknowledgements

This work was supported by FCT (INESC-ID multiannual funding) under the project PEst-OE/EEI/LA0021/2011. The authors also would like to acknowledge to the European funded Project Games and Learning Alliance (FP7 258169) the Network of Excellence (NoE) on Serious Games.

References

[1] R. Magalhães, *Organizational Knowledge and Technology*. Edward Elgar Publishing, 2004.

[2] J. L. G. Dietz, “The atoms, molecules and fibers of organizations,” *Data Knowl. Eng.*, vol. 47, no. 3, pp. 301–325, 2003.

[3] J. Dietz, *Enterprise Ontology: Theory and Methodology*. Springer, 2006.

[4] J. Barjis, “A business process modeling and simulation method using demo,” in *Enterprise Information Systems*, vol. 12 of *Lecture Notes in Business Information Processing*, pp. 254–265, Springer Berlin Heidelberg, 2009.

[5] F. Dignum and J. Dietz, “Communication modelling: The language/action perspective,” in *Second International Workshop on Communication Modeling (LAP 97)*, 1997.

[6] J. Searle, *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

[7] A. Caetano, A. Assis, and J. Tribolet, “Using business transactions to analyse the consistency of business process models,” *Hawaii International Conference on System Sciences*, vol. 0, pp. 4277–4285, 2012.

[8] R. Axelrod, *The Evolution of Cooperation*. Basic Books, 1985.

[9] R. Axelrod, *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton University Press, 1997.

[10] K. Leyton-Brown, *Essentials of Game Theory: A Concise, Multidisciplinary Introduction*. Morgan and Claypool Publishers, 2008.

[11] B. Skyrms, *The Stag Hunt and the Evolution of Social Structure*. Cambridge University Press, 2003.

[12] A. Rapoport and A. Chammah, *Prisoner's Dilemma*. Ann Arbor, Mich.: University of Michigan Press, 1965.

[13] T. Hobbes, *Leviathan*. London: Penguin Books, 1985.

[14] M. J. Osborne, *An Introduction to Game Theory*. Oxford University Press, 2009.

[15] A. Caetano and J. Tribolet, “On the use of a role ontology to consistently design business processes,” *Building the e-World Ecosystem*, pp. 163–176, 2011.

[16] D. Aveiro, A. Silva, and J. Tribolet, “Control organization: A demo based specification and extension,” *Advances in Enterprise Engineering V*, pp. 16–30, 2011.

[17] M. Zacarias, H. Pinto, R. Magalhães, and J. Tribolet, “A context-aware and agent-centric perspective for the alignment between individuals and organizations,” *Information Systems*, vol. 35, no. 4, pp. 441–466, 2010.

[18] J. Habermas, *The Theory of Communicative Action, Volume 1: Reason and the Rationalization of Society*. Beacon Press, 1985.

[19] N. Gilbert, *Agent-Based Models (Quantitative Applications in the Social Sciences)*. Sage Publications, Inc, 1998.

[20] M.-H. Chang and J. J. Harrington, “Agent-based models of organizations,” in *Handbook of Computational Economics* (L. Tesfatsion and K. L. Judd, eds.), vol. 2 of *Handbook of Computational Economics*, ch. 26, pp. 1273–1337, Elsevier, 00 2006.

[21] J. Miller and S. Page, *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton University Press, 2007.

[22] C. Macal and M. North, “Tutorial on agent-based modelling and simulation,” *Journal of Simulation*, vol. 4, no. 4, pp. 151–162, 2010.

[23] U. Wilensky, “NetLogo.” http://ccl.northwestern.edu/netlogo/, 1999. [Online; accessed 05-November-2012].

System Dynamics Inspired Sensor Modeling and Simulation

Sören Schweigert
OFFIS e.V.
Oldenburg, Germany
Soeren.Schweigert@offis.de

***Abstract*—Applications within the automotive and robotics domain highly depend on the correct sensor perception. In order to validate the partial safety-critical requirements for sensor processing, simulation tools are prevalent. The comparison between simulated and perceived environments allows conclusions about the quality of the sensor processing. To obtain accurate quality estimations, the simulation has to provide realistic sensor measurements. This is important in order to facilitate a subsequent integration with the physical sensors. In this work a system dynamics inspired modeling and simulation approach is presented, that allows describing both the sensors, as well as the so far often neglected environmental conditions and sensor interferences. In addition the model is capable to be transferred into a context specific shader program at simulation runtime to enable fast and efficient computation on the used graphics hardware.**

Keywords-sensor simulation*; *system dynamics*; *ground truth generation*; *error models

I. Introduction

Autonomous guided vehicles (AGV) are increasingly used in logistics and factory automation. Currently these systems operate either in closed environments like factory halls with limited access or have to operate at very low speed to ensure the safety of other participants, e.g., humans or other (human guided) vehicles. To increase the speed and thus the effectiveness, while maintaining a high safety level, the AGV handles many sensor data in order to construct an image of the environment. Some commonly used sensors for this task include both; optical sensors such as cameras, and rangefinders such as laser scanners, ultrasound or Photonic Mixing Devices (PMD cameras).

Since the sensor perception is essential for the safety aspects, it has to be comprehensively tested under various conditions. As testing in the real world is time-consuming, expensive, and especially in the first development phase risky, sensor simulations can be used for early tests. In addition simulation provides the possibility to compare the captured environmental image against the simulated environment and thus allows evaluating the sensor processing. However to obtain a realistic validation for the processing quality, the simulated sensors have to provide realistic sensor measurements, taking into account different error models and environmental influences.

The rest of the paper is structured as follows. Section II discusses requirements for sensor simulations in general and in terms of quality. These requirements are compared to currently available simulations. In Section III a modeling approach is introduced, that allows to model environmental influences on sensor measurements to fulfill the requirements discussed in Section II.

II. Requirements

To use simulation for AGV development, certain requirements must be met. Some of these requirements concern integration issues; others are related to simulation quality, yet others are useful for testing purposes.

A. Integration requirements

Simulations are used to develop AGV in a safe and convenient environment. This does speed up the development process since a lot of technical issues do not need to be considered in the early stages. However, when porting the developed algorithm from simulation to real hardware integration issues on how to **access resources** like laser scanners or cameras arise. To address this problem, most simulation environments skip along with a sensor/actor middleware that harmonizes the access to simulated and real hardware. One of the most famous representatives of this technique is the Player/Stage/Gazzebo project, started in 2000 [1]. The Robot Operating System (ROS) [2] and the Mobile Robotic Programming Toolkit (MRPT) [3], mainly use the same approach. The major drawback of this technique is that the developer is **forced to use a given architecture**.

Another approach is to recreate the interfaces of the real sensor within the simulation, including the real protocols. The described simulation uses a combination of both approaches. If possible, we recreate the interface provided by the real sensor. If the recreation is not possible, for example due to hardware limitations, a self-designed data structure is used, that allows easy transformation to other formats.

B. Testing requirements

Using simulation allows to use a user defined test environment, with all states formally described. This information should be used to provide **symbolic information** about the objects within a certain area or an **accurate occupancy grid** (containing only free or occupied cells) within the

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-234-9

environment. This information can be collected manually, as it is done in the Tunnel Simulator [4] or extracted from the simulation.
To test the quality of a certain sensor processing algorithm additional information about the sensor **ground truth** should be provided. For example the disparity map for a stereoscopic camera or a symbolic representation of all objects within a camera image.

A useful but yet not essential requirement is **real time performance**. If the simulation can provide real time sensor measurements, the performance of complex sensor processing pipelines can be measured. Faster than real-time may allow the usage of optimization or learning algorithms on both sides the sensor processing or the simulation itself. For example to learn the boundaries of the processing algorithms.
Some simulation platforms like Player/Stage/Gazzebo or the simulation component of the MRPT are using a simplification of the sensor data generation process, to gain more than real-time performance. Rossmann et al. [5] follows an interesting approach, to use VR-Hardware for efficient and accurate laser scanners. Others like the RoboCup-Rescue League simulator USARSim [6] are designed to run in real-time. Yet others like VANE [7] or the Tunnel Simulator produce highly accurate sensor measurements but cannot simulate in real-time.

C. *Quality requirements*

The most important factor for testing is the quality or accuracy of the provided sensor measurements. Since simulation tends to generate ideal measurements, real sensors do not. The following subsection will discuss some of the influences according to sensor measurement generation accuracy.

When talking about quality of virtual or simulated sensors, it has another meaning than the sensor processing community. In case of simulation, sensor quality means the availability to masquerade the existence of the simulation, by providing realistic sensor measurements, like described in Siegel et al. [8]. Within the real world there are several factors, having an impact on the quality of sensor measurements. These Factors need to be considered, when creating a sensor simulation:

1) stochastical noise
2) systematical noise
3) context sensitive noise
4) environmental influence
5) influence of other (active) sensors

Whereas the first two factors are commonly known and may be caused by thermal noise, in case of **stochastical noise**, or due to small errors in sensor-production and ageing effects, in case of **systematical noise**. (A more detailed description for these errors can be found in [9].)
The other factors are quite complex and often ignored during sensor simulation.

Context sensitive noise refers to the current situation within the simulation. Meaning, these errors result through interaction with other objects inside the simulation environment. In most cases they are reasoned by the physical principle of taking measurements. One example is an ultrasonic sensor mounted on a vehicle at low height. Even if the expected range is $6m$, the device could measure the ground at a distance of $1m$. As result the effective range for this device would be less than one meter. Similar examples for laser scanners can be found in Goodin et al. [10] where the beam divergence is taken into account. To observe these effects inside the simulation, the physical principle implemented in the real sensor needs to be matched as close as possible.

Each off these effects considers only one instance of the simulated sensor. However **environmental factors** like rain, fog and temperature do not affect only one sensor. Thus if the sensor processing detects that the range of a laser scanner is reduced because of fog, a camera near to this scanner should also indicate the existence of fog, to support reasoning algorithms.

The last effect, **influence by other (active) sensors**, can easily be observed when using the Microsoft Kinect sensor, emitting an infrared dot pattern [11]. This pattern can influence normal Webcams and thus have an important effect for computer vision algorithms that may be part of the processing pipeline that is under test.

III. Modeling approach

This section provides a general overview of the proposed modelling approach and how it is intent to solve the above discussed requirements, especially the quality requirements. A distinction is made between simulation and virtual sensors. While the simulation is used to generate an accurate representation of the environment, the sensors perceive the provided data to calculate their return values. It follows that the measured values always refer to an object in the simulated environment. In case of optical sensors it can be broken down into specific points on the surface of a three dimensional object. A prominent example is the light calculations in modern computer games which simulate the interaction between a light source and a surface. The usual approach is to manually define a computation rule for each surface. Since 3D games often consist of a variety of static and dynamic lights the defined computation rule needs to consider all kinds of light sources. Modern 3D engines like OGRE [12] use shader generators to automatically create such computation rules.

This technique should be applied to other sensors than cameras and other emitters as light sources. Since each

sensor needs a different input for a reasonable calculation of sensor measurements, a modeling methodology is introduced inspired by system dynamics. [13]
The model consists of three sub-models which are converted automatically into a computation model that is used to generate shader code during runtime of the simulation. Thereby the current context of the sensor and its environment is taken into account. The tree sub-models are:

1) Emitter Model: Defines the factor of influence (FOI) that is required to perform the calculation for at least one class of sensors.
2) Sensor Model: Defines how the specified FOI needs to be combined for one sensor class, thus defines the sensor itself.
3) Sensor Instance Model: This model allows a differentiation between two sensors of the same class. It mainly defines a post processing step, and allows to define error models for individual sensors.

A. *Emitter Model*

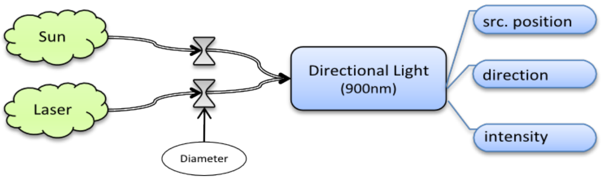


Figure 1. Example Emitter model containing two emitter classes, that do both emit directed light with a wavelength of $900nm$.

The emitter model defines an FOI that is required by at least one class of sensors as input. It is associated with all object classes which have an influence on this factor. As can be seen in Figure 1. Each factor may consist of several attributes. In the example, the directed light is determined by its source position, direction and intensity. An emitter is a class of objects that have an impact on the FOI in a certain area. They serve as global input variables in the model and thus represent the parameter of the simulation. In most cases they are associated with a 3D object within the simulation containing some basic properties like position, and orientation. Their emitted values can be described by either a constant value or a temporally resolved function. In Figure 1, there are two FOI, the sun that represents a temporarily resolved function, which provides intensity over one day, and the Laser range finder, that emits a light beam of constant intensity. The auxiliary node describes how the emitted value can be transformed into the FOI. This is done by functions for each attribute of the FOI.

B. *Sensor Model*

The sensor model describes how the FOI interacts with a surface. As input the previously defined FOI, properties of the surface and optionally further emitters can be used. The result is a new surface property that can be perceived by the sensor. Figure 2 shows an example for the sensor model. It contains the FOI defined in Figure 1, and an additional emitter for fog. To specify the interaction with the surface some of the basic surface properties like normal, reflection and position need to be known. These properties can either be defined as scalar or per point on the surface by using a texture, as usual in shader programming. This allows to model high precise sensor responses like described in [5].

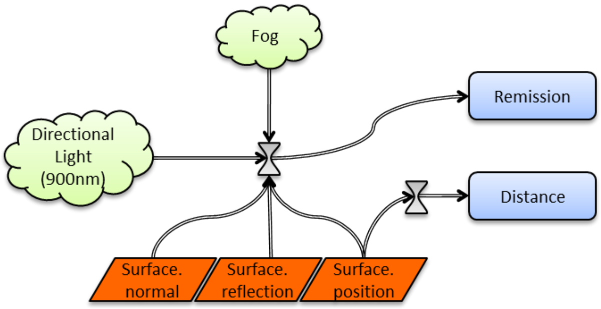


Figure 2. Example Sensor model that represent a simple laser range finder.

C. *Sensor Instance Model*

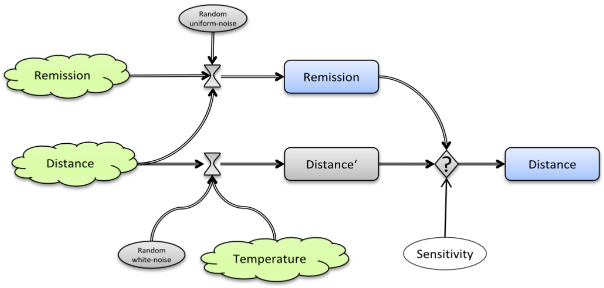


Figure 3. Example Sensor Instance model for a specific laser range finder. Whereas this device is sensitive to temperature fluctuations.

The results calculated in the sensor model, refer to a class of sensors, for example laser range finder and thus are calculated for each instance equally. They can be seen as ideal measurements, without noise or instance specific errors. Within the sensor instance model, a concrete instance of a sensor can be modeled. The previously defined surface properties serve as input. Properties with the same name but a different value define the final result. Like in the model above, further optional FOI, may be used as additional input. In Figure 3 an additional symbol is introduced, the decision between Distance and the final result. In this example it is used to describe the sensitivity of the light receiver of the laser range finder. In other words, if the received light lies below a certain threshold, the measured distance will be set to unknown or zero. Also first used in Figure 3 are random number generators, to model statistical noise. To cover most of the real noise distribution, random numbers can be generated using different probability distributions and combined if necessary. This post processing step is also intended to model sensor faults, that could follow from an incorrect parameterization or calibration.

D. Computational Model

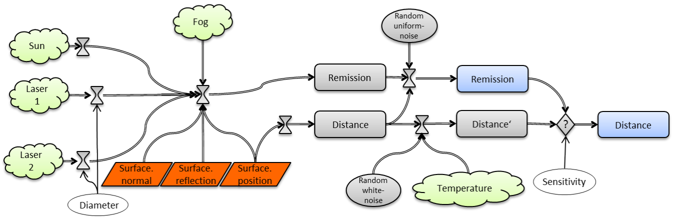


Figure 4. Generated computational model for two active sensors and three emitters of environmental influences.

Figure 4 shows the three previously defined models, summarized within the computational model. The major change is that the emitter (Directional Light in Figure 1) of the emitter model will be replaced by concrete instances of the corresponding class (Laser1, Laser2 and Sun in Figure 4) during combination. In this example, two laser scanners, the sun and fog have an impact on the observed surface. This was determined by the combination of the area observed by the sensor in charge, and the area affected by the emitters. Only those emitters will be considered which have an impact on the perceived surface properties. This is mostly described within the sensor model section.

The computational model is reviewed and newly generated whenever a sensor is intended to record new measurements. Finally the shader code is generated using the computational model in order to enable efficient computation on the graphics hardware. This needs to be done for all objects or surfaces, within the area observed by the current sensor.

IV. Conclusion and Future Work

This paper has dealt with the dynamic simulation of environmental factors and cross interferences between sensors. First, the requirements for creating a simulation environment, with the goal to produce realistic sensor data, has been discussed and compared with currently available simulations. It was found, that in favor of performance, correct influences and error models are often ignored.

In the second part, a method to model environmental influences and sensors has been presented which allows a comfortable way to model realistic sensors. The separation between environmental conditions, sensors and hardware specific errors, helps to incrementally increase the simulation detail until the desired degree of realism is reached.

Since the computational model is reviewed every frame the model can adapt to the currently simulated situation. The usage of shader programs during simulation runtime allows an efficient execution of the computational model which can be interpreted as a system dynamics simulation for each point on objects surfaces.

Acknowledgment

This work has been supported by the Federal Ministry of Economics and Technology (BMWi) under the grant 01MA09037. This text reects the views only of the authors.

References

[1] B. Gerkey, R. Vaughan, and K. Stoy, “Most valuable player: A robot device server for distributed control,” *International Conference on Intelligent Robots and Systems*, no. Iros, pp. 1226–1231, 2001.

[2] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *ICRA Workshop on Open Source Software*, 2009.

[3] (2012) The Mobile Robotik Programming Toolkit . Retrieved: July. 2012. [Online]. Available: http://http://www.mrpt.org/

[4] S. Van Hoecke, S. Verstockt, K. Samyn, M. Slembrouck, and R. de Walle, “Tunnel simulator for traffic video detection,” in *SIMUL 2011*. International Academy, Research, and Industry Association (IARIA), 2011, pp. 57–62.

[5] J. Rossmann, N. Hempe, and M. Emde, “New methods of render-supported sensor simulation in modern real-time VR-simulation systems,” in *Proceedings of the 15th WSEAS international conference on Computers*. Stevens Point, Wisconsin, USA: WSEAS, 2011, pp. 358–364.

[6] S. Balakirsky and C. Scrapper, “USARSim : Providing a Framework for Multi-robot Performance Evaluation,” *Simulation*, pp. 98–102.

[7] C. Goodin, P. J. Durst, B. Gates, C. Cummins, and J. Priddy, “High Fidelity Sensor Simulations for the Virtual Autonomous Navigation Environment,” pp. 75–86, 2010.

[8] M. Siegel, “Sensor modeling and simulation: can it pass the Turing test?” *VIMS 2001. 2001 IEEE International Workshop on Virtual and Intelligent Measurement Systems (IEEE Cat. No.01EX447)*, pp. 92–96, 2001.

[9] C. Rosen, U. Jeppsson, L. Rieger, and P. a. Vanrolleghem, “Adding realism to simulated sensors and actuators.” *Water science and technology*, vol. 57, no. 3, pp. 337–44, Jan. 2008.

[10] C. Goodin, R. Kala, A. Carrrillo, and L. Y. Liu, “Sensor modeling for the Virtual Autonomous Navigation Environment,” *2009 IEEE Sensors*, pp. 1588–1592, Oct. 2009.

[11] K. Khoshelham and S. O. Elberink, “Accuracy and resolution of Kinect depth data for indoor mapping applications.” *Sensors (Basel, Switzerland)*, vol. 12, no. 2, pp. 1437–54, Jan. 2012.

[12] (2012) OGRE. Retrieved: Sept. 2012. [Online]. Available: http://www.ogre3d.org/

[13] F. Ford, *Modeling the environment: an introduction to system dynamics models of environmental systems*. Island Pr, 1999.

Mesoscopic Level: A New Representation Level for Large Scale Agent-Based Simulations

Laurent Navarro
Thales / UPMC – LIP6
Paris, France
laurent.navarro@thalesgroup.com

Vincent Corruble
UPMC – LIP6
Paris, France
vincent.corruble@lip6.fr

Fabien Flacher
Thales
Vélizy, France
fabien.flacher@thalesgroup.com

Jean-Daniel Zucker
UPMC – IRD
Paris, France
jean-daniel.zucker@ird.fr

Abstract — **Large-scale simulations often use multiple models to find a balance between run-time performance and number of simulated entities. Although these approaches are effective, they do not always offer the needed level of analysis, especially when this one is between the resolutions of the models used. In this paper, we aim at offering a finer resolution in exploring this trade-off by introducing an intermediate level between two given resolutions, which can apply to all agent models and allows a more progressive transition to offer the desired analysis level. We introduce a framework for such a methodology and evaluate it through the extension of an existing approach along two criteria: its impact on the computational resources, and an estimate of the dissimilarity between a simulation with our methodology and one without. Initial experiments show that the consistency is almost maintained while the CPU gain varies from low to significant depending on the context.**

Keywords – Agent-based and Multiagent Systems; Model-Based Systems; Large-scale simulations; Level of Detail.

I. INTRODUCTION

To understand the complex world surrounding them, humans often resort to models. Models are abstractions of the reality, extracting and simplifying some aspect of the real world to understand its process and predict it. All models are built with a given resolution – or Level Of Detail (LOD) – which depends on the representation level of the entities composing the system and of their behaviors, on the input and output domains and on the scope of the system [1]. The LOD can vary from high-resolution – microscopic – levels, where each component of the system is fully detailed, to lower-resolution – macroscopic – levels that simplify or omit parts of the system to focus only on a behavior of interest. For example, a microscopic description of a gas may be given by the description of each of its particle whereas its macroscopic model may be obtained using the so-called ideal-gas equation law.

Each of those LOD levels addresses different issues and has advantages and drawbacks. High-resolution models allow for a precise grasp of a specific phenomenon and tend to simulate the world more realistically, but they can be very difficult to design due to their high number of parameters. Moreover, their accuracy is often achieved at the cost of huge computational cost. A possible microscopic description of a gas is the position and velocity of each particle as well as the model of their interactions. Considering there are roughly 2.7×10^{19} molecules / cm^3 at room temperature and pressure, it is actually impossible to simulate such a system within a reasonable time. On the other hand, low-resolution models allow a better overall understanding, by focusing on the forest rather than the trees, and are therefore more suitable for decision support. They can be calculated faster but are less accurate and may not fit all situations. Indeed, the ideal gas law requires a few parameters and is far more accessible and usable for engineering applications, although it is restricted to equilibrium states.

Several approaches exist between the molecular model and the ideal gas law, such as Kinetic Molecular Theory, Brownian Motion or high level Boltzmann and Navier–Stokes equations [3]. Each of them is designed for a specific purpose and only works in a particular context – such as low pressure or homogeneity of the molecules – outside of which it is usually inoperative. Thus, there may be questions whose answers lie at the intersection of multiple resolutions, part in the microscopic and part in the macroscopic. In our example, taking care of the particles' motion in a gas while monitoring the whole temperature of a room is impossible because this answer needs both models to be run simultaneously – which cannot be done for the microscopic one. To be obtained, this answer requires a new model.

Those points are particularly important in computer agent-based simulation where the right level of resolution must be found to get the best compromise between the genericity of the system, its intelligibility and its need for CPU and memory resources. In this context, we present here a novel approach for multi-level agent-based simulations, by introducing an intermediate level between microscopic and macroscopic resolutions, which can apply to all agent models. This level, referred here as *mesoscopic*, allows a more progressive transition between two models to offer the desired analysis level given the context of the simulation and the user needs. To do so, we first extend the generic notions of dynamic change of representation and spatial aggregation introduced in previous work on the dynamic LOD for agent models [6]. Then, we position ourselves in the context of multi-agent simulations and define several environments in which we evaluate the approach experimentally. Finally we discuss the results obtained and propose enhancements for future work.

In this study, we define several criteria to evaluate the models, such as their scalability – their capacity to simulate a high number of entities – and their precision, thus their ability to give accurate results. We also focus on the design cost needed to implement them as well as the understandability of their results, which is the ease with which the users can understand what is happening.

II. RELATED WORK

Finding the resolution that best suits a given problem, among several models of a given phenomenon, has been widely studied within the Multi Representation Modeling (MRM) field through the joint execution of multiple models. In selective viewing, only the highest resolution model is executed at all times, and all other models are emulated by abstracting the representation of the most detailed one [7]. This approach is used when the simulation requires a phenomenon to be modeled in detail. Although it may be efficient for applications, which need high precision, it requires huge computational resources. Moreover, multiple models do not necessarily have hierarchical relationships between them, preventing designers to define which model is the most detailed. Finally, executing the most detailed model does not facilitate decision making.

In aggregation / disaggregation techniques, one model is executed at a given time; but, here it is not necessarily the most detailed one. Aggregation corresponds to the transition from high-resolution entities to a low-resolution one, while disaggregation is the opposite process. The choice of model depends on the user's need or the necessity to match the resolution of other interacting entities. Several variants exist, such as full disaggregation [10], partial disaggregation [5], playboxes [9] and pseudo-disaggregation [8]. Each of them can lead to speedup when a balance between complexity and simulation needs is found. But they require huge resources when moving from one model to another, and problems – such as chain disaggregation – may arise in case of cross-levels interactions. Finally, transition latency, network flooding and thrashing may impact simulation consistency.

Variable Resolution Modeling (VRM) allows the creation of families of models that support dynamic changes in resolution [7] by introducing several constraints. Thus, all the models parameters are standardized within a dictionary and inserted in a hierarchical structure symbolizing their dependencies. Rules are defined between models to match the computation time steps, ensure the consistency of the simulation and allow the calibration. Following those rules, a designer can create a family of models that can adapt their resolution level to the simulation needs. But this approach is mainly theoretical and is not suitable when the models are pre-designed and cannot be adapted to the VRM approach.

Multiple Representation Entities [4] is a last example from the MRM field that is of particular interest here. This approach maintains, at all time in the simulation, all representations through all available models of a given phenomenon, using appropriate mapping functions to translate changes between two representations. This allows interactions between all the representations, and avoids loss of resources when scaling from one model to another. MRE is a powerful way to deal with complex MRM, which offers a remedy for the weakness of aggregation / disaggregation methods and requires lower resources than simultaneous execution of multiple models. But, it only gives mathematical requirements for mapping functions, through the use of attributes dependency graphs. Also, it does not identify the representation at any level nor relationships between representations.

Some approaches in Multi-Agent Simulation (MAS) field also exploit the principle of simultaneous use of microscopic and macroscopic models, by partitioning the environment and running a model in each zone. The pedestrian's simulation described in [13] uses high-level flow and distribution models to steer non-visible agents along a network of nodes that describe the accessible areas of a city, and a microscopic collision avoidance model with speed adjustment for visible actors. Similarly, Bourrel and Henn [11] and El Hmam et al. [12] describe traffic simulations using a static predesigned world. Thus, a macroscopic model based on the flow theory is used in low interest areas without crossroads, and a microscopic multi-agent car-following model in high interest areas. Those architectures can handle several thousand agents with high consistency level and offer a good interactivity with the agents' behavior within both macroscopic and microscopic areas. But, they require a preprocessed environment and predefined transition functions between the agent models.

III. MESOSCOPIC LEVEL

A. *The mesoscopic representation*

Our approach start from the foundations defined in [6], where a method to go from several microscopic agents to a single macroscopic aggregate is detailed. Our goal here is to create an intermediate level between the microscopic and the macroscopic ones. Unlike the macroscopic level in which all agents are aggregated into a single one, this mesoscopic level centralizes parts of the computation performed on the microscopic agents in order to free computational resources while letting other parts be updated according to their initial level.

We define an agent model M as a computational abstraction of the global behavior of a synthetic actor. Thus, it takes as input the representation of the agent being driven and a representation of its environment, and outputs an action or a modification of the agent's representation. This representation – denoted by $Rep(M)$ – is the set of attributes needed by the agent model to perform its task and is usually assimilated to the internal state of the agent. Similarly, the representation of an agent A_1 in M at time t is the vector of attributes' values, denoted $Rep(A_1; M; t)$ such as:

$$Rep(M) = \{a_i\}_{i\in[1..|M|]} \text{ and } Rep(A_1; M; t) = \begin{pmatrix} a_{1;1}(t) \\ \vdots \\ a_{1;|M|}(t) \end{pmatrix}$$

An agent model can usually be split into several distinct processes, each being a mostly autonomous module leading to a particular skill of the agent such as navigation, decision, emotions, planning, communication or social interactions. For example, in well-known cognitive architectures such as SOAR and ACT-R [2], the processes could be the emotional, decisional or sensitive / short term / long term memories modules for the first one, or the declarative procedural memory, pattern matching, and production execution modules for the second. Those processes are themselves models, taking as inputs a subset of the agent model's inputs and outputting modifications of the agent's representation as well as specific data. They are usually chained, each of them requiring others to do their work before it can execute its own. For example, in SOAR, the decision process needs the elaboration process to fire all production rules, meaning that the working memory has been previously updated by the perception module. Thus, it is often possible to identify a hierarchy of dependencies between processes within a single agent model.

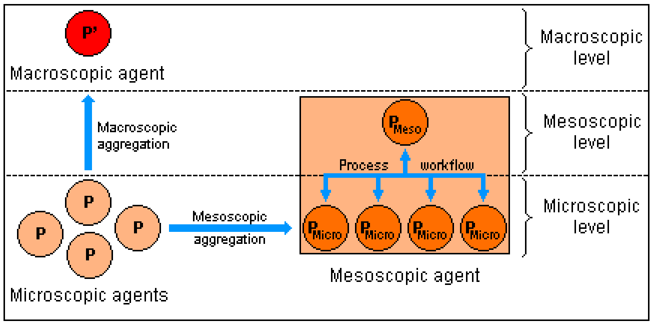


Figure 1. Example of the macroscopic and mesoscopic aggregation of 4 agents implemented as a set of P processes.

We consider that the agent model M is composed of a set of m processes $P = \{P_1;\ P_2;\ \dots;\ P_m\}$. The goal, schematized in Figure 1, is to allow the subset $P_{meso} \subset P$ to be run at the mesoscopic level while $P_{micro} = P \setminus P_{meso}$ will remain at the microscopic level. The microscopic representation of an agent A_1 and the one of a set of n agents $A = \{A_1;\ A_2;\ \dots;\ A_n\}$ in $P_k \in P_{meso}$ at time t are respectively denoted by $Rep_{micro}(A_1; P_k; t)$ and $Rep_{micro}(A; P_k; t)$ such as:

$$Rep_{micro}(A_1;\ P_k; t) = \begin{pmatrix} a^k_{1;1}(t) \\ \vdots \\ a^k_{1;|P_k|}(t) \end{pmatrix} \ with\ Rep(P_k) \subset Rep(M)$$

$$Rep_{micro}(A;\ P_k; t) = \left(Rep_{micro}(A_1;\ P_k; t); \dots ; Rep_{micro}(A_n;\ P_k; t)\right)$$
$$= \begin{pmatrix} a^k_{1;1}(t) & \cdots & a^k_{n;1}(t) \\ \vdots & \ddots & \vdots \\ a^k_{1;|P_k|}(t) & \cdots & a^k_{n;|P_k|}(t) \end{pmatrix}$$

For a process $P_k \in P_{meso}$, we need to compute the mesoscopic representation of A at time t denoted by $Rep_{meso}(A; P_k; t)$. To do so, we use the methodology described in [6]. Firstly, we partition the representation of the agents in M among several attributes classes in which each attribute share the same meaning and therefore a common dynamics. Then, we link each class with an aggregation operator and its corresponding disaggregation and memory operators. Then, we are able to define aggregation, disaggregation and memory functions, respectively denoted by F_{Ag}, F_{Disag} and F_{Mem} and for the process P_k such as:

$$F_{Ag}[Rep_{micro}(A;\ P_k; t)] = Rep_{meso}(A; P_k; t)$$

$$F_{Mem}[Rep_{meso}(A; P_k; t); Rep_{micro}(A;\ P_k; t)] = Mem(A;\ P_k; t)$$
$$= \begin{pmatrix} m^k_{1;1}(t) & \cdots & m^k_{n;1}(t) \\ \vdots & \ddots & \vdots \\ m^k_{1;|P_k|}(t) & \cdots & m^k_{n;|P_k|}(t) \end{pmatrix}$$

$$F_{Disag}[Rep_{meso}(A; P_k; t');\ Mem(A;\ P_k; t)] = Rep_{micro}(A;\ P_k; t')$$

This method allows a single process to work at the mesoscopic level. However, this process is part of a hierarchy and may have dependencies with other processes. In order to avoid inconsistencies in the computation of the agent model, we must consider the attributes of $Rep(M)$. If an attribute a is only used at the microscopic level, then it is ignored. On the other hand, if a is only used at the mesoscopic level, meaning that $a \notin \bigcup_{P_k \in P_{micro}} Rep(P_k)$, it is aggregated once. Finally, if a is used at both level, we need to maintain both microscopic and mesoscopic values of a when it is updated by any process, with the aggregation, disaggregation and memory functions described above for the attributes class to witch a belong. In practice, it is possible to restrict such computation – which can be CPU intensive – by updating the microscopic values of an attribute only if a mesoscopic process has updated it earlier in the agent model update and vice versa.

Such approach permits the migration of any process constituting the agent model from the microscopic to the mesoscopic level, resulting in the freeing of computation time. The choice of the aggregation functions – and their corresponding disaggregation and memory operators – must be done wisely in order to maintain simulation consistency as defined in [7]. The choice of the process to migrate is also an important issue. It is better to migrate processes that require high computation resources, and it is easier to migrate those that have few dependencies with others. However, the choice of the processes to migrate has also an impact on the simulation consistency. Finally, it is important to notice that migrating all processes from the microscopic to the mesoscopic level is equivalent to aggregating the set of agents to a macroscopic one driven by the same agent model as defined in [6].

B. *Spatial aggregation*

This section tackles the problem of finding the agents that should be aggregated to form a mesoscopic agent, and which processes of this new agent must be migrated to the mesoscopic level. The philosophy employed here is similar to the one detailed in [6] with the definition of a spatial distance D_Θ, and a psychological distance D_Ψ combined to estimate an affinity between two agents A_1 and A_2, denoted by $Aff(A_1; A_2)$, and the affinity between two agents A_1 and A_2 and a set of M events $E = \{E_1; E_2; \dots; E_M\}$, denoted by $Aff(A_1; A_2; E)$, such as:

$$Aff(A_1; A_2) = f[D_\Theta(A_1; A_2); D_\Psi(A_1; A_2)]$$

$$Aff(A_1; A_2; E) = \underset{i \in [\![1;|E|]\!]}{\mathrm{Max}} \left[f[D_\Theta(A_1;\ A_2;\ E_i); D_\Psi(A_1;\ A_2;\ E_i)]\right]$$

$$with \begin{cases} D_\Theta(A_1; A_2;\ E_i) = \mathrm{Min}[D_\Theta(A_1; E_i);\ D_\Theta(A_2; E_i)] \\ D_\Psi(A_1; A_2; E_i) = \mathrm{Min}[D_\Psi(A_1; E_i);\ D_\Psi(A_2; E_i)] \end{cases}$$

ISBN: 978-1-61208-234-9

However, the definition of D_Θ and D_Ψ is not trivial because parts of the physical or psychological processes may be at the microscopic or the mesoscopic level. In [6], the affinities are combined to define the aggregation utility U_{Ag} between two agents, which represents the usefulness of creating a mesoscopic entity considering those agents and the simulation context. This functionality is enhanced here by the ability to choose which process should migrate to the mesoscopic level while the others stay at the microscopic one. To do so, we define for each process an aggregation threshold, thus creating a total order over them, meaning that the lower the threshold, the more likely the process is to be migrated to the mesoscopic level. Those thresholds do not depend on the dependency hierarchy described above. However, choosing the processes order according to the hierarchy lowers the risk of having attributes at microscopic and mesoscopic levels which, as seen before, need to be maintained in both representations to ensure the consistency of the processes computation. Of course, it may happen that two processes cannot be separated because of some characteristics of their implementation or of the high number of attributes they share. In this case, a possible solution would be to assign the same threshold to both.

The disaggregation of a mesoscopic agent proceeds of the same idea, via the definition of a disaggregation utility. However, unlike the macroscopic approach where this utility has to be computed once for the whole aggregate, it must here be computed for each microscopic entity composing the mesoscopic agent, because some of its processes might remain at the microscopic level and are involved in the calculus of D_Θ and D_Ψ. Although this approach requires significant computational resources, it allows disaggregating a single microscopic agent due to the simulation context, which was not possible with the macroscopic aggregate. For example, if a microscopic agent tries to communicate with some microscopic entities of a mesoscopic agent, and if the communication process is still at the microscopic level, then the disaggregation utility of the communicating entities – and only them – will allow a partial disaggregation of the mesoscopic agent. We then have, for a mesoscopic agent A:

$$Aff(A_i;E) = \underset{i\in[\![1;|E|]\!]}{\mathrm{Max}}[Aff(A_i;E_i)], A_i \in A$$

The method described above only applies when the aggregation and disaggregation utilities between two agents must be computed. However, it does not allow the processes to migrate dynamically when the mesoscopic agent is alone. However, it is possible to define a representation change utility for a mesoscopic (or macroscopic) agent A, denoted by $U_{RC}(A)$, such as:

$$U_{RC}(A) = g\left[\underset{i\in[\![1;|E|]\!]}{\mathrm{Max}}\left[f[D_\Theta(A;\ E_i); D_\Psi(A;\ E_i)]\right]\right]$$

The representation change utility has nearly the same meaning as the aggregation utility except that it applies to a single agent. As a result, comparing it to the processes aggregation thresholds lets the aggregate adapt dynamically the computation level of its own processes. While this approach allows a complete control over the processes migration, it may require additional CPU resources if it is applied at each LOD update for every single agent.

IV. Experimental evaluation

The approach described above has been implemented and evaluated within a Thales proprietary multi-agent simulator named SE-*. This system is a synthetic environment engine in which each agent has a motivational tree containing predefined attributes, internal variables, emotion and motivations, and can exhibit complex adaptive behaviors. The agent model contains several processes on which our approach can work, such as perceptions, emotions, decision, planning, navigation and interaction with the environment through Smart Objects. Currently, SE-* can animate up to 20,000 agents driven by more than 20 motivations within a complex environment.

For these experimentations, and to keep a common context with [6], we split the representation of the agent model between two main attributes classes: physical and psychological. We do the same for the processes, thus linking the mental processes – emotions, decision and planning – to a unique aggregation threshold while the other physical processes – perception and navigation – are assigned an infinite threshold. The goal is to allow only the processes working on psychological attributes to migrate at the mesoscopic level. Doing so, the microscopic agents will share their thoughts through the mesoscopic one while their bodies will remain on the simulation. This LOD approach tries to reflect the human ability to be more sensitive to the physical or visual inconsistencies – wrong trajectories, oscillations, bad avoidance – than the psychological ones.

By keeping the physical parts of the microscopic agents, we hope to solve the spatial inconsistencies observed during the experimentations in [6]. Thus, letting the perception process at the microscopic level means that the perception of the mesoscopic agent are an automatic aggregation of those of its microscopic entities. Moreover, the choice of leaving the process manage the interactions with the environment at the microscopic level implies that all parts of the mesoscopic agent interacting with a Smart Object will be disaggregated, following the definition of the disaggregation utility defined above. Such a choice leads to an additional cost in computing resources, but is the easiest way to handle interactions here. Indeed, migrating this process to the mesoscopic level would require specific interaction models in the objects themselves, giving them the ability to interact with only a part of a mesoscopic agent. This point is the most important functional difference between macroscopic and mesoscopic simulations.

We use two scenarios that were already defined in [6]. The first one takes place in a subway station initially empty, including various objects such as ATMs, ticket vending machines, beverage dispensers and ticket barriers. The second one occurs in a large city that incorporates the subway station described above. In each scenario, the agents are driven by a dozen different motivations, such as going to work, drinking, wondering, destroying or repairing a machine, or fleeing. Each incoming agent in the simulation has random internal traits and inventory. We run each scenario with different values for the maximum number of actors allowed in the environment and the maximum size allowed for an aggregate.

As in [6], each scenario is run twice – one as a fully microscopic simulation without any LOD process and one with our dynamic aggregation method activated – during 30 minutes on an Intel Core i5 2.50 GHz laptop with a memory of 4 GB. Three criteria are computed: the *actual size* of the aggregates, the *CPU gain* and *the consistency*. The actual size tends to estimate the actual impact of the approach on the simulation and to link it with the two other criteria. The CPU gain is computed by comparing the time needed by both simulations to compute 60 frames. Finally, the consistency is calculated by comparing the cumulative number of uses of each object as a function of time between both simulations. With $U_o(t)$ the cumulative number of uses of object o at time t during the microscopic simulation, and $U_o'(t)$ the cumulative number of uses of the same object at the same time during the LOD simulation, then:

$$Consistency = 100\left(1 - \frac{1}{N_{Objects}}\sum_{o=1}^{N_{Objects}}\left\{\frac{\sum_{t=0}^{T}[U_o(t) - U_o'(t)]}{\sum_{t=0}^{T} U_o(t)}\right\}\right)$$

The results of the experimentations done on the subway station are shown in TABLE 1. It appears that the mesoscopic level allows a slight gain in CPU while the consistency reaches a very high level. Moreover, the real group size is relatively low, regardless of the configured maximum size. As the maximum number of entities in the station increases, the CPU gain decreases and the consistency remains stable. Finally, unlike the simulations with macroscopic aggregates, the strong dissimilarity observed when the maximum number of agents exceeds 500 no longer exists.

TABLE 1. EXPERIMENTATION RESULTS ON THE SUBWAY STATIONS.

Entities	Group Size	Actual Group Size		CPU Gain (%)		Consistency (%)	
		Macro	Meso	Macro	Meso	Macro	Meso
100	5	2,8	2,1	53,1	10,8	98,0	98,8
	10	3,6	2,3	58,3	11,9	97,6	97,2
	20	4,3	2,4	61,2	13,2	92,3	97,6
300	5	3,6	1,7	69,9	7,6	92,6	98,9
	10	4,7	1,9	74,5	9,5	90,7	98,7
	20	5,4	1,9	77,5	10,1	87,4	99,1
500	5	3,5	1,5	71,1	4,8	78,0	98,7
	10	4,0	1,6	74,0	7,0	80,1	98,9
	20	4,6	1,7	76,3	7,6	81,8	98,7
1000	5	3,6	1,1	71,4	2,0	76,3	99,3
	10	4,3	1,3	74,6	3,6	77,8	99,5
	20	4,6	1,3	73,9	3,7	78,1	99,0

This evolution of the criteria can be explained by the preservation of the interaction process at the microscopic level. Indeed, all agents entering the station have at least one interaction with the ticket barriers – and most of them have 2 or 3 more interactions – before reaching a train or an exit. Thus, a lot of disaggregation occurs and the microscopic agents queuing at the machine lower the mean group size as well as the CPU gain. Moreover, only the mental processes were set to migrate to the mesoscopic level, leaving some heavy processes with quadratic complexity, like navigation or perception, at the microscopic level. This explains why the CPU gain is not linear in the actual group size.

The impact of the interactions can be observed in the second experimentation. Here, only a few agents among the 10000 want to take the train in the subway station, the others just walk randomly in the city. Thus, the number of interactions with objects is smaller than in the first scenario. TABLE 2 shows that the actual group size is nearly the same for the macroscopic and the mesoscopic scenario, meaning that the limiting parameter is only the aggregation threshold applied to the aggregation utility. As a consequence, the mesoscopic CPU gain is far higher in this scenario. This result is encouraging as it implies that the approach can save more computational resources in large spaces where agents limit their interactions with the environment.

TABLE 2. EXPERIMENTATION RESULTS FOR THE CITY ENVIRONMENT.

Entities	Group Size	Actual Group Size		CPU Gain (%)	
		Macro	Meso	Macro	Meso
10000	5	5.0	4,9	69,0	31,3
	10	9,2	9,4	73,4	36,4
	20	13,7	13,4	81,7	42,2

The comparison between the approach in [6] and the one described here shows that in terms of CPU gain and consistency, the mesoscopic level is an intermediate between the microscopic and macroscopic resolutions. This point is of particular interest here as the mesoscopic level is – by construction – an intermediate toward the construction of the macroscopic one. Thus, if we link the non-mental processes to a second aggregation threshold which value is higher than the one defined for the mental processes, then this second aggregation would lead to the creation of a unique macroscopic aggregate as detailed in [6]. The mesoscopic state is then an intermediary step to another resolution level, possibly driven by a different agent model.

Moreover, the stability of the consistency of the mesoscopic level for the simulations involving more than 500 actors, where the macroscopic level shows an important dissimilarity, means that our approach can model the congestions in the station and the evacuation of the agents which are under psychological stress. Indeed, when the subway station is crowded, we see agents that cannot access the machines getting nervous and leaving the station. This phenomenon, which does not exist in the fully macroscopic simulation, remains in the mesoscopic experimentations.

Moreover, we observe that the stress of the mesoscopic agent is increasing due to its perceptions leading it to leave the station. This shows that the interaction between both resolution levels in the mesoscopic agent leads to consistent actions and can reproduce microscopic behaviors observed in real settings.

V. Discussion and future work

In this paper, we presented a novel approach for multi-levels simulations, by introducing an intermediate level between microscopic and macroscopic resolutions, which allows a smoother resolution change between models. Indeed, it supports the definition of several aggregation steps, each corresponding to a process composing the initial agent model, and the migration of the agents to the appropriate aggregation step in regard to the context of the simulation. It corresponds to a kind of continuum between the lowest and highest level of simulation.

The results detailed in Section IV show a very high and steady consistency between the fully microscopic and the LOD simulations. On the other hand, the computational gain is not significant in constrained environments where the agents must often interact, but significant when those interactions are less intensive. So, our approach is able to reduce computational needs with no consistency loss as long as the processes maintained at the microscopic level do not need recurrent interactions with their counterpart in other mesoscopic agents, creating partial disaggregation.

This point highlights the importance given to model design in this approach. Indeed, to apply our method, one needs to have a precise view of the available processes, as well as the complete representation of the agent model. While this is always theoretically possible, in practice this may require some modifications of a simulator to control the update of each process and catch the transiting data between them. Moreover, the choice of the aggregation threshold is fundamental as it has a direct impact on the resources, because the processes do not have the same complexity – thus not the same interest to migrate to the mesoscopic level – and because having attributes involved in both microscopic and mesoscopic representations requires the use of the aggregation and disaggregation functions associated with their attribute class. It would be interesting to study the rules that define the optimal aggregation thresholds, depending on the complexity of the processes and their dependency hierarchy. Machine learning approaches could also help find the best values for the aggregation thresholds.

Finally, it would be particularly interesting to enhance the experimental part of our work. Thus, by setting different aggregation thresholds we could test mesoscopic agents having shared perceptions but separate decision and navigation processes, or having only a common long term memory, to test the impact on the consistency. Moreover, we could create enhanced scenarios. Firstly, we could use a train station with a larger scale than our subway station, allowing the agents to have complex behaviors without having too many interactions. We could see if the CPU gain tends to reach the one observed on the city while the consistency remains maximal. Secondly, and more important, we could add more mesoscopic aggregation steps and combine mesoscopic and macroscopic approaches into a unique scenario to verify that the smooth aggregation has an impact on consistency.

References

[1] J. D. Zucker. *A grounded theory of abstraction in artificial intelligence*. Philosophical Transactions of the Royal Society (volume 358, number 1435). 2003.

[2] P. Langley, J. Laird and S. Rogers. *Cognitive Architectures: Research Issues and Challenges*. Cognitive Systems Research. 2009.

[3] J. Uffink. *Compendium of the foundations of classical statistical physics*. Philosophy of Physics (volume 2, number 1). 2006.

[4] P. Reynolds and A. Natrajan. *Consistency Maintenance in Multiresolution Simulations*. ACM Transactions on Modeling and Computer Simulation. 1997.

[5] D. Hardy and M. Healy. *Constructive & Virtual Interoperation: A Technical Challenge*. In proceedings of the 4th Conference on Computer Generated Forces and Behavioral Representation (CGF-BR 1994). 1994.

[6] L. Navarro, F. Flacher and V. Corruble. *Dynamic Level of Detail for Large Scale Agent-Based Urban Simulations*. In proceedings of the 10th International Confer-ence on Autonomous Agents and Multiagent Systems (AAMAS'11). 2011.

[7] P. Davis and R. Hillestad. *Families of Models that Cross Levels of Resolution: Issues for Design, Calibration and Management*. In proceedings of the 25th Winter Simulation Conference (WSC'93). 1993.

[8] R. Calder, J. Peacock, B. Wise, T. Stanzione, F. Chamberlain and J. Panagos. *Implementation of a Dynamic Aggregation / Disaggregation Process in the JPSD CLCGF*. In proceedings of the 5th Conference on Computer Generated Forces and Behavioral Representation (CGF-BR 1995). 1995.

[9] C. Karr and E. Root. *Integrating Aggregate and Vehicle Level Simulations*. In pro-ceedings of the 4th Conference on Computer Generated Forces and Behavioral Representation (CGF-BR 1994). 1994.

[10] Calder, J. Peacock, J. Panagos and T. Johnson. *Integration of Constructive, Virtual, Live, and Engineering Simulations in the JPSD CLCGF*. In proceedings of the 5th Conference on Computer Generated Forces and Behavioral Representation (CGF-BR 1995). 1995.

[11] E. Bourrel and V. Henn. *Mixing micro and macro representations of traffic flow: a first theoretical step*. In proceedings of the 9th Euro Working Group on Transportation Meeting (EWGT2002). 2002.

[12] M. El Hmam, D. Jolly, H. Abouaissa and A. Benasser. *Modélisation Hybride du Flux de Trafic*. Revue électronique Sciences et Technologies de l'Automatique. 2006.

[13] S. Stylianou, M. Fyrillas and Y. Chrysanthou. *Scalable Pedestrian Simulation for Virtual Cities*. In proceedings of the 11th ACM Symposium on Virtual Reality Software and Technology (ACM VRST 2004). 2004.

Object-Oriented Paradigms for Modelling Vascular Tumour Growth: A Case Study

Anthony J. Connor*, Jonathan Cooper*, Helen M. Byrne* †, Philip K. Maini‡ and Steve McKeever*
*Department of Computer Science
University of Oxford, Oxford, England.
†Oxford Centre for Collaborative Applied Mathematics
University of Oxford, Oxford, England.
‡Centre for Mathematical Biology
University of Oxford, Oxford, England.
Emails: anthony.connor@cs.ox.ac.uk, jonathan.cooper@cs.ox.ac.uk, helen.byrne@cs.ox.ac.uk,
maini@maths.ox.ac.uk, steve.mckeever@cs.ox.ac.uk.

***Abstract*—Motivated by a family of related hybrid multiscale models, we have built an object-oriented framework for developing and implementing multiscale models of vascular tumour growth. The models are implemented in our framework as a case study to highlight how object-oriented programming techniques and good object-oriented design may be used effectively to develop hybrid multiscale models of vascular tumour growth. The intention is that this paper will serve as a useful reference for researchers modelling complex biological systems and that these researchers will employ some of the techniques presented herein in their own projects.**

***Keywords*-multiscale; hybrid; object-orientation; vascular tumour growth; modelling.**

I. Introduction

Object-orientation was originally heralded as a “silver bullet” [1] for dealing with the software crisis, which was characterised by, amongst other things, the high complexity, low productivity, and poor reliability of software systems. Object-oriented programming (OOP) [2] promises to encourage code re-use, maintainability, understandability and extensibility for those systems which are amenable to object-oriented design. However, producing an object-oriented system or program with long term re-usability, extensibility and maintainability requires considerable investment of time and effort at the start of the project, for what at first may seem like little return; it is an art-form, often requiring years of experience and/or a time-consuming trial and error approach. Nevertheless, the potential benefits of a good object-oriented design are well worth the time and effort spent.

OOP today remains the most popular programming paradigm world wide, being used extensively by both software engineers and mathematical modellers. However, OOP is often used poorly or inappropriately, particularly by mathematical modellers, who may not have a background in computer science and little experience with OOP. This issue is compounded by the fact that modelling academics are under tremendous pressure to produce and publish results from implemented models quickly, before competing groups beat them to the punch. This inherently leads to little time and effort being put in to the design of model code, which might have ensured that the model implementation would be more understandable and could be more easily maintained and extended at a later date.

In this paper, we advocate the use of good object-oriented techniques for developing multiscale models of vascular tumour growth. Motivated by models proposed by Alarcón and co-workers [3], [4], [5], [6], [7], [8], we have developed an object-oriented modelling framework in which a range of hybrid multiscale models of vascular tumour growth are implementable. To date, the functionality of the framework focusses around the methodologies employed by Alarcon and co-workers. However, by employing an object-oriented framework, and designing it with re-usability and extensibility in mind, our framework could be extended to allow for additional functionality with relative ease.

Our paper focuses on the aspects of OOP which make it appropriate for developing models of complex biological systems, using the models of Alarcón and co-workers as a case-study. We exploit good object-oriented design principles, describing the models abstractly and decoupling the framework design from model implementations. These aspects of our work may have implications for increasing the trust-worthiness of model execution by remote users and improve the prospects for model re-use in the modelling community. By decomposing models into collaborating classes of biological entities and behavioural algorithms, we also enable the rigorous validation of models and a test-driven approach to model development to be implemented.

The remainder of the paper is structured as follows. In Sections II and III, we briefly introduce the application domain and some basic concepts of OOP. We also address what aspects of the paradigm make it appropriate for developing the complex multiscale biological models which we consider. In Section IV, we present the results of an exemplar simulation whose implementation was realised using the object-oriented framework we have developed. We then discuss the implications of our work and avenues of future work in Section V, before concluding in Section VI.

ISBN: 978-1-61208-234-9

II. The application domain

The entry of a tumour into its vascular growth phase marks the transition from a phase in which the tumour is essentially harmless to one in which it is potentially fatal [9]. During avascular growth, tumours are limited in size because they rely on diffusion to obtain nutrients. As a tumour grows, parts of it become deprived of oxygen, resulting in some of the tumour cells becoming quiescent and expressing various angiogenic factors. These tumour angiogenic factors (TAFs) diffuse throughout the surrounding tissue and, upon reaching a blood vessel, stimulate the formation of new vessel sprouts which migrate towards the tumour. After flow has been established in the new vasculature the surrounding tissue has increased access to nutrients, allowing the tumour to continue growing and to invade the adjacent healthy tissue. This process is known as tumour-induced angiogenesis. Once vascularised, the tumour also gains its own transport network by which tumour cells may be transported around the body to form metastases in any part of the host organism.

There is a large body of literature devoted to modelling both avascular tumour growth and tumour-induced angiogenesis. Broadly speaking such models can be placed into three categories: continuous, discrete and hybrid. For extensive reviews, see [10], [11].

Multiscale modelling involves the integration of several biological models, each describing a certain process at a particular time and length scale. Furthermore, each process may be represented using different mathematical modelling methodologies. For example, coupled ODEs may be used to describe subcellular processes and protein interaction networks and PDEs may be used to describe the diffusion of nutrients or chemical signals through tissues, while cell-cell interactions may be modelled discretely using on- or off-lattice techniques. Multiscale models allow modellers to capture the interdependence of biological phenomena which occur at different biological scales. They offer a natural framework for studying biological phenomena, such as angiogenesis and tumour growth, which are inherently multiscale in nature, and thus appear to offer the cutting-edge with regards to potential predictive power and clinical applicability. As such, multiscale models of angiogenesis and vascular tumour growth have gained in popularity over the last decade and have begun to show promise at the clinical level. It is for these reasons that we too focus on multiscale models of vascular tumour growth in this work.

Alarcon, Byrne, Maini and co-workers were one of the first groups to develop a vascular tumour model which incorporated multiple biological scales in a systematic way [3], [4], [5]. Their hybrid multiscale model has continued to develop through contributions from Betteridge [6], Owen [7] and Perfahl [8], amongst others. This family of models couples biological phenomena that include vascular adaptation and remodelling, blood flow, nutrient and vascular endothelial growth factor (VEGF) diffusion throughout the extracellular space and the cellular and subcellular dynamics of normal and cancerous cells. The mathematics that underpins the model implementations is described in the original papers and, for brevity, is not repeated here. Other notable multiscale models of tumour growth are presented in [12], [13]; for a further review, readers may consult [14].

One of the best hopes for developing increasingly complex models of vascular tumour growth that span multiple biological scales is by re-using and extending existing models. However, the integration or extension of existing models of vascular tumour growth (or elements of those models) currently represents a substantial technical challenge in the field. At present, mathematical models of cancer are often implemented by hand, in the language of choice of a modeller, with little or no thought given to how code may be re-used or models extended at a later date. This means that modellers must be familiar with the code in order to manipulate and evaluate simulation runs. Additionally, modellers are met with significant issues when they wish to re-use, extend, or maintain model code. As such, the development of cancer models and the success of the cancer modelling community as a whole is severely hindered. These are the principal factors which have inspired our work.

III. Object-oriented design for *in silico* models of vascular tumour growth

The way in which we think about and describe the world usually involves looking at it in terms of objects and the interactions between those objects. For complex biological systems, such as a growing tumour, verbal modelling does not enable us to describe or communicate aptly the complex non-linear feedback interactions which characterise those systems [15]. For this, we need mathematics. OOP strives to describe systems in terms of objects and the interactions between those objects computationally, thus enabling the quantitative mathematical analysis of complex biological systems. By describing systems in a way with which we are familiar, OOP promotes both an understanding of the system which we are trying to model and also an understanding of the model code itself; OOP helps us to manage the *essential* complexity [1] associated with real-world modelling.

Many of the object-oriented techniques described in this section help modellers to accurately describe the world in a way which is natural and understandable to others. This is particularly important within the field of biological modelling. Not only is it desirable for other modellers and programmers to understand our code, so that they may easily use, re-use and extend our models, but it is extremely desirable for experimentalists to be able to understand the system described through the code. This is because systems biology and the development of accurate and validated complex multiscale models heavily rely on the co-operation between experimentalists and modellers. Making computational models easier to understand for experimentalists, who may have little programming experience, will promote the closer collaboration of experimentalists and modellers.

A. *Objects, classes and modularity*

Objects in the context of OOP are complex data structures which contain data fields (attributes) and a complete set of operational methods to which that object may respond (its interface). In this way, objects represent a convenient data-centric way of decomposing systems into understandable and manageable sets of modules. In OOP, one actually usually designs *classes*, not objects. These classes, then, form the templates from which multiple objects may be instantiated and exist concurrently inside a program at run-time. Classes may be developed and tested individually before being integrated into larger systems being developed by a team or, in our case, potentially a larger community of programmers. Designing classes appropriately may make possible their re-use in other applications.

Simply by enforcing boundaries and structure on a program or model, objects as modules increase the maintainability of code. In software design, *low coupling* and *high cohesion* are often highlighted by programmers as qualities which a re-usable, extensible and maintainable piece of software should possess. Broadly, coupling refers to the interdependencies between different modules and cohesion refers to how strongly related the functions within a module are. High cohesion is an important quality because it increases the understandability of code. Low coupling, where one module interacts with another through a stable and well-defined interface and independently of the internal structure of the other module, increases maintainability, re-use and extensibility of code. Designing a library of classes with these properties is technically demanding and time-consuming but the benefits are invaluable, especially later in the life-cycle of a project.

B. *Encapsulation and information and implementation hiding*

The wrapping up of operations and attributes into a class, so that those attributes may only be manipulated through or accessed via the operations provided by the class, is encapsulation. Good encapsulation hides the details of an object's internal attributes and implementation of operations from modellers using that object [2]. These techniques are known as information hiding and implementation hiding, respectively, and their use is essential for promoting the understandability of code in our framework.

Many complex algorithms are employed in the models implementable in our framework, none more so than the structural adaptation algorithms used to determine the pseudo-steady-state radii of vessels and haematocrit distribution in a vessel network (see [3] for details). However, by packaging the operations and parameters required by this algorithm into a single `StructuralAdaptationAlgorithm` class and providing a single operation by which the entire algorithm may be implemented, we ensure that modellers may easily run the structural adaptation algorithm on a simulated vessel network without necessarily knowing the detailed mathematics involved in the algorithm. By encapsulating all parameters associated with the structural adaptation algorithm in this class, we also reduce the possibility of introducing errors into model implementation. This is established by ensuring that these parameters may only be accessed from a single point in the code and thus may not inadvertently affect other parts of the model implementation. In this way, encapsulation facilitates model code maintainability and changeability as well as understandability.

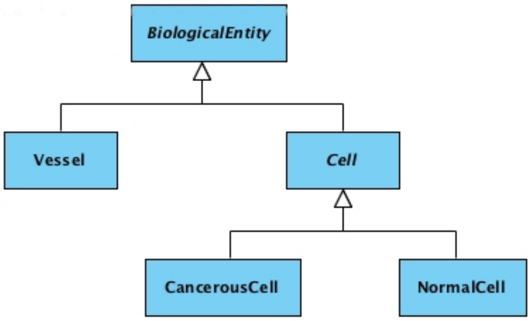


Figure 1: UML [16] diagram illustrating the class hierarchy for biological entities in our framework.

C. *Class and containment hierarchies*

In the real world, objects are often related by *is-a* type relationships. This type of relationship is also realisable in OOP languages by using class inheritance. For instance, in the models which motivate our work *vessels* and *cells* are the primary *biological entities* of interest. Additionally, we consider two types of cells: *cancerous* and *normal*. These relationships are represented in OOP using inheritance to define class hierarchies, such as that in Figure 1. The classes `CancerousCell` and `NormalCell` inherit from the class `Cell`, which in turn inherits from the `BiologicalEntity` class. The class of `Vessel` also inherits from the `BiologicalEntity` class. In this way, appropriate relationships, which mimic real-world relationships, are defined between some classes of objects in our modelling domain. Furthermore, class inheritance provides a powerful mechanism by which code may be re-used. By inheriting from an existing class, such as our `Cell` class, the subclasses (`CancerousCell` and `NormalCell`) automatically get all of the functionality (operations) and attributes defined in the `Cell` class. Moreover, new operations and attributes may be defined in the subclasses. In this way, inheritance allows programmers to define classes as incremental variations of more basic and abstract classes.

In OOP, inheritance also enables *substitutability*. In our example, this means that objects of type `CancerousCell`

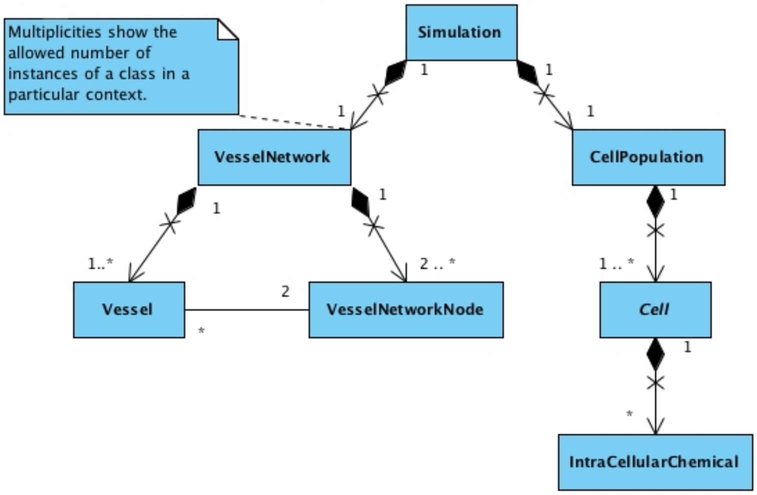


Figure 2: Containment hierarchy for major biological classes in our modelling framework.

or `NormalCell`, if defined appropriately, may be provided at run-time in any context where an object of type `Cell` is expected without affecting the correctness of the model implementation. This property of an object-oriented system imposes strict rules on class inheritance, namely that any class inheritance hierarchy must follow the principle of type conformance [2]. Similarly, the Liskov substitution principle [17] defines the appropriate use of subtyping relationships. The ability to substitute classes dynamically at run-time is hugely advantageous and we expand on this idea in Sections III-D and III-E.

OOP also allows objects to be contained within other objects. This is what is meant by a containment hierarchy. A simple example, implemented in our framework is shown in Figure 2. This hierarchy reflects the fact that vessel networks comprise many vessels which are connected at nodes in that vessel network and also that a cell population contains cells, each of which contain a set of intra-cellular chemicals, whose concentrations may dictate that cell's behaviour.

The construction of containment hierarchies is a useful technique, particularly for multiscale modelling. Objects contained within other objects may be used to represent nested levels of organisation, naturally mimicking the biologists' view of their systems. This technique also provides an intuitive mechanism for simplifying the implementation of individual submodels at the specific biological scale(s) with which that submodel is concerned. More complex containment hierarchies may be constructed to model biology at increased levels of detail, as demonstrated in [18].

In a wider context, certain classes of objects may be re-used by allowing instances of those classes to form components of more than one type of composite object. Class composition and inheritance may also be combined effectively in order to promote further re-use and extensibility of model code. This is best exemplified by the strategy pattern which is explained in Section III-E2.

D. *Polymorphism*

There are two distinctly different types of polymorphism in OOP: *static polymorphism* and *dynamic polymorphism*. Static polymorphism is also known as *overloading* and involves defining operations with the same name, but which take different types and/or numbers of input arguments. An appropriate example of overloading would be when requesting that a `CellPopulation` provide access to a `Cell` contained within that population. We may wish to access a `Cell` object by providing the `CellPopulation` object with either a unique id number, which each cell possesses, or with the location of the cell. In order to implement these requirements we define two operations inside the `CellPopulation` class, each called `GetCell`, which each return a pointer to the `Cell` object of interest:

1) `GetCell( int idNumber )` returns the `Cell` with `id` equal to `idNumber`.
2) `GetCell( coordinate cellLocation )` returns the `Cell` with `location` equal to `cellLocation`.

This technique aids in program design and model implementation by providing a uniform and intuitive interface by which object attributes may be accessed or manipulated [19].

Dynamic polymorphism, or *overriding*, is closely related to the concepts of inheritance and substitutability (Section III-C). We have already noted that inheritance guarantees that a subclass will contain at least the same operations that are defined in that class' superclass and sometimes more. OOP also allows the implementation of inherited operations to be redefined in a subclass: this is overriding. Calls to an operation which have been overridden in a subclass generally perform a slightly modified function. The functionality executed by a program then depends on what type of object the method is actually called on at run-time. This technique empowers modellers to create implementations which are very malleable and extensible by allowing interchangeable classes within a class hierarchy to perform moderately different functions in response to the same operation call. Such polymorphism plays an important role in the design patterns discussed in the next section.

E. Design patterns

Whilst design patterns are not strictly a part of OOP, the terms object-oriented design and design patterns became almost synonymous after the “Gang of Four” published their book, *Design Patterns: Elements of Reusable Object-Oriented Software* [20], in 1994. Design patterns represent elegant and re-usable solutions to problems commonly encountered in software design. Simply using a design pattern can improve code readability and understandability, especially for programmers and modellers who are familiar with the pattern used. Utilising these tried and tested solutions to software development problems may prevent subtle issues that may not have been considered but which could cause problems at a later stage in the life-cycle of a development process. From a biological modelling perspective, certain design patterns, for example the visitor pattern (Section III-E1), enable programmers to separate out biological structure from algorithms which determine the behaviour of the biological entities in a modelling domain. Such design patterns further promote model understandability for biologists.

We do not intend to explain or provide an exhaustive list of design patterns here, but instead present two behavioural design patterns that are used extensively throughout our framework: the visitor pattern and strategy pattern. We have also made use of the factory pattern, the template method pattern and the null object pattern in our code but do not present full descriptions of these additional patterns here.

1) Visitor pattern

The main appeal of using the visitor pattern is that it allows programmers to add new algorithms which operate over a class hierarchy without having to modify that class hierarchy, i.e., by adding operations specific to that algorithm to each member of that hierarchy. Implementation of this pattern involves defining two class hierarchies: one for the classes of objects being operated on (*elements*) and one for *visitors* which define the operations on elements. A dynamically polymorphic `accept` operation is defined in the element hierarchy which accepts a visitor object as an argument. A visited element responds to an `accept` operation call by calling the `visit` operation on the provided visitor, giving itself as an argument. Generic `visit` operations defined in the visitor hierarchy are overridden and overloaded in each member of the hierarchy, in order to define the behaviour of each algorithm on each element class, respectively. By grouping together related operations on one class hierarchy within a separate class hierarchy, we simplify both the class of objects carrying out a specific behaviour and the implementations of the various algorithms which determine that behaviour. In our framework, this pattern is used extensively to implement models for the behaviour of the vasculature, diffusible substances and cells. We will describe the visitor pattern as implemented in the context of modelling cell behaviour to better illustrate its use.

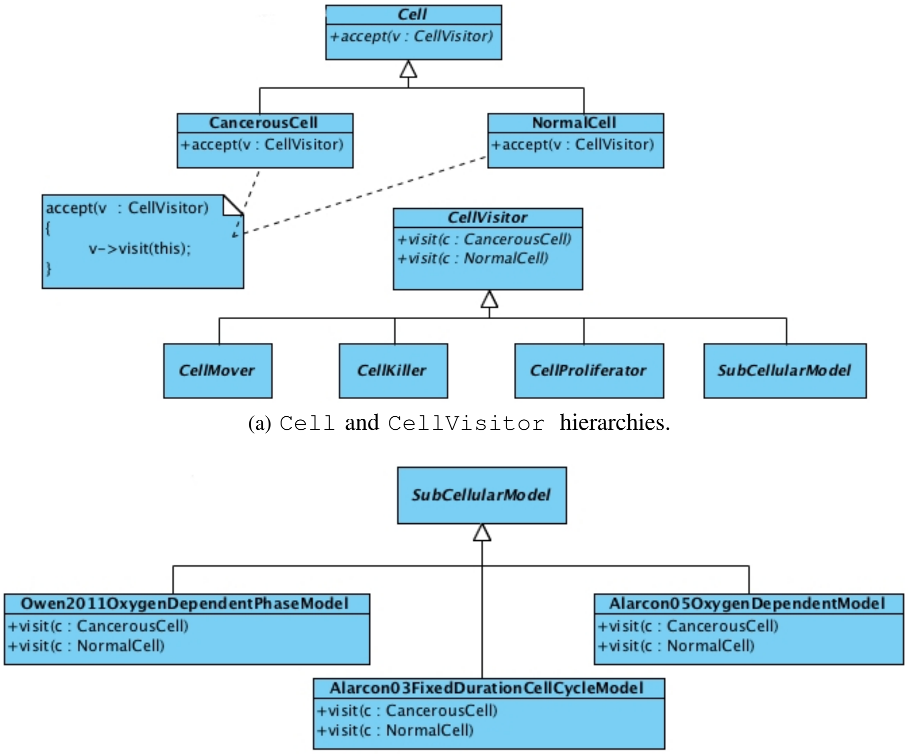
In our framework, populations of cells are modelled using an agent-based approach, where the behaviours of the cells are modelled by the application of rules, which may differ according to the type of cell to which the rule is being applied. We model three aspects of cell behaviour: movement, death, and proliferation. In addition we also model subcellular events. Models for subcellular processes include rules which dictate how the cell-cycle progresses; when cells should undergo apoptosis, when they should enter a quiescent state, when they should divide and how VEGF is produced and released. A number of subcellular models have been implemented in our framework, including a model in which division occurs after a fixed period of time [3], a simple oxygen dependent phase model [21], and a more complex subcellular model, first introduced in [4], involving the solution of seven coupled ODEs which represent how various intra-cellular chemicals evolve in time and account for the effect of local oxygen concentrations on cell-cycle progression. Similarly, different submodels dictate how cells should move and proliferate and also how and when a cell should die, e.g., in response to the presence of a drug [5].

Following the template outlined above, we define a `Cell` class hierarchy and a `CellVisitor` class hierarchy, whose members define the algorithms which determine various aspects of a `Cell` object’s behaviour (Figure 3 (a)). Dynamic polymorphism allows the different types of `CellVisitor` classes to modulate the type of cell behaviour which the visitor executes by responding to the same `visit` operation calls. Static polymorphism allows the same `CellVisitor` to implement slightly modified behaviours depending on what type of `Cell` object that visitor is operating on. In this way, an algorithm used to determine the behaviour of multiple cell types may be encapsulated inside a single class. Additionally, algorithm specific data structures and parameters are encapsulated within the visitor classes thus making the code more understandable and maintainable.

We extend the `CellVisitor` hierarchy in order to account for the different algorithms which may be used to model the same aspects of cell behaviour. For example, since we use several different types of subcellular model in our framework, we construct a hierarchy of `SubCellularModel` classes, each of which encapsulate a particular submodel, as shown in Figure 3 (b). At runtime, we may then choose which model we would like to implement very easily. We expand upon this idea in Section III-E2, where we present the strategy pattern.

It is particularly easy to vary and add functionality to our framework using this pattern. A new type of cell behaviour

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-234-9



(a) `Cell` and `CellVisitor` hierarchies.

(b) Hierarchy of classes that encapsulate the various subcellular models used in our framework.

Figure 3: Visitor pattern as implemented within the context of modelling discrete cell behaviour.

or a modified algorithm for, for instance, cell movement may be easily added to the framework by simply creating a new concrete class in the `CellVisitor` hierarchy. In this regard, this pattern makes extending certain aspects of the framework extremely easy and intuitive.

In general, the visitor pattern is applied only when the element class hierarchy is unlikely to change. Adding a new class to this hierarchy means that a new concrete implementation of the `visit` operation, which takes an instance of the new class as an argument, must be added to each concrete `CellVisitor` with an appropriate implementation. Thus, extending model code in this way can prove difficult. This issue may be partially negated by defining appropriate default implementations in the abstract `CellVisitor` class, however, this opens the door for potential bugs to creep in to model implementations. Nonetheless, this pattern is both adequate and appropriate for our purposes since we are currently interested primarily in changing the algorithms that determine the behaviour of biological entities rather than changing the types of biological entities that are present.

2) Strategy pattern

As well as modelling the behaviour of biological entities using the visitor pattern, we have also utilised the strategy pattern, which allows us to vary the precise algorithms which determine the various aspects of biological entity behaviour at run-time. Again, we clarify this statement with an example. At the highest level, for the running of simulations we utilize this pattern by defining a `Simulation` class. This class co-ordinates the events which occur during a model simulation, whilst delegating the responsibility of carrying out specific tasks to other classes. For example, a `Simulation` has a reference to the abstract `SubCellularModel` class. Different algorithms for this particular aspect of the system's behaviour are implemented as concrete subclasses, as shown in Figure 3(b). Due to substitutability, an instance of one such subclass may be assigned to this reference at run-time, providing the implementation as desired by a modeller for a particular realisation of a model. Figure 4 displays a class diagram which illustrates how a model simulation is constructed in our framework. Given that some simulations may not require a particular behaviour to be modelled, we complement the implementation of this design pattern by additionally implementing the *null object pattern* (an instance of this pattern involves defining an additional class within a class hierarchy which has an appropriate interface but possesses neutral behaviour). An example showing how we construct a simulation in our framework using this pattern is presented in the next section.

The strategy pattern promotes the understanding of model code by factoring out common functionality of algorithms into a suitable superclass, and promotes code maintainability and extensibility, allowing us to implement a large range of different models within our framework in an intuitive way.

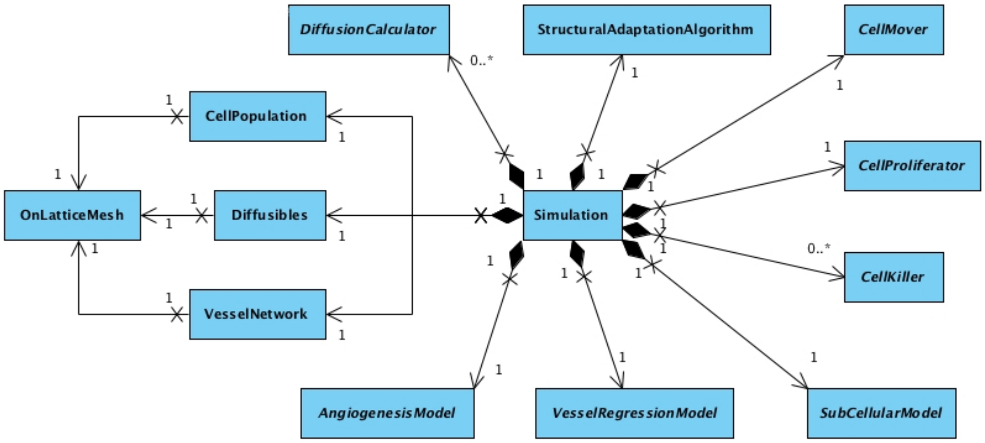


Figure 4: Class diagram showing the major classes involved in a simulation of vascular tumour growth in our modelling framework.

IV. Model Realisation

Due to its strongly object-oriented nature, a language supporting object-orientation is required to implement our framework. The nature of the models we consider also requires a language which supports pseudorandom number generation and multi-dimensional arrays. We chose C++ as our programming language because it fits these requirements, is relatively fast and is well-known and familiar to the scientific community. A wide range of open-source and reliable C++ libraries are also available which provide relevant trusted functionality and optimised algorithms. Several Boost [22] libraries are used extensively throughout our code, as is the Standard Template Library. We also employ the PETSc [23] library to solve large, sparse linear systems of equations. Simulations are visualised using Paraview [24].

Running a complex multiscale simulation of vascular tumour growth requires the co-ordinated interactions of many objects. One of the aims of our framework was to make the construction and running of model simulations intuitive and simple. An example of tumour growth in 2-D on an embedded vascular network is shown in Figure 5. To illustrate how such a simulation is set up in our framework we also provide a walk-through in pseudocode of the main steps involved in Algorithm 1. The first step in setting up this simulation is to initialise the spatial mesh. The spatial mesh is then used to initialise a `Diffusibles` object, a `VesselNetwork` object and a `CellPopulation` object. Diffusible species of interest are added to the `Diffusibles` object and the `CellPopulation` object is populated with cells, each of which contain various intracellular chemicals: proteins involved in the cell-cycle and species which diffuse throughout the model domain. A `StructuralAdaptationAlgorithm` object is then created and appropriately customised. Finally, a `Simulation` object is instantiated to which the various model elements are added before the simulation is run.

There is considerable scope for changing parameters in submodels and for removing and adding submodels. By employing C++'s class templating capabilities in our framework design, we have also ensured that simulations may be carried out in 3-D by varying a template parameter representing the dimensionality of the system. 3-D simulations of vascular tumour growth will be presented in future work.

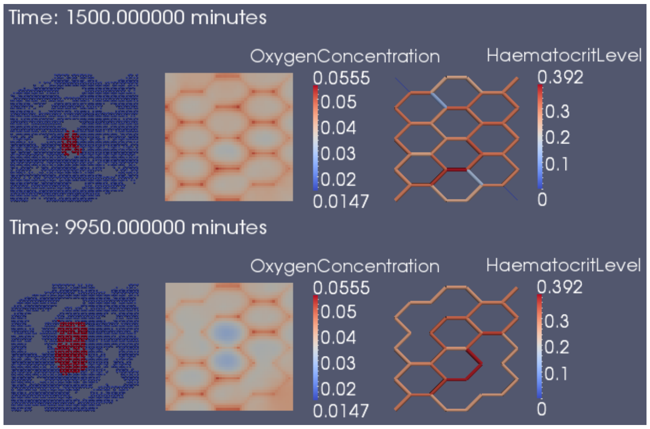


Figure 5: Two snapshots from a 2-D simulation of vascular tumour growth. Cell distributions are shown in the left-hand-side (cancerous cells are red and normal cells blue). The oxygen distributions, in terms of dimensionless concentrations, are shown in the middle panels and vessel networks in the panels on the right. Vessel colours correspond to the haematocrit level inside a vessel and the radius of a vessel is represented by its relative thickness.

A significant advantage offered by our object-oriented framework is the ability to isolate, analyse, test and validate submodels at the various biological scales which we consider. Figure 6 illustrates an example of data extracted from a simulation which considered only the progression of the Alarcón *et al.* 2005 cell-cycle model [4]. This functionality allows us to customise submodels to experimental and/or patient-specific data and facilitates further development.

V. Discussion

In order for the full potential of *in silico* models of cancer to be realised, the re-use of model code and the understandability of models must increase. In particular, attention

Algorithm 1 Example simulation pseudocode.

```
OnLatticeMesh(latticeSiteSize,domainSize_X,domainSize_Y)
Diffusibles(OnLatticeMesh)
Diffusibles.AddDiffusibleSpecies("Oxygen")
Diffusibles.AddDiffusibleSpecies("VEGF")
HexagonallyTesselatedVesselNetworkFactory(OnLatticeMesh)
HexagonallyTesselatedVesselNetworkFactory.CreateVesselNetwork()
CellPopulation(OnLatticeMesh)
for i = 1 .. numberOfCells
        Cell(location)
        Cell.AddIntraCellularChemical("Cdh1")

                ⋮

        Cell.SetMass(initialCellMass)
        CellPopulation.AddCell(Cell)
end for
StructuralAdaptationAlgorithm()
StructuralAdaptationAlgorithm.SetHaematocritCalculator(HaematocritCalculator())
Simulation(CellPopulation,VesselNetwork,Diffusibles)
Simulation.AddCellMover(Betteridge06OccupationBasedMover())
Simulation.AddSubCellularModel(Alarcon05OxygenDependentModel())
Simulation.AddDiffusionCalculator(OxygenCalculator())

                ⋮

Simulation.SetDuration(simulationRunTime)
Simulation.SetTimestep(timestep)
Simulation.Run()
```

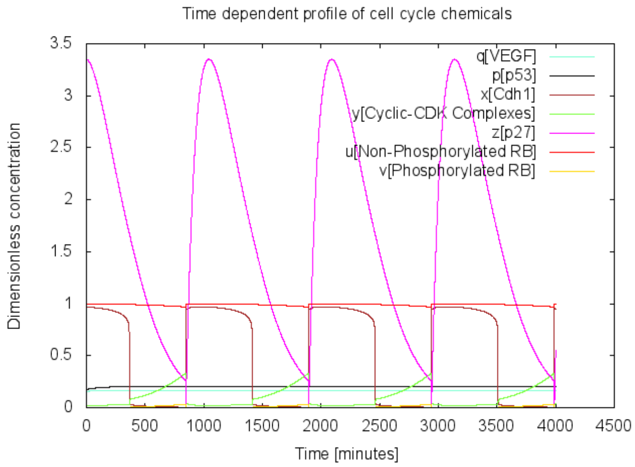


Figure 6: Time-evolution of intra-cellular chemical concentrations involved in the Alarcón *et al.* 2005 cell-cycle model [4] for a normal cell exposed to a dimensionless oxygen concentration of one.

must focus on closing the gap between experimentalists and modellers in order to encourage greater collaboration between them. An object-oriented approach to modelling cancer is ideally suited to these endeavours. That being so, motivated by the multiscale models of vascular tumour growth developed by Alarcón and co-workers, we have developed an object-oriented framework for developing and implementing multiscale models of vascular tumour growth which is both flexible and intuitive to use. Object-orientation has been used to describe elements of the application domain in a way which should be understandable to both experimental biologists and mathematical modellers with relatively little programming experience. We have provided a hierarchy of biologically-based classes which model populations of cells and vasculature, and several physics- and rule-based class hierarchies which describe how those biological entities behave and interact. We have focused on ensuring that code in our framework is understandable and maintainable, and that models implementable in our framework are easily extensible. This was accomplished by exploiting various object-oriented techniques, which have formed the focus of this paper, and by the application of several well-known design patterns, in particular, the visitor and strategy patterns. We have found that, by employing this design, we are able to move from one model implementation to another simply by defining a few new classes in pre-existing class hierarchies. This drastically reduces the development time required to produce a novel model implementation.

The development of our framework has been use-case driven, the primary use-cases being that the models of Alarcón and co-workers must be implementable in our framework. By basing our design around these models, we ensure that it will be more useful than a purely abstract design. Additionally, this family of models employs a diverse range of interchangeable algorithms for modelling various types of behaviour at multiple biological scales. This has enabled us to explore the extensibility of model code in the context of existing models. The framework provides a convenient plug-and-play environment in which a variety of different models may be implemented. Furthermore, models may be easily deconstructed into their component submodels which may be individually tested, analysed, validated and further developed before being re-integrated into the larger models of vascular tumour growth. Thus, our framework simultaneously addresses what we consider to be the two main challenges in cancer modelling; it facilitates the validation of complex multiscale models of cancer and their extension to incorporate novel data and new functionality.

Other groups have adopted similar approaches to develop multiscale modelling frameworks. In [25], an object-oriented framework was constructed in C++ to support the simulation of avascular tumour growth using an agent-based hybrid approach, and, using another object-oriented framework, Gao *et al.* [26] implemented a hybrid model of avascular and vascular tumour growth. To the best of our knowledge, however, our investigation is the first to explicitly explore the extensibility of model code in the domain of *in silico* oncology.

In wider biological fields, CompuCell3D [27] and Virtual Cell [28] are both well established and widely used frameworks, which aim to facilitate the simulation of models and re-use of model code. Chaste is another object-oriented mathematical modelling and simulation framework whose development to date has focussed on cardiac physiology and multi-cell models of tissue growth and carcinogenesis in the intestinal crypt [29]. Chaste aims to provide a reasonably generic framework for modelling biological systems. The models considered herein could be implemented within Chaste, however, the implementation would not be easily reconcilable with the real-world system. In contrast, the bottom up approach we employ allows us to minimise the conceptual distance between our model code and the corresponding real-world systems. Nevertheless, we plan to exploit existing links with Chaste and other related projects to advance the development of our framework.

Our current study addresses the re-use and extensibility of model code for a set of hybrid models of vascular tumour growth. Whilst focussing on describing the models abstractly, we have not attempted to provide a standard for describing or implementing models of this type. Thus, the models implementable in our framework are not interoperable with similar models from the wider modelling community. We will address this issue by further abstracting concepts within our application domain and developing an XML-based description of hybrid models of vascular tumour growth. By describing the models in a way which is not tied to a specific programming language and by adopting a specific ontology, perhaps extending that under development by the National Cancer Institute [30], we aim to make our framework accessible to the wider modelling community and facilitate the interoperability of model components. By choosing XML as our language of choice we will also enable the easy incorporation of models described in similar XML-based standards such as CellML [31] and the Systems Biology Markup Language (SBML) [32].

This effort will prove highly complementary to the work undertaken by the Centre for the development of a Virtual Tumour (CViT), whose Digital Model Repository (DMR) is now live. The CViT DMR has made great steps forwards in the sharing of cancer models across the entire modelling community, however, the models contained within the DMR are not interoperable since they are not written to an agreed upon standard or even in a common language. The future re-use and interoperation of models within the repository relies heavily on the adoption of a coding standard for cancer modelling which we hope to help establish. Enabling cancer modellers to download markup language (ML) descriptions of models which could be executed on a locally available solver tool, as opposed to downloading model executables, also offers significant advantages with regards to the trustworthiness of a model's execution. Whilst executables downloaded from an online repository should all individually be verified, since they carry the potential risk of having a virus or trojan embedded in them, downloaded ML descriptions need not be verified in this way because they are not capable of supporting the embedding of such malicious materials. Instead only a single tool, used to resolve the ML descriptions and run the models, need be verified.

VI. Conclusion and Future work

We have described some of the OOP techniques which we have found useful in the construction of an object-oriented framework for modelling vascular tumour growth. We hope that this paper will serve as a useful reference for biological modellers who do not have large amounts of experience with OOP, but who may benefit from employing these techniques in their own projects. We have explained the merits of various techniques, outlining specifically how each one may help programmers to produce *in silico* models which are extensible, maintainable, re-usable and understandable.

Our framework provides a convenient and intuitive plug-and-play environment in which a variety of different models may be implemented. It has laid the ground work for the further development of an XML-based domain specific language for modelling vascular tumour growth. The development of this language will form the focus of our future work. We also plan to develop a GUI in which modellers may engage with our plug-and-play functionality directly.

Acknowledgments

AJC acknowledges the financial support of the EPSRC and Hoffman la-Roche. This work was funded in part by award KUK-C1-013-04 made by the King Abdullah University of Science and Technology and by Award Number U54CAI43970 from the National Cancer Institute. We would like to thank Alexander Manta and Eliezer Shochat (Hoffman la-Roche) for their invaluable feedback.

References

[1] F. Brooks Jr, “No silver bullet: essence and accidents of software engineering,” *Computer*, vol. 20, no. 4, pp. 10–19, 1987.

[2] M. Page-Jones, *Fundamentals of object-oriented design in UML*. Addison-Wesley, 2002.

[3] T. Alarcón, H. M. Byrne, and P. K. Maini, “A cellular automaton model for tumour growth in inhomogeneous environment,” *Journal of Theoretical Biology*, vol. 225, no. 2, pp. 257–274, 2003.

[4] ——, “A multiple scale model for tumor growth,” *Multiscale Model Simulation*, vol. 3, no. 2, pp. 440–475, 2005.

[5] T. Alarcón, M. R. Owen, H. M. Byrne, and P. K. Maini, “Multiscale Modelling of Tumour Growth and Therapy: The Influence of Vessel Normalisation on Chemotherapy,” *Computational and Mathematical Methods in Medicine*, vol. 7, no. 2-3, pp. 85–119, 2006.

[6] R. Betteridge, M. R. Owen, H. M. Byrne, T. Alarcón, and P. K. Maini, “The impact of cell crowding and active cell movement on vascular tumour growth,” *Networks and heterogeneous media*, vol. 1, no. 4, pp. 515–535, 2006.

[7] M. R. Owen, T. Alarcón, P. K. Maini, and H. M. Byrne, “Angiogenesis and vascular remodelling in normal and cancerous tissues,” *Journal of Mathematical Biology*, vol. 58, no. 4-5, pp. 689–721, Apr. 2009.

[8] H. Perfahl *et al.*, “Multiscale modelling of vascular tumour growth in 3D: the roles of domain size and boundary conditions,” *PLoS ONE*, vol. 6, no. 4, p. (17 pages), 2011.

[9] J. Folkman *et al.*, “Tumor Angiogenesis: Therapeutic Implications — NEJM,” *New England Journal of Medicine*, vol. 285, no. 21, pp. 1182–1186, 1971.

[10] S. M. Peirce, “Computational and Mathematical Modelling of Angiogenesis,” *Microcirculation*, vol. 15, pp. 739–751, 2008.

[11] H. M. Byrne, “Dissecting cancer through mathematics: from the cell to the animal model,” *Nature Reviews Cancer*, vol. 10, pp. 221–230, Jan. 2010.

[12] A. L. Bauer, T. L. Jackson, and Y. Jiang, “A cell-based model exhibiting branching and anastomosis during tumor-induced angiogenesis,” *Biophysical Journal*, vol. 92, no. 9, pp. 3105–3121, 2007.

[13] P. Macklin, S. McDougall, A. R. A. Anderson, M. A. J. Chaplain, V. Cristini, and J. Lowengrub, “Multiscale modelling and nonlinear simulation of vascular tumour growth,” *Journal of Mathematical Biology*, vol. 58, no. 4-5, pp. 765–798, Sep. 2008.

[14] T. S. Deisboeck, Z. Wang, P. Macklin, and V. Cristini, “Multiscale Cancer Modeling,” *Annu. Rev. Biomed. Eng.*, vol. 13, no. 1, pp. 127–155, Aug. 2011.

[15] S. Schnell, R. Grima, and P. K. Maini, “Multiscale modeling in biology,” *American Scientist*, vol. 95, no. 2, pp. 134–142, 2007.

[16] M. Fowler, *UML Distilled*, 3rd ed. Addison-Wesley, 2006.

[17] B. Liskov, “Keynote address-data abstraction and hierarchy,” *ACM Sigplan Notices*, vol. 23, no. 5, pp. 17–34, 1987.

[18] K. Webb and T. White, “UML as a cell and biochemistry modeling language,” *BioSystems*, vol. 80, no. 3, pp. 283–302, Jun. 2005.

[19] A. Gupta, S. Chempath, M. J. Sanborn, A. Clark, and R. Q. Snurr, “Object-oriented Programming Paradigms for Molecular Modeling,” *Molecular Simulation*, vol. 29, no. 1, pp. 29–46, 2003.

[20] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, “Design patterns: elements of reusable object-oriented software,” *Reading: Addison Wesle*, 1995.

[21] M. R. Owen *et al.*, “Mathematical modelling predicts synergistic antitumor effects of combining a macrophage based, hypoxia-targeted gene therapy with chemotherapy,” *Cancer Research*, vol. 71, no. 8, pp. 2826–2837, Apr. 2011.

[22] “http://www.boost.org,” 14.09.2012.

[23] “http://www.mcs.anl.gov/petsc,” 14.09.2012.

[24] “http://www.paraview.org,” 14.09.2012.

[25] A. C. Abajian and J. S. Lowengrub, “An agent-based hybrid model for avascular tumor growth,” *UCI Undergrad. Res. J*, vol. 11, 2008.

[26] X. Gao, M. Tangney, and S. Tabirca, “2D simulation and visualization of tumour growth based on discrete mathematical models,” *Bioinformatics and Biomedical Technology (ICBBT), 2010 International Conference on*, pp. 35–41, 2010.

[27] J. A. Izaguirre *et al.*, “COMPUCELL, a multi-model framework for simulation of morphogenesis,” *Bioinformatics*, vol. 20, no. 7, pp. 1129–1137, Apr. 2004.

[28] I. I. Moraru *et al.*, “Virtual cell modelling and simulation software environment,” *Systems biology, IET*, vol. 2, pp. 352–362, 2008.

[29] J. Pitt-Francis *et al.*, “Chaste: A test-driven approach to software development for biological modelling,” *Computer Physics Communications*, vol. 180, no. 12, pp. 2452–2471, Nov. 2009.

[30] J. Golbeck *et al.*, “The National Cancer Institute’s Thésaurus and Ontology,” *Journal of Web Semantics*, 2003.

[31] A. Garny *et al.*, “CellML and associated tools and techniques,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 366, no. 1878, pp. 3017–3043, Sep. 2008.

[32] M. Hucka *et al.*, “The Systems Biology Markup Language (SBML): Language Specification for Level 3 Version 1 Core (Release 1 Candidate),” *Nature Precedings*, Jan. 2010.

Capacity Planning for Elderly Care in Ireland Using Simluation Modeling

Mohamed A.F. Ragab, Waleed Abo-Hamad, and Amr Arisha
3S Group, College of Business
Dublin Institute of Technology
Dublin 2, Ireland
Email: mohamed.ragab@mydit.ie, waleed.abohamad@dit.ie, amr.arisha@dit.ie

Abstract— **Global population aging is creating an immense pressure on healthcare facilities making them unable to cope with the growing demand for elderly healthcare services. Current demand-supply gaps result in prolonged waiting times for patients and substantial cost burdens for healthcare systems due to delayed discharges. This paper describes a project aimed at presenting modeling and simulation to address elderly care pathways within the Irish healthcare sector. The management of frail patients admitted to acute hospitals and the introduction of the new intermediate care beds are alternative interventions that healthcare executives are interested in simulating to examine their impact on the performance of the elderly care system. The developed simulation model, along with the statistical analysis, have enabled the management to assess the current system under the critical financial and performance issues. It also highlights the decision variables that significantly improve the flow of elderly patients.**

Keywords - Population Ageing; Elderly Care; Discrete Event Simulation; Discharge Planning.

I. Introduction

Advances in pharmaceutical and medical technology during the past century have caused a major shift in global demographics by increasing life expectancy to unprecedented figures. The result is that there are more elderly people today than ever before [1]. Furthermore, the world's elderly population is expected to grow from 650 million to 2 billion people by 2050 [2]. In Europe, there are currently 108 million elderly people who constitute 15% of the European population, and this figure is forecasted to increase to 26% by 2050 [3]. A similar trend is expected in Ireland where the elderly population is projected to grow from 500,000 to 1.3 million over the next 30 years [4]. Although elderly patients represent 11% of the Irish population, they account for up to 50% of hospital bed usage [4]. Consequently, pressures are now on Irish hospitals, not only due to the increase in demand for acute hospital beds, but also because elderly patients use acute hospital resources disproportionately. In response to such demographic changes, hospitals in Ireland are striving to fill the existing supply-demand gap while maintaining their quality of service [4]. Global economic crisis implied a severe cut in healthcare funds and elicited a limited resource policy in hospitals and other healthcare services. This caused Irish hospitals and elderly healthcare facilities to equally face a grave capacity planning issue to respond to the increased demand.

The subsequent shortage of beds has numerous facets that impact adversely on the overall performance of the Irish healthcare system. Firstly, it has a significant impingement to Emergency Department (ED) overcrowding, a problem that has detrimental consequences including higher mortality rates for elderly patients [5]. Secondly, shortage of community care beds leads to delayed discharges from acute hospitals, which not only delays new admissions into hospitals, but also burdens hospitals with high unjustified costs since acute beds are considered among the most expensive resources of the entire healthcare system [6]. Finally, delays due to the lack of short-term and long-term bed supply create substantial waiting times in most stages of the healthcare system. Long waiting times in elderly care services, as well as other services, are the most frequent complaints reported by patients to healthcare executives every year [4].

Simulation models have been proven to be an excellent and flexible tool for modeling processes in such stochastic complex environments [7]. Healthcare managers can apply simulation for assessing current performance, predicting the impact of operational changes, and examining the tradeoffs between system variables [8]. Furthermore, areas of improvement can be identified using simulation models through possible reorganization and allocation of existing resources [9][10]. On a micro-level, simulation is well-suited to tackle problems in hospital departments such as emergency departments [11][12] and operating rooms [13], where resources are scarce and patients arrive at irregular times [14]. Various alternatives and interventions can be evaluated and tested effectively [15]. A more accurate interpretation of the utilization of hospital resources can be envisaged using dynamic capabilities of simulation [16], which in turn supports the hospital management in their decisions on bed usage and patient flow [17]. This can be achieved by modeling the flow of patients through the hospital [18] then using scenarios to illustrate the consequences of possible potential decisions suggested by hospital management [19].

This paper presents a project implemented to support Irish health executives when decisions are to be made regarding elderly care in the Irish system. The developed framework enables the directors to examine the dynamics embedded in the system through the modeling of patients' care pathways. The model also highlights the high level of variability within patients demand and limitations of available resources within healthcare facilities. It aims to provide a comprehensive capacity-planning tool that can be

ISBN: 978-1-61208-234-9

used to assess proposed strategies to handle the existing bottlenecks and improve the overall experience of elderly patients. The underlying objective of this project is to introduce a tool that will contribute into significant improvements in elderly patient service by increasing throughput and reducing waiting times and operations costs.

The paper begins by introducing the conceptual model depicting the elderly patient journey within the healthcare system and the different discharge destinations. The development phases of the simulation model are then presented starting by data collection, coding, then validation. Using the model, two scenarios proposed by healthcare policy makers to improve patient flow are then examined followed by a design of experiment and statistical analysis to determine the most significant factors that affect patient flow and the magnitudes of their impact. Finally, the paper's findings are reported along with recommendations for future work.

II. Problem Contextualization

A. *Background*

Elderly patients are usually defined as those who are aged 65 and older and this convention is used in this study [20]. The most challenging of elderly patients are those referred to as *frail*. Frailty is characterized by suffering from an array of medical conditions that individually may be curable, but collectively create an overwhelming and complex burden of disease [1]. Frail patients constitute 18-20% of the Irish elderly population and usually require treatment for an extended period of time in the healthcare facility followed by rehabilitation and/or community care. Adhering to the length of stay (LOS)-based cut-off point, frail patients were characterized in this study by a treatment period of more than 15 days in the acute system (i.e., hospitals). The remaining 80 - 82% of elderly patients who receive treatment for less than 15 days are referred to as *non-frail.*

The initial scope of this study focused solely on frail patients, however, since all 65+ patients utilize the same resources, it was imperative to widen the project's scope to encompass all 65+ elderly patients, both frail and non-frail. Although elderly patients utilize a wide range of resources, the initial phase of the proposed model gives special attention to bed capacity within healthcare facilities based on a request from healthcare executives. Accordingly, elderly care services that do not necessitate admission, such as outpatient clinics, are excluded from the model because they do not affect hospital bed utilization.

B. *Conceptualization*

The journey of an elderly patient usually begins with their arrival at the ED by ambulance, walk-in or a referral by a General Practitioner (GP). After admission, elderly patients receive treatment in an acute bed until their care pathway is assigned subject to their diagnosis and frailty level. The duration of this process ranges from few days to two weeks for non-frail patients, but usually exceeds 45 days for frail patients.

Following their stay in acute beds, elderly patients are discharged to one of the following destinations:

Another Hospital: Certain medical procedures may require equipment that is not available in the acute hospital to which an elderly patient has been admitted. In such a case, elderly patients are transferred to another hospital where the required technology is available to undertake the procedure they need. Discharge figures to another hospital (6% of all elderly patients) include patients being moved to undergo a certain procedure, and patients who have received such a procedure and are returning to their original hospital.

Rehabilitation: Patients who are deemed to be in a frail status but have the potential to improve their functional independence are discharged to an on-site or off-site facility where they receive rehabilitation. Rehabilitation is an intermediate destination in which frail patients are no longer categorized as acutely ill, but still need close medical observation with hope that they would recover [21]. After rehabilitation, the majority of patients (80%) are discharged home and the remaining 20% who have not recovered are discharged to long term care.

Convalescence: Around 10% of non-frail patients are usually discharged to a convalescent care facility for a short stay during which they would recover from a medical procedure. Compared to rehabilitation, convalescence offers less intensive care as it prepares patients to go home. In several cases, convalescence may take place within a nursing home facility on dedicated short stay beds.

Long Term Care (LTC): More than a quarter of frail elderly patients would not be able to live alone at their homes because they are unable to care for themselves or sometimes require perpetual medical supervision. They are discharged to a public or private nursing home to receive LTC and usually stay there for years until they die. This prolonged stay in nursing homes hampers the supply of LTC beds into the healthcare system resulting in waiting times that amount to several months. In addition to hospital demand, there is also a community demand in which frail patients apply for LTC and wait in their homes until they are placed in a nursing home.

Home: The vast majority of non-frail elderly patients are eventually discharged to their homes, whether directly or after a short stay in convalescence. On the other hand, 24% of frail patients are directly discharged to their homes followed by another 28% that go home after rehabilitation. More than half of frail patients continue to require medical care within their own homes and thus are provided Home Care Packages (HCP). A HCP comprises a set of services provided by the state that may include home help, nursing, physiotherapy, occupational therapy and other services [4].

Consequently, shortages in rehabilitation, convalescence, LTC and HCP capacity are the main reasons behind delayed discharges from acute hospitals. Elderly patients occupy acute beds for an extended LOS that exceeds their treatment period not because they require acute health services, but because they are waiting to be discharged [22]. The alternative care pathways and their required bed resources are illustrated in Figure 1. The percentages of patients discharged to each destination are then listed in Table 1.

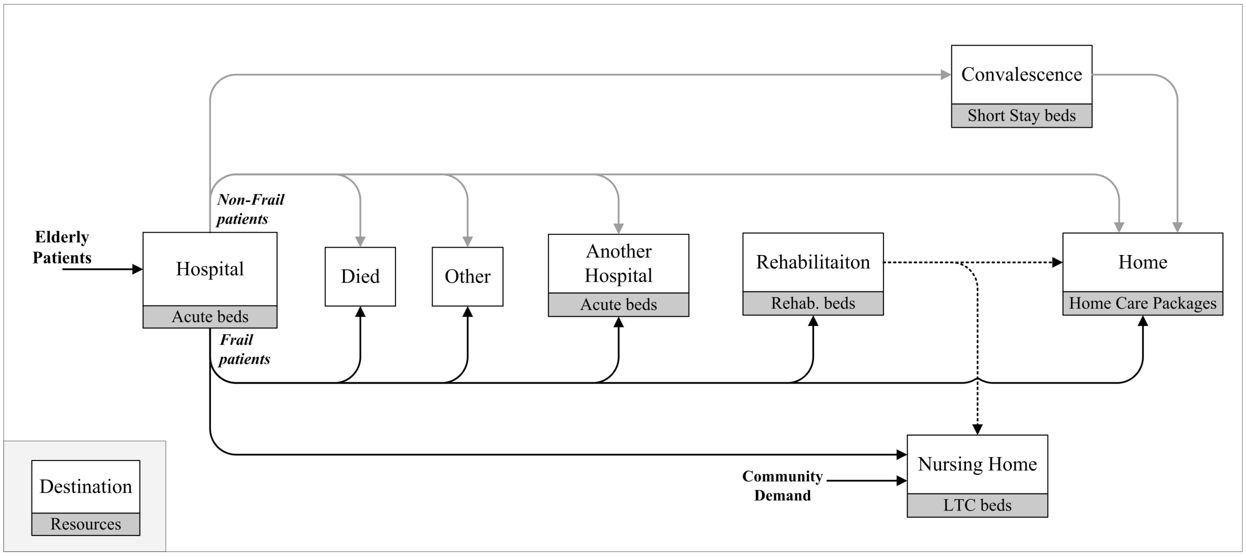


Figure 1. Elderly care pathways

TABLE I. DISCHARGE DESTINATION PERCENTAGES

Discharge Destinations	Percentage of Patients		
	Frail	*Non-frail*	*All 65+ Patients*
Home	24.2 %	78.4 %	68.6 %
Another Hospital	8.2 %	5.7 %	6.1 %
Rehabilitation	36 %	0%	6.5 %
Convalensence	0 %	10.5 %	8.6 %
Long Term Care	19.5 %	0 %	3.5 %
Died	10.8 %	4.3 %	6.1 %
Other	1.3 %	1.1 %	1.1 %

In addition to the previous discharge destinations, 6% of elderly patients may die during their acute stay, with the probability of mortality increasing proportionally with the frailty level. Another minimal number of patients (almost 1%) who have special conditions are discharged to destinations referred to as "*other*" such as a prison or psychiatric facility.

III. SIMULATION MODEL

A. Data Collection

Data quality and precision determines the validity of the simulation model. Hence, the data collection phase represents a critical milestone of any simulation project. Historical admission and discharge data was collected from the central healthcare information system, while bed capacities and LOS data were gathered through surveys. Similar to other healthcare modeling projects, collection of relevant modeling data presented considerable challenges [23]. The first was the dearth of data about certain parameters that were not captured by the central information system. It is worthy to note that a similar project undertaken in the UK to study elderly care diverted its objective from producing quantitative results to only building a simulation model due to the lack of relevant data [21]. The second challenge was data provided in aggregate figures while modeling inputs required them to be broken down into their individual elements. An example was the combination of the numbers of patients discharged to multiple destinations into one numerical figure. The third problem with data was inconsistencies found between different data sources such as variations in figures between hospital data and annual reports. After numerous extensive meetings with hospital officials, the absence of certain data and lack of information on how to decompose aggregated figures were overcome by the use of assumptions based on the opinions of experts in the field [24]. By gaining a deeper understanding of what each figure reflected, in most cases misunderstandings of terminology or scope were the reasons behind what seemed to be inconsistencies in the data.

Patient information was extricated from the raw data by data manipulation and reorganization. Data analysis was then performed to extrapolate important inputs for the model including arrival and discharge patterns, and to segment frail patient data. By clustering this data, frail patients were grouped according to their acute LOS into four categories coded numerically from zero to four, each representing a *degree of complexity* based on the validated assumption that the most complex cases spend more time in hospital. All 65+ patient data was also categorized by age group into five clusters. Based on the data analysis and segmentation, elderly patients' degree of complexity and age group would be used during simulation to define their care pathways within the model. The percentages of patients classified with each degree of complexity are shown in Table 2.

TABLE II. DEGREE OF COMPLEXITY PERCENTAGES

Degree of Complexity	Percentage of Patients		
	Frail	*Non-frail*	*All 65+ Patients*
0	0 %	100 %	82 %
1	42 %	0 %	8 %
2	20 %	0 %	4 %
3	17 %	0 %	3 %
4	21 %	0 %	4 %

B. Model Development and Validation

Based on the conceptual model and the empirical data analysis, a comprehensive discrete-event simulation model was constructed using a simulation package and an input/output *Excel* spreadsheet was developed as a user-friendly interface. Modules of the simulation model were connected similar to the conceptual flow chart, which eases the model construction phase. Accordingly, the top-level of the simulation model defined the overall model structure, and sub-level blocks comprised additional modules with more details. Object-oriented programming was used to customize pre-defined blocks for constructing the simulation model. The main entities for the simulation were elderly patients, where each patient is assigned a set of attributes that represent their degree of complexity and age group to determine their discharge destination. Statistical assumptions were included by using a Poisson distribution for the arrival rate and exponential distributions for service times [22]. The time unit used was *days* for all modeling inputs and outputs. A database was used to save the measured Key Performance Indicators (KPI) after each simulation run, followed by exporting the KPIs in a tabular form for further analysis and validation.

To reduce the model development cycle time and to increase the confidence in the simulation model results, verification and validation were carried out all the way through the development phase to confirm the model represents the actual patient flow [23]. After each model development phase, the model was verified and validated with respect to other previously completed phases. For the verification process, the model logic was verified to ensure that patients followed the correct care pathway as expected. This was achieved by visual tracking of patients using animation and by checking intermediate output values such as queue lengths and waiting times.

Initially, queues at each stage of patient care were set as empty and idle. A warm-up period of three months was found to mitigate any bias introduced by the initial conditions of the simulation model until the steady state was achieved. In order to be comparable with the provided data, results for one year were generated for each scenario by running the model for 465 days and discarding the results of the first 100 days that represented the warm up period. Different number of runs (i.e., replicates) were tested and it was found that 10 runs per scenario were sufficient to obtain unbiased estimators of the expected average of each KPI.

IV. SCENARIOS

To improve patient flow, a number of strategies were proposed by the project team. Examined scenarios in addition to performance metrics are presented in this section.

A. Key Performance Indicators

Although the model produced a portfolio of results, the following KPIs that focus on acute hospital measures were selected:

- *Acute waiting time*: the average time spent by patients waiting for admission to an acute hospital.
- *Acute access*: the ratio of admitted elderly patients to the demand for admission.
- *Throughput rate*: the total number of elderly patients discharged per year.
- *Average cost per patient*: this cost perspective was added to the model to reflect financial effects of different scenarios. The average cost per patient was calculated by dividing total cost incurred through bed usage by the total number of discharged patients.

Due to data confidentiality of the project, the results reported for each scenario in this paper have been anonymized by normalization by setting the current "*as-is*" values at one and reporting scenario results as percentages relative to the as-is figure.

B. Shorter Acute LOS for Frail Patients

One of the first strategies that the management team proposed to improve patient flow was to set a target of maximum acute LOS for frail elderly patients, whose current LOS exceeds 45 days. In such a case, hospitals would be instructed to make earlier decisions about an elderly patient's medical needs and degree of frailty to accelerate their discharge from hospital. A scenario was tested assuming that frail elderly patients would have a maximum acute LOS of 18 days, slightly longer than non-frail patients.

The results of testing this scenario, presented in the bar chart in Figure 2, show some improvement in patient flow. Throughput rate and acute access have increased by 6% and 8% respectively, while acute waiting time and cost/patient have decreased with similar percentages. Performance improvement in this scenario could be viewed as somewhat limited due to the fact that frail patients whose LOS currently exceeds 18 days constitute 54% of all frail patients and only 10% of the entire elderly population and therefore reducing this duration would not have a major global impact on the efficiency of the entire system.

Despite their interest in testing this scenario, healthcare policy makers foresaw its drawbacks. The dependence of acute LOS on patient diagnosis and required medical procedures could hamper the implementation a maximum LOS policy and may face resistance from medical staff. Hence, other more effective and pertinent solutions should be sought.

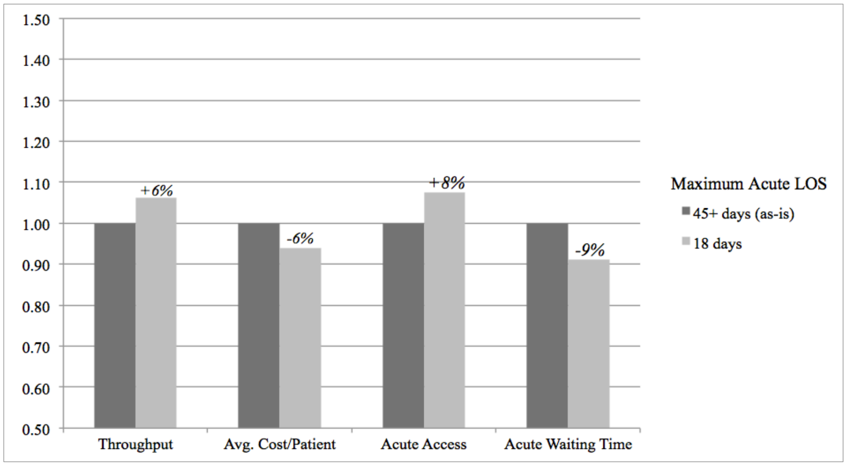


Figure 2. Impact of reducing maximum acute LOS to 18 days

C. *Intermediate Care*

The second proposed strategy was the introduction of a new service similar to *Intermediate Care* in the UK that can serve patients who require an acute or rehabilitation bed for prolonged periods only because they are awaiting discharge to LTC [21]. Intermediate care beds will be mostly located offsite and will provide a transitional venue where frail elderly patients can spend time before they will be placed in a long term facility. The anticipated advantage of intermediate care is reducing the overall time spent by elderly patients in hospitals. This should result in significant cost savings since the operational cost of an intermediate care bed is estimated to be almost half of the cost of an acute bed for the same period of time. To assess the impact of this service on the elderly care system, different scenarios were examined using the developed simulation model where each experiment used a set of different capacities of intermediate care beds. The gradient increase in intermediate care beds is proportional to the static number of acute beds in the system starting with 5% of the acute bed capacity and increasing the intermediate care-to-acute bed ratio up to 20% in subsequent scenarios.

Introducing intermediate care beds appears to have an overall positive effect on patient flow by noticeably increasing throughput rate and acute access to up to 2.5 times while reducing acute waiting time and cost/patient to up to 50% of the current figures as shown in Figure 3. Intermediate care beds reduce waiting times for acute admission and rehabilitation because they accelerate the release of acute and rehabilitation beds back into the system. This results in having more beds available for the incoming demand. Despite the fact that intermediate care would be the last stage that precedes LTC, it is observed that it has almost no effect on LTC waiting time. This was not unexpected, as LTC waiting time is constrained by LTC bed supply, regardless of *where* elderly patients would wait for LTC placement.

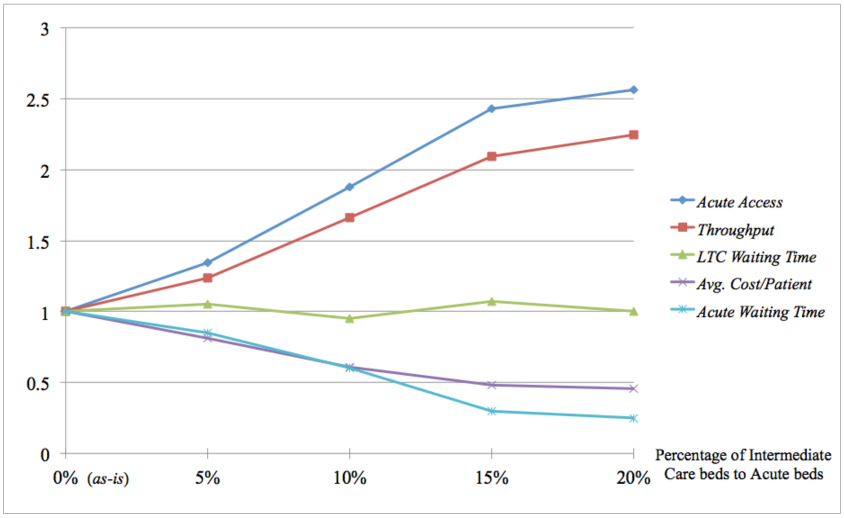


Figure 3. Effect of introducing intermediate care beds on KPIs

D. *Design of Experiments and ANOVA*

In addition to evaluating the previous strategies, there was an interest from healthcare executives to gain insights of the dynamics of the elderly care system and also to identify the most significant factors that affect its overall performance. Using an orthogonal array (L27) a fractional factorial design of experiment was conducted [25][26]. The L27 design allows for up to 13 factors where each factor is tested at three levels: high, medium and low (*H-M-L*). Six selected factors were tested and the values for H-M-L levels where determined in relation to the current state figures, where one of the three levels was set as the as-is value. Based on the selected orthogonal array, twenty-seven experiments were carried out and the response (i.e., output) measured in each experiment was the system's throughput rate, as recommended by healthcare executives. This was followed by a six-way Analysis Of Variance (ANOVA) test to determine the significance of the six selected factors (Table 3). To mitigate the inflation of error, Bonferroni correction was used to compute the significance level (α) using the following equation:

$$\alpha = \alpha[PT] / n \qquad (1)$$

where *α[PT]* is the significance level per test (i.e., alpha per experiment) and *n* is the number of comparisons [27]. With α[PT] = 0.05 and n = 18, the equation produces a significance level of α = 0.00278.

TABLE III. ANOVA TEST RESULTS FOR THROUGHPUT RATE

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Squares	F Ratio	P Value
Model	18	0.8678	0.0482	16.369	0.0002
A: Acute bed capacity	1	0.0272	0.0272	9.2441	0.0161
B: Rehab. bed capacity	1	0.0037	0.0037	1.2573	0.2947
C: LTC bed capacity	1	0.0011	0.0011	0.3566	0.5669
D: Acute LOS	1	0.0102	0.0102	3.4698	0.0995
E: Rehab. LOS	1	0.0671	0.0671	22.801	0.0014
F: Percentage of rehab. patients	1	0.0617	0.0617	20.965	0.0018
AB	1	0.0029	0.0029	1.0005	0.3465
AC	1	0.0056	0.0056	1.9165	0.2036
AE	1	0.0038	0.0038	1.2824	0.2903
AF	1	0.002	0.002	0.6796	0.4336
BC	1	0.0054	0.0054	1.8211	0.2141
BE	1	0.0033	0.0033	1.1083	0.3232
BF	1	0.009	0.009	3.065	0.1181
CE	1	0.0057	0.0057	1.9457	0.2006
CF	1	0.0055	0.0055	1.8736	0.2083
DE	1	0.0062	0.0062	2.1218	0.1833
DF	1	0.0055	0.0055	1.8607	0.2097
EF	1	0.0141	0.0141	4.7899	0.0601
Residual	8	0.0236	0.0029		
Lack of Fit	8	0.0236	0.0029		
Total	26	0.8913			
Significance level α = 0.00278					

Accordingly, ANOVA results illustrate that the LOS in rehabilitation and percentage of patients that receive rehabilitation are the only significant factors that affect the throughput rate as indicated by P-values that are lower than the computed significance level. Results also clarify that there are no significant interactions between any two factors.

Regression analysis followed the ANOVA test to determine the relative impact of the significant factors on the throughput rate when moving from one level to another. The regression produced negative coefficients for the two significant factors indicating that they are both inversely proportional to the throughput rate. This is explained by the fact that increasing the percentage and LOS of rehabilitation patients decreases throughput rate as less patients are discharged per year. The negative correlations of both factors with the throughput rate are plotted in Figure 4.

V. CONCLUSION AND FUTURE WORK

Healthcare executives in Ireland are confronted by a critical capacity planning challenge due to the mounting demand for elderly healthcare services instigated by population ageing. Developing a simulation model to investigate the service constraints was found to be a well-suited approach to provide decision makers with a tool to evaluate proposed strategies. Conceptual modeling was used to illustrate different elderly patient care pathways and provide a better understanding of resources required during the care journey. This phase was followed by developing a discrete-event simulation model with an objective of investigating the impact of demand uncertainty on available capacity. The model was of great benefit to policy makers in forecasting the outcomes of potential strategies that were under investigation. The reduction of average length of stay of patients using acute beds in hospitals, if possible, can offer a mediocre improvement in patient flow. Results have also shown that the introduction of intermediate care beds can enhance the system's performance significantly by reducing delays and patient cost of stay by almost 50%. Moreover, the proposed model has the potential to fully examine the economic feasibility of implementing this intermediate bed solution based on a cost-benefit analysis in addition to any other scenarios proposed by policy makers.

An ANOVA statistical analysis revealed that the rehabilitation phase is a bottleneck that affects untoward patient flow. It could therefore be concluded that efforts to improve the flow of elderly patients within the healthcare system should be directed more towards rehabilitation rather than other stages of the patient treatment journey. Hence, it is strongly recommended that future research would study the impact of the rehabilitation stage and its capacity on patient throughput. Potential strategies to be considered include setting a maximum rehabilitation LOS and transferring a number of acute beds to rehabilitation.

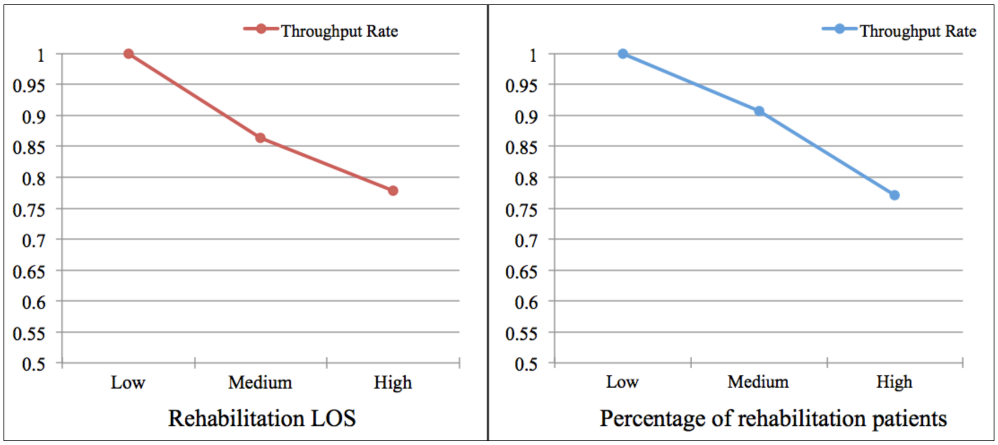


Figure 4. Effect of significant factors on throughput rate

It is worth mentioning that the main challenge in this study was the data collection phase. Problems varied between irrelevant, insufficient, or accuracy issues. In several instances, the lack of data was overcome by relying on assumptions made by healthcare experts. Comprehensive and periodic collection of elderly patient data is strongly recommended to provide decision makers with a solid foundation to use for process improvement strategies. Furthermore, a detailed cost analysis was not possible in this phase of the study due to two main reasons; (1) lack of cost related information and, (2) the high variability in cost models used within Irish public hospitals that creates a high level of complexity. Nevertheless, a recently launched project within the same research group will attempt to create a financial model for public hospitals in Ireland to facilitate cost analysis and optimization.

ACKNOWLEDGMENTS

The authors would like to thank Pharos University in Alexandria for its support of this research. Special thanks to Dr. Amani Gomaa for her valuable contribution in the analysis of results.

REFERENCES

[1] R. C. McDermid and S. M. Bagshaw, "ICU and critical care outreach for the elderly," Best practice & Research Clinical Anaesthesiology, vol. 25, Sep. 2011, pp. 439-49, doi:10.1016/j.bpa.2011.06.001.

[2] World Health Oorganization. "Ten facts on ageing and the life course," Mar. 2012, Available: http://www.who.int/features/factfiles/ageing/en/index.html/, retrieved September, 2012.

[3] R. D. Piers et al., "Advance care planning in terminally ill and frail older persons," Patient Education and Counseling, Aug. 2011, doi:10.1016/j.pec.2011.07.008.

[4] Health Service Executive. "Annual report and financial statements," Mar. 2012, Available: http://www.hse.ie/eng/services/Publications/corporate/annualrpt 2010.html/, retrieved August, 2012.

[5] D. B. Richardson, "Increase in patient mortality at 10 days associated with emergency department overcrowding," Medical Journal of Australia, vol. 184, no. 5, Mar. 2006, pp. 213.

[6] G. Liotta, S. Mancinelli, P. Scarcella, and L. E. Gialloreti, "Determinants of acute hospital care use by elderly patients in Italy from 1996 to 2006," Archives of Gerontology and Geriatrics, vol. 54, issue 3, Sep. 2011, pp. 364-369, doi: 10.1016/j.archger.2011.08.001.

[7] A. Arisha and P. Young, "Intelligent simulation-based lot scheduling of photolithography toolsets in a wafer fabrication facility," Winter Simulation Conference, IEEE, Dec. 2004, pp. 1935-1942, doi: 10.1109/WSC.2004.1371552.

[8] A. P. Wierzbicki, "Modelling as a way of organising knowledge," European Journal of Operational Research, vol. 176, no. 1, 2007, pp. 610-635, 2007, doi:10.1016/j.ejor.2005.08.018.

[9] P. R. Harper, A. K. Shahani, J. E. Gallagher, and C. Bowie, "Planning health services with explicit geographical considerations: A stochastic location-allocation approach," Omega, vol. 33, no. 2, 2005, pp. 141-152, doi: 10.1016/j.omega.2004.03.011.

[10] J. E. Stahl et al., "Reorganizing the system of care surrounding laparoscopic surgery: A cost-effectiveness analysis using discrete-event simulation," Medical Decision Making, vol. 24, no. 5, 2004, pp. 461-471, doi: 10.1177/0272989X04268951.

[11] K. Ismail, W. Abo-Hamad, and A. Arisha, "Integrating balanced scorecard and simulation modeling to improve emergency department performance in Irish hospitals," Winter Simulation Conference, IEEE, Dec. 2010, pp. 2340-2351, doi: 10.1109/WSC.2010.5678931.

[12] S. Samaha, W. S. Armel, and D. W. Starks, "The use of simulation to reduce the length of stay in an emergency department," Winter Simulation Conference, IEEE, Dec. 2003, pp. 1907-1911, doi: 10.1109/WSC.2003.1261652.

[13] J. P. M. Arnaout and S. Kulbashian, "Maximizing the utilization of operating rooms with stochastic times using simulation," Winter Simulation Conference, IEEE, Dec. 2008, pp. 1617-1623, doi: 10.1109/WSC.2008.4736245.

[14] J. Jun, S. Jacobson, and J. Swisher, "Application of discrete-event simulation in health care clinics: A survey," Journal of the Operational Research Society, vol. 50, no. 2, Feb. 1999, pp. 109-123, doi: jstor.org/stable/3010560.

[15] J. R. Swisher and S. H. Jacobson, "Evaluating the design of a family practice healthcare clinic using discrete-event simulation," Health Care Management Science, vol. 5, no. 2, Apr. 2002, pp. 75-88, doi: 10.1023/A:1014464529565.

[16] M. Thorwarth, A. Arisha, and P. Harper, "Simulation model to investigate flexible workload management for healthcare and servicescape environment," Winter Simulation Conference, IEEE, Dec. 2009, pp. 1946-1956, doi: 10.1109/WSC.2009.5429210.

[17] P. R. Harper, "A Framework for operational modelling of hospital resources," Health Care Management Science, vol. 5, no. 3, Aug. 2002, pp. 165-173, doi: 10.1023/A:1019767900627.

[18] D. Retzlaff-Roberts and S. Ezelle, "A simulation case study of patient flow at the University of South Alabama Medical Center," Winter Simulation Conference, IEEE, Dec. 2007, pp. 2391-2391, doi: 10.1109/WSC.2007.4419906.

[19] M. M. Gunal and M. Pidd, "Interconnected DES models of emergency, outpatient, and inpatient departments of a hospital," Winter Simulation Conference, IEEE, Dec. 2007, pp. 1461-1466, doi: 10.1109/WSC.2007.4419757.

[20] M. Rosenberg and J. Everitt, "Planning for aging populations: inside or outside the walls," Progress in Planning, vol. 56, no. 3, Oct. 2001, pp. 119-168, doi: 10.1016/S0305-9006(01)00014-9.

[21] K. Katsaliaki, S. Brailsford, D. Browning, and P. Knight, "Mapping care pathways for the elderly," Journal of Health Organisation and Management, vol. 19, no. 1, 2005, pp. 57-72, doi: 10.1108/14777260510592130.

[22] Y. Zhang, M. L. Puterman, M. Nelson, and D. Atkins, "A simulation optimization approach for long-term care capacity planning," Working Paper, University of British Columbia, 2010.

[23] S. D. Roberts, "Tutorial on the simulation of healthcare systems," Winter Simulation Conference, IEEE, Dec. 2011, pp. 1403-1414, doi: 10.1109/WSC.2011.6147860.

[24] C. R. Standridge, "A tutorial on simulation in health care: Applications and Issues," Winter Simulation Conference, IEEE, Dec. 1999, pp. 49-55, doi: 10.1109/WSC.1999.823051.

[25] G. Taguchi and S. Konishi, Orthogonal arrays and linear graphs: tools for quality engineering: American Supplier Institute Allen Park, MI, 1987.

[26] F. F. Baesler, H. E. Jahnsen, and M. DaCosta, "The use of simulation and design of experiments for estimating maximum capacity in an emergency room," Winter Simulation Conference, IEEE, Dec. 2003, pp. 1903-1906, doi: 10.1109/WSC.2003.1261651.

[27] H. Abdi, "Bonferroni and sidak corrections for multiple Comparisons," Encyclopedia of Measurement and Statistics, Thousand Oaks: Sage, 2007, pp103-107.

A Whole Trajectory Simulation for the Electromagnetic Rail Gun

Ping Ma, Ming Yang, Yuwei Hu
Control and Simulation Center
Harbin Institute of Technology
Harbin, P.R.China
pingma@hit.edu.cn, myang@hit.edu.cn, 2006huyw@163.com

***Abstract*—With the improvement of electromagnetic launch technology, electromagnetic rail gun will emerged in future battlefield. A whole trajectory simulation is proposed in order to analyze ballistic characteristics. According to analysis of the whole trajectory simulation process of electromagnetic rail gun, the whole trajectory models including inner ballistics model and exterior model are developed to perform simulation task. Several simulation experiments are accomplished under the developed system. Simulation results show that the system can simulate the whole trajectory of electromagnetic rail gun. Meanwhile, the whole trajectory simulation system provides a foundation for the ballistics integration design of electromagnetic rail gun system.**

Keywords-Electromagnetic Rail Gun (EMRG); Simulation System; Whole Trajectory Simulation; Ballistics Model.

I. INTRODUCTION

Electromagnetic launch is a new concept in military technology, which utilizes electromagnetic force to accelerate the projectile and transform the electric energy of power into kinetic energy of launch package to complete many missions [1]. The system consists of energy storage devices, pulsed power, rails, armature, projectile and other related devices. Pulsed power, the core of Electromagnetic Rail Gun (EMRG), is controlled by adjusting the working state to perform charging and acceleration mission. Pulsed power, composed of sets of high-capacity capacitor modules, thyristor switch, pulse-shaping inductor, crowbar diode, can generate very powerful instantaneous current pulse. When the rail is connected power, current flows into armature along one rail and flows back along the other to form close circuit. When strong currents flow along the two parallel rails, a great magnetic field, formed between the two rails, interacts with the current flowing through armature to generate powerful electromagnetic force. This force accelerates armature and projectile along the rails to the bore with high speed, as shown in Figure 1.

EMRG has obvious advantages. Firstly, it has strong survivability and is very safe in application. Secondly, the launch velocity can be controlled by the magnitude of current to increase the shooting accuracy. Thirdly, it has low costs and allows continuous multi-shot. Lastly, the projectile shot from the bore with high initial speed can achieve wide range.

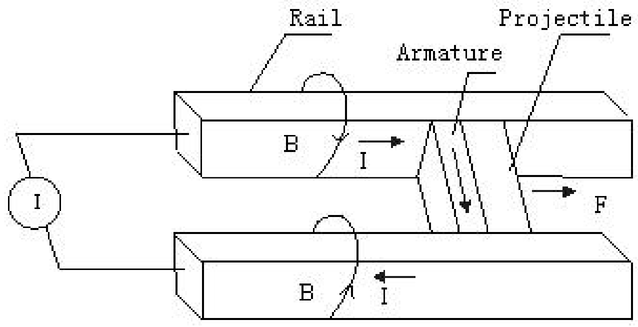


Figure 1. Principle of electromagnetic rail gun

Currently, the development of electromagnetic rail gun mainly focuses on key components, such as pulsed power, rail, armature and projectile [2-4], while researches on ballistics performance, especially on whole-trajectory characteristics, are relatively weak.

The Department of the Navy's science and technology corporate board chartered the Innovative Naval Prototype (INP) construct to foster game-changing and disruptive technologies ahead of the normal requirements process [5].

This paper analyzes the whole trajectory simulation process of EMRG system and, the system requirements of EMRG simulation are determined. The ballistics model including inner ballistics model and exterior ballistics model is established. Then, the EMRG simulation is realized to perform the whole-trajectory simulation and analysis.

II. WHOLE TRAJECTORY SIMULATION PROCESS OF EMRG SYSTEM

There are five stages about the whole trajectory simulation process of electromagnetic rail gun system [6].

--Firstly, discharging and attacking Stage: Pulsed power is charged by energy storage device and launching equipment prepares to launch. Projectile is pushed into the launching rails; Pulsed power discharges to supply appropriate powerful current, which generates a strong magnetic field between rails.

--Secondly, projectile accelerating stage: The projectile is accelerated along the rails by the great Lorentz force. Due to the drag forces, the projectile and rails generate much heat, and the thermal aggregation causes the characteristic alteration of component material.

--Thirdly, projectile flight stage: The off-rail projectile, affected by aerodynamic force and gravity, flies in atmosphere with high speed.

--Fourthly, attack completion stage: The projectile enters the target area and causes the attacking effect, so that the mission of single shot is accomplished.

--Lastly, launching ending stage: The motion state of the projectile in the barrel is calculated, and analysis of the ballistics performance for rail gun is performed.

III. DEVELOPMENT OF EMRG SIMULATION MODEL

A. Requirement of EMRG Simulation

The functions of the electromagnetic rail gun simulation system include:

Firstly, simulation system should simulate the launch process of the electromagnetic rail gun and the flight process of the projectile.

Secondly, simulation system can manage the various simulation processes including inner ballistics process, exterior ballistics process and whole trajectory to meet many demands.

B. Inner Ballistics Model of EMRG

The ballistics performance of projectile in the barrel is directly governed by effects of various forces on it. The equation expressing the mechanical equilibrium is:

$$(m_p + m_a)\frac{dv}{dt} = F - F_f - F_p \qquad (1)$$

where F_f is the friction drag force between rail and armature and F_p is the drag force on the projectile. F is electromagnetic force accelerating armature and projectile, which is determined by

$$F = \frac{1}{2}L'i^2 \qquad (2)$$

where L' is per unit inductance of the rails and i is the current in the rails.

The friction drag force on the armature can be expressed by [7]

$$F_f = \mu_f\left(F_{n0} + \frac{kS_b}{2A_b}\left(F + F_p\right)\right) \qquad (3)$$

where μ_f is the friction coefficient, F_{n0} is the initial pressure on the rail, S_b and A_b are the perimeter and cross-sectional area of the armature. The drag force on the projectile F_p in (1) is estimated by considering the status of compressed air ahead of projectile [8]

$$F_p = (\frac{\gamma+1}{2})\rho_0[A_b v^2 + A_b xa + (1/2)C_f S_b v^2 x] \qquad (4)$$

where C_f, γ and ρ_0 are the viscous coefficient, specific heat and density of compressed air, v and a are velocity and acceleration of the projectile, x is distance along the rail.

The rail gun is a launch device to transform the electric energy to kinetic energy. In the application, the rails and armature are considered as the load of pulsed power supply system. The equivalent electrical model for the n segments capacitor banks is shown in Figure 2.

The electrical equilibrium equation of the equivalent circuit is expressed as

$$L_n\frac{di_{\text{n}}}{dt} + R_n i_{\text{n}} + \int\frac{1}{C_n}i_{\text{n}}\text{dt}\text{=-}R_{armature}i\text{-}R_{rail}i - \frac{d(L_{rail}i)}{dt} \qquad (5)$$

where $i = i_1 + i_2 + \ldots i_n$ is the total current flowing through the rails, R_n, L_n and C_n are the equivalent resistance, inductance and capacitance of the n pulsed power segment.

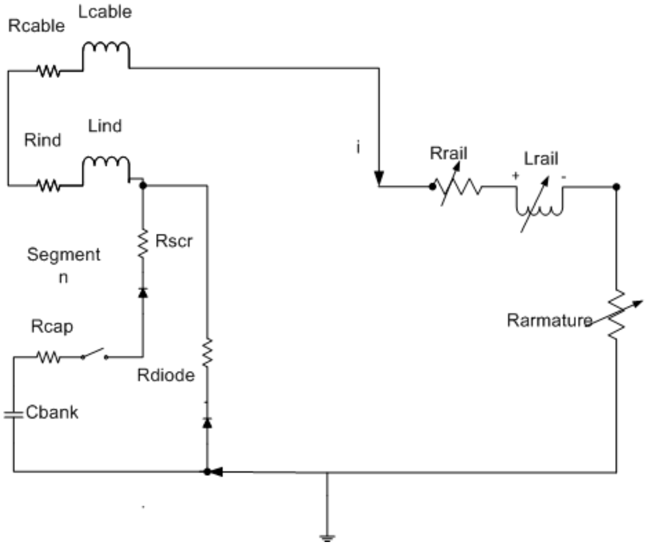


Figure 2. Equivalent electrical model for the n segments capacitor banks

The rail resistance includes heating effect and the current skin effect. The resistance is given by [9]

$$R_{rail} = R_{r0} + \frac{8}{3h}\sqrt{\frac{\pi\mu_r\rho_r}{2t}}x \qquad (6)$$

where μ_r is the permeability, h is the rail height, R_{r0} is the initial resistance of the rail, ρ_r is the instantaneous resistivity , which is determined by [10]

$$\rho_r = \rho_0 + \beta\frac{i}{h} \qquad (7)$$

where β is the current dependent resistivity.

The rail inductance proportional to the distance along the rail x is expressed by

$$L_{rail} = L'x \qquad (8)$$

The armature provides a closed electrical path between the rails. The time varying resistance can be expressed by [11]

$$R_{armature} = \frac{d}{h_a}\sqrt{\frac{\pi\mu_a\rho_a}{2t}} \qquad (9)$$

where d is the rail separation, h_a is the height of armature, μ_a and ρ_a are permeability and resistivity of the armature.

C. Exterior Ballistics Model of EMRG

When the projectile flies to the target, the projectile launched by EMRG is assumed to be a pellet with six degrees of freedom in the air. During the flight, the projectile with high speed is affected by the aerodynamics and gravity. In the study, we make assumptions that the effects of the aerodynamic heating and the earth's rotation on the projectile are ignored. The dynamic and kinematic equations of the projectile in the air are expressed by (10)-(13).

Kinematic equations of the moving gravity center

$$\begin{cases} \dfrac{dx}{dt} = v\cos\psi_2 \cos\theta_1 \\ \dfrac{dy}{dt} = v\cos\psi_2 \sin\theta_1 \\ \dfrac{dz}{dt} = v\sin\psi_2 \end{cases} \tag{10}$$

Dynamic equations of the moving gravity center

$$\begin{cases} m\dfrac{dv}{dt} = -\dfrac{1}{2}\rho SC_x v_r (v-\omega_{//}) + \dfrac{1}{2}\rho SC_y'[\omega_{x\perp}^2 + \omega_z^2 - \\ \omega_{x\perp}(v-\omega_{//})\delta_1 + \omega_z (v-\omega_{//})\delta_2] - mg\sin\theta_1 \cos\psi_2 \\ mv\cos\psi_2 \dfrac{d\theta_1}{dt} = F_p \cos\delta_2 \sin\delta_1 - \dfrac{1}{2}\rho SC_x v_r \omega_{x\perp} + \\ \dfrac{1}{2}\rho SC_y'[v_r^2\delta_1 - (v-\omega_{//})\omega_{x\perp}] - mg\cos\theta_1 \\ mv\dfrac{d\psi_2}{dt} = \dfrac{1}{2}\rho SC_x v_r \omega_z + \dfrac{1}{2}\rho SC_y'[v_r^2\delta_2 + \\ (v-\omega_{//})\omega_z] + mg\sin\theta_1 \sin\psi_2 \end{cases} \tag{11}$$

Dynamic equations rotating the moving gravity center

$$\begin{cases} C\dfrac{d\omega_\varepsilon}{dt} = \dfrac{1}{2}\rho v_r^2 Slm_{xw} - \dfrac{1}{2}\rho v_r Sldm_{xd}'\omega_\xi \\ A\dfrac{d\omega_\eta}{dt} + C\omega_\xi\omega_\varepsilon - A\omega_\xi^2 \tan\varphi_2 = -\dfrac{1}{2}\rho Slm_z' v_r \\ [(v-\omega_{//})\delta_2 + \omega_z] + \dfrac{1}{2}\rho Slm_y' v_r [(v-\omega_{//})\delta_1 - \omega_{x\perp}] \\ A\dfrac{d\omega_\xi}{dt} - C\omega_\varepsilon\omega_\eta + A\omega_\eta\omega_\xi \tan\varphi_2 = \dfrac{1}{2}\rho Slm_z' v_r \\ [(v-\omega_{//})\delta_1 - \omega_{x\perp}] + \dfrac{1}{2}\rho Slm_y' v_r [(v-\omega_{//})\delta_2 + \omega_z] \end{cases} \tag{12}$$

Kinematic equations rotating the moving gravity center

$$\begin{cases} d\varphi_a / dt = \omega_\xi / \cos\varphi_2 \\ d\varphi_2 / dt = -\omega_\eta \\ d\gamma / dt = \omega_\xi - \omega_\xi \tan\varphi_2 \end{cases} \tag{13}$$

Geometric relation equations is given by (14)

$$\begin{cases} \sin\delta_{r2} = \sin\varphi_2 \cos\psi_r - \sin\psi_r \cos\varphi_2 \cos(\varphi_a - \theta_r) \\ \sin\delta_{r1} = \sin(\varphi_a - \theta_r)\cos\varphi_2 / \cos\delta_{r2} \\ \cos\delta_r = \cos(\cos\delta_{r1} \cos\delta_{r2}) \\ \sin\alpha_{ar} = \sin(\varphi_a - \theta_r)\sin\psi_r / \cos\delta_{r2} \end{cases} \tag{14}$$

IV. EMRG SIMULATION AND RESULTS

Simulation experiments are performed by utilizing the described model including inner ballistics model and exterior ballistics model to analyze the ballistics performance. In these experiments, length of rail is 3 meter, mass of projectile is 50 gram, and six capacitor modules are triggered to supply pulsed current. Some simulation results of position and velocity of projectile are shown in Figure 3 and Figure 4. It indicates that the velocity of the projectile is about 2095 m/s and the launch time is 3.3 ms. Slope of the velocity curve is decreased to zero at the final stage in launch process and the shot out velocity achieved the peak value for this experiment.

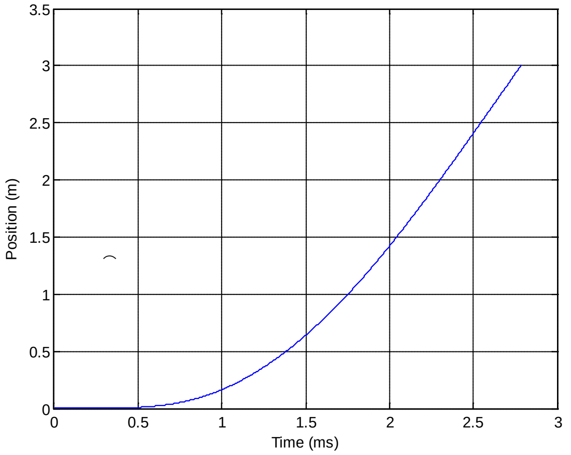


Figure 3. The simulation results of projectile position in the bore

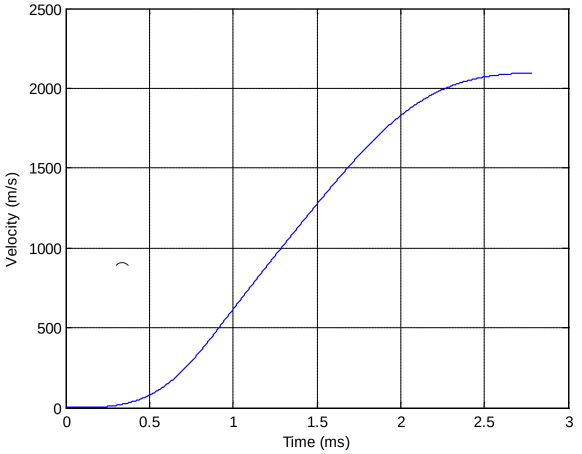


Figure 4. The simulation results of projectile velocity in the bore

The curve of discharging at 6 kv is shown in Figure 5. The performance of pulsed power is 1.1 ms flat-top pulse duration, 0.59 MA peak current and 0.22 MA residual current. The six capacitor modules are switched by thyristor in a certain time sequence to supply flat top current pulse, which makes the rising time of current reduced and the peak current increased greatly.

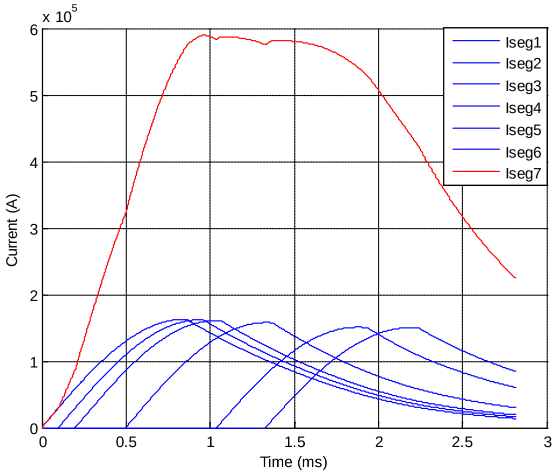


Figure 5. Discharge current of each module and the total current of six modules

The acceleration of the projectile in the bore is shown in Figure 6. It is found that the acceleration curve has the similar shape with that of the total current. The launch process is easily controlled by adjusting the magnitude and shape of the current in the rail.

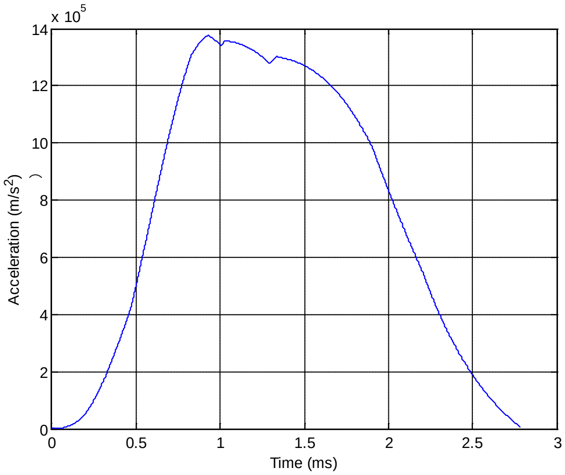


Figure 6. The simulation results of projectile acceleration in the bore

The simulation results of Lorentz force, friction force and air drag force are shown in Figure 7. Comparing these forces, it is obvious that the Lorentz force proportional to the current plays a major effect and friction force is very small and stable during launch. The air drag force is relatively small at the initial stage of acceleration process, but it will increase dramatically with the velocity of the projectile.

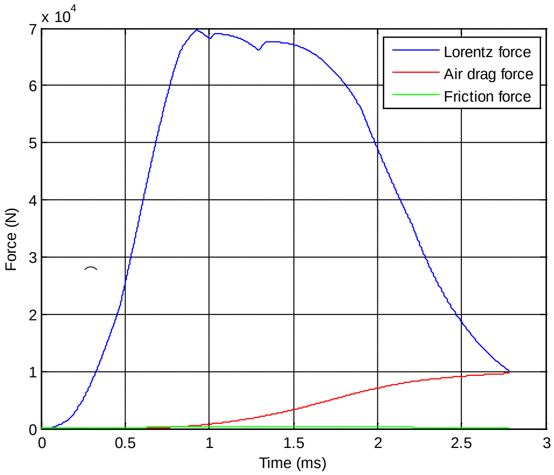


Figure 7. Lorentz force, air drag force and friction force

After the capacitor modules finish discharging, the projectile is shot out the bore with 45° departure angle. The flight velocity of the projectile in the air is shown in Figure 8. The flight velocity decreased greatly in the early stage of flying process because of the effect of the gravity and aerodynamic drag on projectile, but it will slowly decline in the latter stage.

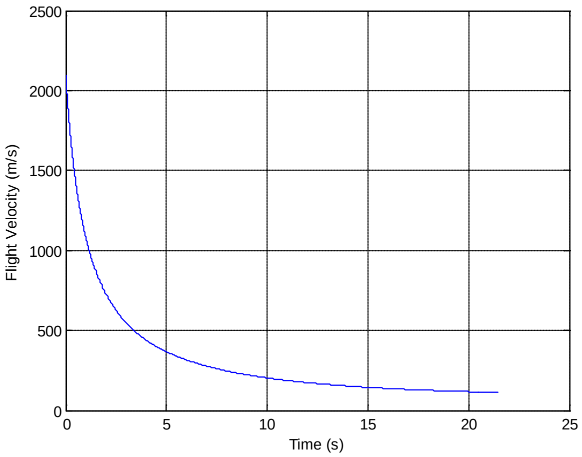


Figure 8. Flight velocity of the projectile in air

The simulation results of projectile range and height in the air are shown in Figure 9 and Figure 10.

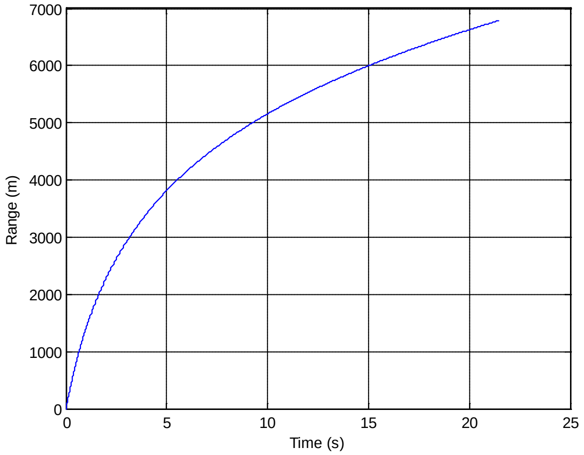


Figure 9. The simulation results of projectile range in air

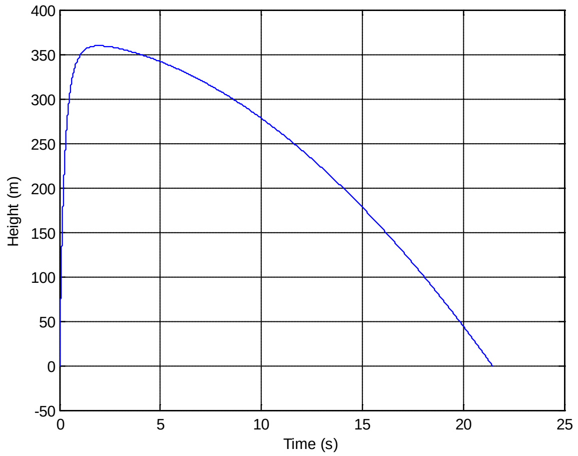


Figure 10. The simulation results of projectile height in air

The range of projectile is 6785 m and the flight height is 365 m during about 23 ms flight time. In the most time of flight the projectile is descend in the air. To improve the exterior ballistics characteristics of the electromagnetic rail gun, decreasing departure angle is an effective way.

V. CONCLUSION AND FUTURE WORK

Some whole trajectory simulations for electromagnetic rail gun are proposed based on the development of whole trajectory simulation models of EMRG. The system can simulate the launching process and the flight process of projectile. The simulation result comparison to the real data will be done in the future work once the data of the prototype or existing system can be obtained.

The application of the simulation system contributes to improve the efficiency of EMRG system designing and lays foundation for the ballistics integration design of EMRG in early design stage.

ACKNOWLEDGMENT

This work was supported in part by the Innovative Team Program of the National Natural Science Foundation of China under Grant No. 61021002 That support is gratefully acknowledged.

REFERENCES

[1] H. D. Fair, “Electromagnetic launch science and technology in the United States enters a new era,” IEEE Trans. Magn., vol. 41, pp. 24-28, 2005.

[2] G. A. Shvetsov, P. G. Rutberg, and A. F. Savvateev, “Overview of EML Research in Russia,” IEEE Trans. Magn., vol. 43, pp. 99-106, 2007.

[3] I. R. McNab, “Pulsed power for electric guns,” IEEE Trans. Magn., vol. 33, pp. 453-460, 1997.

[4] P. Lehmann, “Overview of electric launch activitiesat the French-German research instituteof Saint–Louis,” IEEE Trans. Magn., vol. 39, pp. 24-28, 2003.

[5] http://www.onr.navy.mil/Media-Center/Fact-Sheets/Electromagnetic-Railgun.aspx 11.06.2012.

[6] Yuwei Hu, “Modeling and simulation of electromagnetic rail gun system,” Master’s Thesis of Harbin Institute of Technology, 2007:17-19.

[7] S. Aigner and E. Igenbergs, “Friction and ablation measurement in a round bore railgun,” IEEE Trans on Magn., vol. 25, pp. 33-39, 1989.

[8] G. E. Rolader and J. H. Batteh, “Effect of in-bore gas on railgun performance,” IEEE Trans on Magn., vol. 27, pp. 120-125, 1991.

[9] A. N. Smith, R. L. Ellis and J. S. Bernardes. “Thermal management and resistive rail heating of a large-scale naval electromagnetic launcher,” IEEE Trans on Magn., vol. 41, pp. 235-240, 2005.

[10] F. J. Deadrick, “MAGRAC—A railgun simulation program,” IEEE Trans on Magn., vol. 18, pp. 94-104, 1982.

[11] J. S. Bernardes, M. F. Stumborg and T. E. Jean, “Analysis of a capacitor-based pulsed-power system for driving long-Range electromagnetic guns,” IEEE Trans on Magn., vol. 39, pp. 486-490, 2003.

Modeling Material Heterogeneity by Gaussian Random Fields for the Simulation of Inhomogeneous Mineral Subsoil Machining

Nils Raabe, Anita Monika Thieler, Claus Weihs, Roland Fried
TU Dortmund University
Faculty of Statistics
Dortmund, Germany
raabe@, thieler@, weihs@, fried@ ...statistik.tu-dortmund.de

Christian Rautert, Dirk Biermann
TU Dortmund University
Institute of Machining Technology
Dortmund, Germany
rautert@, biermann@ ...isf.de

***Abstract*—In general, tools used in concrete machining operations are not adapted to the particular machining processes whereas tool wear and production time are the main cost causing factors. A geometrical simulation model describing cutting forces and wear of both diamond and workpiece had been proposed in the past. This model takes the abrasive nature of the machined material into account by modeling the microparts of diamond and workpiece as delaunay tessellations of points randomly distributed within the workpiece and simulating the process iteratively. By fitting the model to a series of real experiments, the general appropriateness of the model had been shown. An implicit assumption of these fittings is that the connected processes are stationary. However, after investigating real process data in the time domain, it turns out that this assumption does not hold. Instead, the forces are obviously affected by material heterogeneity which is not taken into account in the first stage model. To fill this gap, now, an extension of the simulation model is introduced, where the material heterogeneity is modeled and simulated by Gaussian Random Fields.**

***Keywords*-*Machining; Numerical Simulation; Gaussian Random Fields.*

I. INTRODUCTION

Tool wear and material removal rate represent two dominant cost factors in machining processes. To obtain durable tools with increased performance, these factors have to be optimized considering the process conditions. Unlike ductile materials such as steel, aluminum or plastics, material characteristics for mineral substrates like concrete are difficult to determine due to their strongly inhomogeneous components, the dispersion of the aggregates and porosities, the time dependency of the compression strength, etc. [2]. As a result of the brittleness of mineral materials and the corresponding discontinuous chip formation, there are varying engagement conditions of the tool which leads to alternating forces and spontaneous tool wear by diamond fracture.

Despite the manifold of concrete specifications, tools for concrete machining are still more or less standardized tools which are not adapted to the particular machining application. The following analysis is carried out in a subproject of the Collaborative Research Center SFB 823 [10]. In non-percussive cutting of mineral subsoil such as trepanning, diamond impregnated sintered tools dominate the field of machining of concrete due to the excellent mechanical properties of diamonds. These composite materials are fabricated powdermetallurgically [6]. Well-established techniques like vacuum sintering with a preceding cold pressing process or the hot-pressing, which is a very fast manufacturing route, are used for industrial mass production. Due to the premixing of metal powders and synthetic diamond grains, the embedded diamonds are statistically dispersed in the metal matrix. Additionally, the composition and allocation of different hard phases, cement and natural stone grit in the machined concrete are randomly distributed. Because of these facts, the exact knowledge of the machining process is necessary to be able to investigate for appropriate tool design and development.

To obtain a better understanding of these highly complex grinding mechanisms of inhomogeneous materials, which are hard to be described by physical means, statistical methods are used to take into account the effect of diamond grain orientation, the disposition of diamonds in the metal matrix and the stochastic nature of the machining processes of brittle materials. The first step to gain more information about the machining process is the realization of single grain wear tests on different natural stone slabs and cement.

This paper is organized as follows. Section II gives an overview of the setup used for the experiments under study. After that, Section III shortly describes the simulation model the extension of which is shown in Section IV. After the presentation of the results in Section V, the paper closes with the conclusion and a short outlook in Section VI.

II. EXPERIMENTAL SETUP

To gain information about the fundamental correlations between process parameters and workpiece specifications, single grain scratch tests have been accomplished. Within these, isolated diamond grains, brazed on steel pikes have been manufactured (see Fig. 1 and 2) to prevent side effects of the binder phase or forerun diamond scratches as they occur in the grinding segments in real life application. To provide consistent workpiece properties, high strength

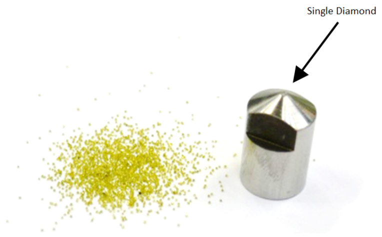


Figure 1. Diamonds and brazed sample.

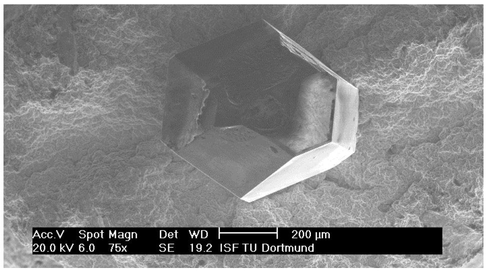

Acc.V Spot Magn Det WD 200 µm
20.0 kV 6.0 75x SE 19.2 ISF TU Dortmund

Figure 2. SEM picture of brazed diamond grain.

concrete specimens of specification DIN 1045-1, C80/90 [3] containing basalt as the only aggregate had been produced. Besides these the two phases, cement binder and basalt, were separately prepared as specimens for an analysis of their material specific influence on forces and wear.

To eliminate further side effects such as hydrodynamic lubrication, interaction of previously removed material and adhesion, the experiments have been carried out without any coolant. The brazed diamond pikes had been attached to a rotating tool holder which in turn had been mounted to the machine (see Fig. 3) to simulate the original process kinematic. Parameters for experimental design were chosen according to common tools and trepanning processes. To guarantee constant depth of cut the rotatory motion of the diamond pike had been superimposed by a constant infeed which generated a helical trajectory. To generate a measureable diamond wear, a certain distance had to be accomplished. Therefore, a total depth of cut of 250 μm had been achieved in every test.

III. Simulation Model

The general aim of the project at hand is the optimization of the machining process w.r.t. production time, forces affecting the workpiece and tool wear. For this aim, knowledge about the relationships between adjustable process parameters, measurable covariates and the outcome is inevitable. From this knowledge, optimal strategies and parameter settings can be derived. As the real machining experiments are very time consuming and expensive it is of primary interest to develop a realistic simulation model. This model then can be used for the derivation and testing of such strategies and settings before verification in real processes.

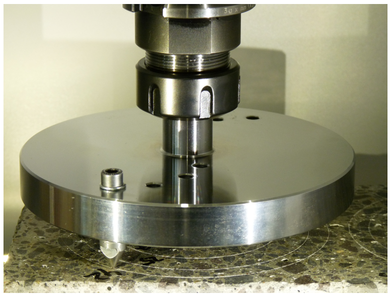


Figure 3. Scratch Test Device on Basalt.

```
compute $(S_k, S_w)$
$S_k^0 \leftarrow S_k R_x R_z R_y + (d_p, h_k, 0) \otimes 1_{n_k}$
for $i = 1 \rightarrow i_{max}$ do
  $S_k^i \leftarrow S_k^{i-1} R_r - (0, a_r, 0) \otimes 1_{n_k}$
  compute intersection volumes $W_s$
  for $j = 1 \rightarrow n_w$ do
    $m_{k;j} \leftarrow \sum_{l:w_{s;lj}>0} w_{k;l} \rho_k$
    compute $(\gamma_w, \gamma_k)$
    $\gamma \leftarrow max(\gamma_w, \gamma_k)$
    $F_{ij} \leftarrow (v_p m_{k;j})/t_d$
    $(F_{n;ij}, F_{r;ij}) \leftarrow F_{ij}(sin\gamma, cos\gamma)$
    if $w_{w;j} \rho_w > \mu_k m_{k;j}$ then
      remove diamond simplices $l : w_{s;lj} > 0$
    else
      reduce heights of diamond
                  simplices $l : w_{s;lj} > 0$ by $\eta_k$
    end if
    remove workpiece simplex $j$
  end for
  $(F_{n;i}, F_{r;i}) \leftarrow (\sum_j F_{n;ij}, \sum_j F_{r;ij})$
end for
```

Figure 4. Pseudocode Representation of Simulation Model. S_k and S_w: vertices of grain and workpiece; R_x, R_z and R_y: random rotation matrices for initial orientation; d_p: diameter of machined hole; h_k: initial grain height; R_r: rotation matrix of tool depending on angle per iteration; a_r: height change per iteration; W_s: matrix of intersection volumes $w_{l,j}$ between l-th grain and j-th workpiece simplex; ρ_k and ρ_w: diamond and workpiece material densities; γ_w and γ_k: angles of contacting workpiece and diamond simplices; v_p: cutting speed in rpm; t_d: time scaling parameter; μ_k: diamond specific threshold factor; η_k: diamond specific flattening factor.

To deal with this task, a simulation model based on Delaunay Tessellations [5] of the workpiece and the diamond was proposed. This approach had been chosen contrarily to the usually chosen discrete model types on regular grids for the simulation of grinding processes (see [1] and [11] for overviews) due to the abrasive nature of the used materials. Fig. 4 shows a pseudocode representation of the proposed simulation model. For full details of the model, see [9].

Beside the extension of the model, our work is focussing

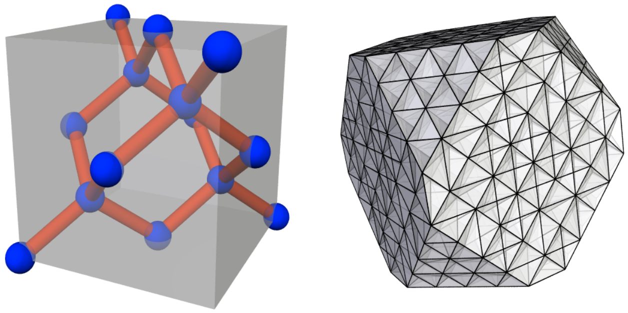


Figure 5. Left: Crystal structure of diamond, right: Simulated diamond grain

about, some slight modifications about how diamond grain and workpiece are modeled had been made. Up to now, the vertices in both grain and workpiece were generated by drawing coordinates from an independent uniform distribution within the material and tool shape. However, to get more realistic results by now the microparts are based on a jittered regular grid for the workpiece. For the tool, the vertices are aligned along the cubic crystal structure of the diamond and also (slightly) jittered (see Fig. 5).

IV. Modeling of Material Heterogeneity

In its latest version, the simulation model (described below) assumed both the diamond and the workpiece to consist of homogeneous material. While this assumption can be seen fulfilled in the case of the diamond it is obviously violated in the case of the workpiece as even relatively homogeneous materials like basalt show a high degree of local differences in hardness. These differences are to be assumed even higher in the case of concrete composites due to air pockets and the concrete additives.

This local heterogeneity in the machined workpiece obviously affects the force signals, as can be illustrated by forces measurements taken during each of the experiments of the basalt series the model had been fitted to in [9]. For better interpretation, Fig. 6 visualizes how the one-dimensional signals are transformed to spatial data. Subfig. a) shows a simulated workpiece with local differences in hardness heterogeneity visualized on a colour scale from blue (low) to red (high). The black arrow line in Subfig. b) shows the course the diamond tip takes during the process as the tool is rotated and shifted towards the workpiece with constant speed. When the tip enters the workpiece it takes force measures from the cylinder highlighted in Subfig. c). By assigning the force measurements to their coordinates on the cylinder and unrolling the resulting cylindral image (Subfig. d)), a two-dimensional image of the signal can be obtained. Note that even though visualizations in the following are partly presented in the plane, computations are always performed in 3D.

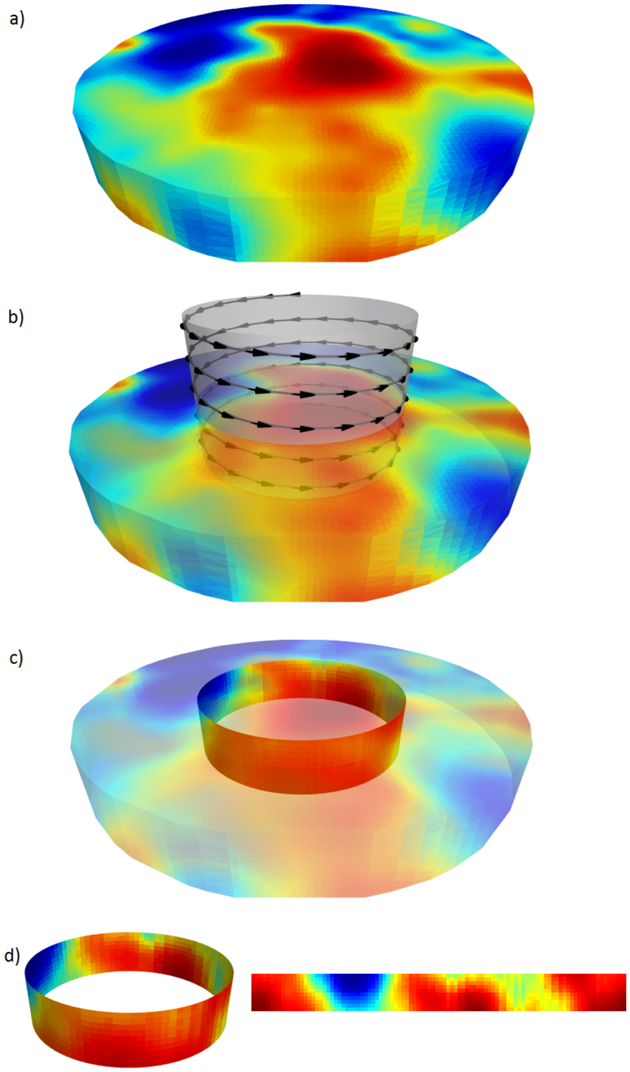


Figure 6. a) Simulated workpiece, b) Course of diamond during process, c) Cylindric bore hole, d) Cylinder cut free (left) and unrolled (right).

Fig. 7 shows exemplary 2D-images for two of the basalt experiments. Obviously, the forces reflect local differences. However, they are disturbed by both random and systematic noise. The random part of the noise is measurement error mainly caused by different chip sizes. The systematic part of the noise appears to be periodic on the one hand and reflects frequencies like gear wheel mesh frequencies which are prominent in the engine spectrum. Another part of the systematic noise is global trend being active during the start of the process while the diamond enters the material.

A. *Robust Statistical Estimation of Local Heterogeneity*

By making the force data accessible for an estimation of the material heterogeneity means the separation of the

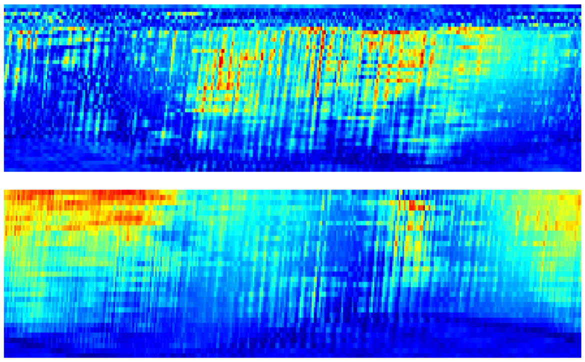


Figure 7. Two-dimensional force images of two basalt machining experiments. Color scale from blue/low force to red/high force.

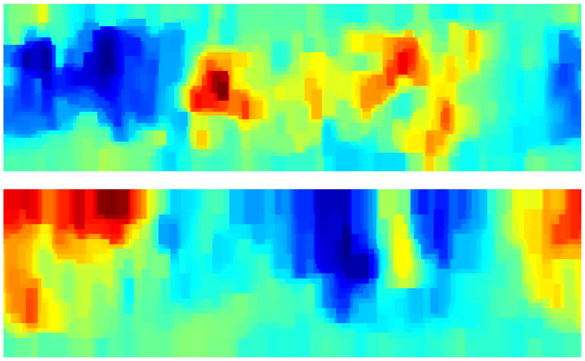


Figure 8. Two-dimensional heterogeneity of two basalt machining experiments. . Color scale from blue/low heterogeneity to red/high heterogeneity.

smooth local heterogeneity and noise. Due to the high variability between the experiments and the severe presence of outliers and possible sudden shifts in the force measurements robust techniques are applied to solve this task.

Mapped back to the time domain, the local heterogeneity can be interpreted as the gradually changing periodic component (seasonality) γ_t in the model

$$Y_t = \mu_t + \gamma_t + u_t + v_t, \quad t = 1, ..., N = l \cdot p, \qquad (1)$$

where the other components building up the specific force signal Y_t are the global trend μ_t, the sum of systematic and random noise u_t and the spiky noise v_t caused by outliers. The period of one revolution is denoted by p, while the number of total observed revolutions is given by l. For the interesting heterogeneity γ_t general smoothness meaning $\gamma_t \approx \gamma_{t-p}$ is assumed while for identifiability the condition $\sum_{i=1}^{p} \gamma_{t+i} = 0, t = 0, p, ..., N-p$ is stated.

Our proposed method for the robust estimation of γ_t is a two-step procedure, the first step of which is the estimation of the trend by $\hat{\mu}$ using running medians of length $2 \cdot \lfloor \frac{p+1}{2} \rfloor + 1$:

$$\hat{\mu} = med\{y_{t-\lfloor \frac{p+1}{2} \rfloor}, ..., y_{t+\lfloor \frac{p+1}{2} \rfloor}\},$$
$$t = \lfloor \frac{p+1}{2} \rfloor + 1, ..., N - \lfloor \frac{p+1}{2} \rfloor.$$

In the second step of our procedure, the moving seasonality γ_t is iteratively estimated by alternating between smoothing the signal in rotational and in feed direction. The smoothing again is obtained by the application of running medians, while the initial heterogeneity estimator is based on the detrended signal meaning

$$\hat{\gamma}_t^{(0)} = med_{j \in J_0}\{y_{t+j} - \hat{\mu}_{t+j}\}, \quad t \in T_0$$
$$\hat{\gamma}_t^{(i)} = med_{j \in J_i}\{\hat{\gamma}_{t+j}^{(i-1)}\}, \quad i \in 1, ..., 2I, t \in T_i$$
$$J_i = \begin{cases} \{-k_h, -(k_h-1), \ldots, (k_h-1), k_h\}, \frac{i+1}{2} \in \mathbb{N}_0 \\ \{-k_v p, -(k_v-1)p, \ldots, (k_v-1)p, k_v p\}, \frac{i}{2} \in \mathbb{N} \end{cases}$$
$$T_i = \begin{cases} \{k_h+1, k_h+2, \ldots, N-k_h\}, \frac{i+1}{2} \in \mathbb{N}_0 \\ \{k_v p+1, k_v p+2, \ldots, N-k_v p\}, \frac{i}{2} \in \mathbb{N} \end{cases}$$
$$i = 1, ..., 2I, \hat{\gamma}_t = \hat{\gamma}_t^{(2I)},$$

where k_h and k_v are the half window widths in rotational and in feed direction and I is the total number of iterations. Values of $\hat{\gamma}_t$ for $t \notin T_i$ are estimated by extrapolation from the closest window.

Within the investigations of the basalt series, it turned out that a common half window width of $k_h = k_v = 7$ gave stable results and that the results converge after the iteration number exceeds the value $I = 16$. Hence, these parameter values had been chosen for the global fitting. Fig. 8 exemplarily shows the 2d-images of $\hat{\gamma}_t$ for the two experiments shown in Fig. 7.

B. Simulation of Local Heterogeneity by Gaussian Random Fields

One aim of our actual work is a realistic simulation of the material heterogeneity within the machining process simulation. A straightforward way for doing so is to simulate the heterogeneity by samples of Gaussian Random Fields, the parameters of which are based on the estimations of γ_t obtained in the way described in the last section.

For this purpose, a joint variogram based on random patches of the in total 73 cylinders derived from each $\hat{\gamma}_t$-series had been computed. The moment estimated variogram based on 50 equally spaced bins from 0 to 130 mm is shown in Fig. 9.

Fig. 9 also shows a theoretical variogram in green which is based on an exponential covariance model (see [12]) with

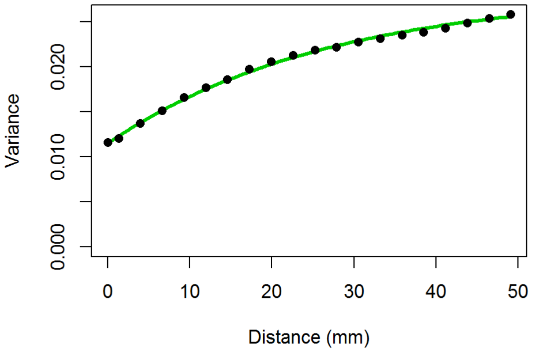


Figure 9. Joint variogram of fitted local heterogeneity. Green: fitted exponential covariance model.

parameters fitted to the empirical variogram by Ordinary Least Squares [7]. These parameters are given by the values 0 for the mean, 0.0168 for the standard deviation, 0.0112 for the nugget and 26.852 for the scale. The corresponding theoretical variogram obviously meets the empirical one very well so it can be used to base the heterogeneity simulations on.

For the simulations between the geometrical initialization of the workpiece and the process simulation, a step is added to the model, where each workpiece simplex gets a heterogeneity value assigned to. To do so, an isotropic Gaussian Random Field [12] is sampled based on the fitted covariance model on a equidistant grid within the workpiece. Then, each workpiece simplex gets the heterogeneity value assigned to that results from interpolating the specific Random Field realization to its center by ordinary Kriging [12].

As the covariance function parameters are based on the additive decomposition of the original signal into trend, seasonality and error, the implementation of the simulated material heterogeneity is simply obtained by adding γ_t to the homogeneously simulated signal y_t.

V. Results

The procedure described in the previous section had been applied to re-simulate a process series with parameter settings defined by the Central Composite Design the basalt experiments were based upon. The simulated output had been compared to the corresponding real data sets and a high degree of accordance between simulated and real data was observed. Fig. 10 shows an exemplary comparison of simulated and measured forces, while Fig. 11 displays a simulated heterogeneous workpiece after machining.

The main remaining differences between the simulated and the real data by now seem to be a periodic noise. As the dominating frequencies of this noise are the same for processes with the same parameter settings, this noise is not caused by material heterogeneity but by the engine and therefore extends the scope of this paper. Obviously, the noise is stationary and by this does not affect the heterogeneity estimation. However, further analysis of this systematic noise type and appropriate model extensions will be made in separate work.

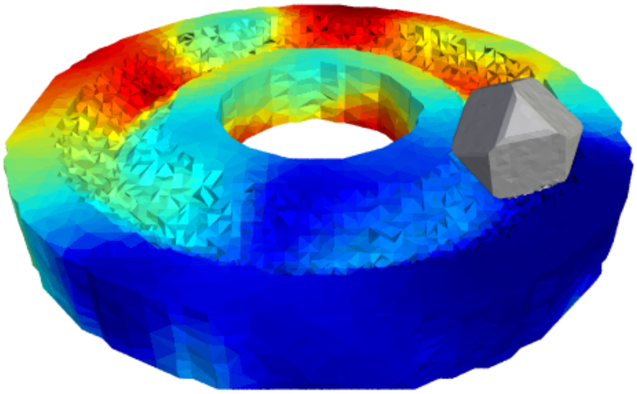


Figure 11. Simulated machined workpiece.

VI. Conclusion and Future Work

In this paper, a major extension of an efficient, flexible and valid model for the simulation of the machining process of inhomogeneous mineral subsoil had been proposed. This extension consists of the shift from static to dynamic modeling where it turned out that material heterogeneity has to be taken explicitly into account. It had been shown that for the integration of this heterogeneity in the case of a comparably homogenous material with presumably continuous heterogeneity structure like basalt the usage of Gaussian Random Fields is appropriate. This result is very feasible since it yields a parsimonious and well identifiable parametrization of heterogeneity.

However, for more complex materials like concrete, overall continuous heterogeneity cannot be assumed as, e.g., aggregates and air pockets cause rapid shifts in heterogeneity. For this reason, heterogeneities of different material phases will have to be modeled separately as, e.g., done in [4] for two-phase materials. For the fit of the correspondingly extended simulation model a procedure for automatic identification of phases is needed, which is actually being developed in the project the work presented here is part of.

Acknowledgment

This work has been supported in part by the Collaborative Research Center Statistical Modelling of Nonlinear Dynamic Processes (SFB 823) of the German Research Foundation

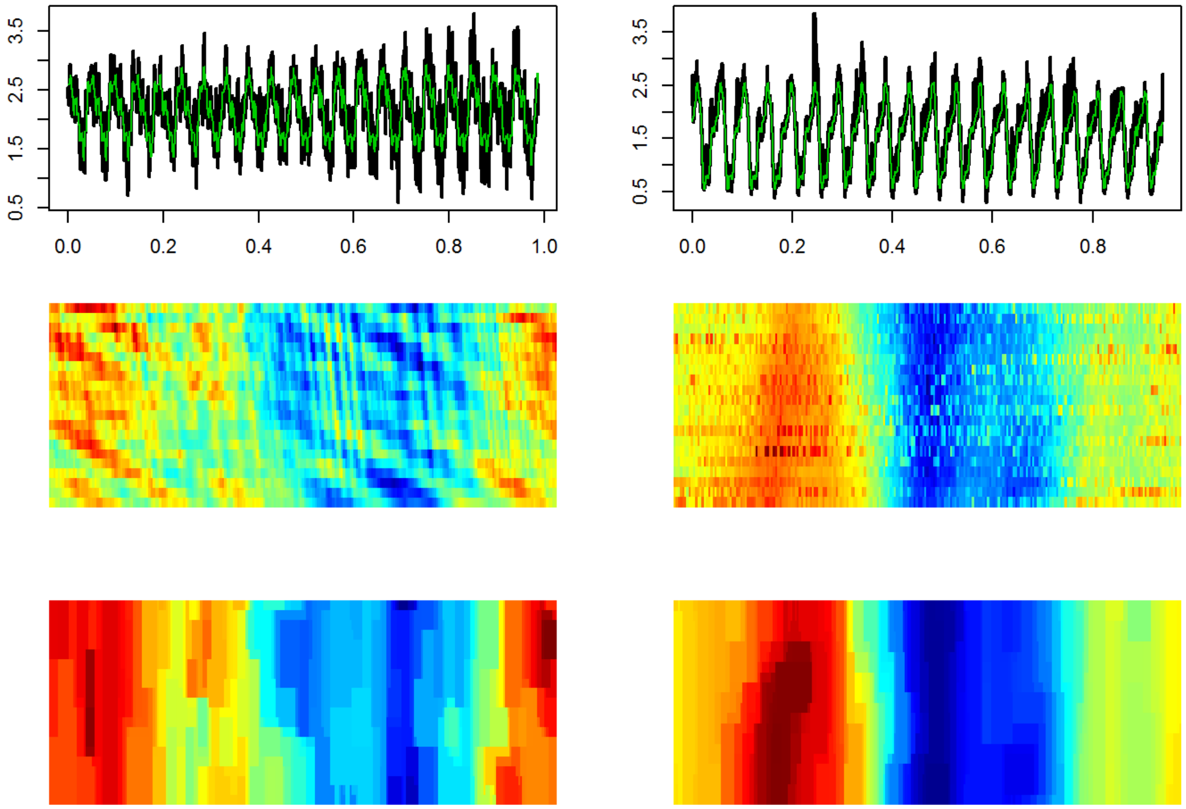


Figure 10. Comparison of real (left) and simulated data (right). Top: signals in time domain with estimated heterogeneities in green. Middle: Signals in 2D-representation. Bottom: estimated heterogeneities in 2D-representation. Note that color scales between signals and heterogeneities are not comparable.

(DFG), within the framework of Project B4, Statistical Process Modelling for Machining of Inhomogeneous Mineral Subsoil.

References

[1] E. Brinksmeier, J. C. Aurich, E. Govekar, C. Heinzel, H.-W. Hoffmeister, J. Peters, R. Rentsch, D. J. Stephenson, E. Uhlmann, K. Weinert, and M. Wittmann "Advances in Modeling and Simulation of Grinding Processes,", in *Annals of the CIRP: Manufacturing Technology*, vol. 55, no. 2, pp. 667-696, 2006.

[2] B. Denkena, D. Boehnke, B. Konopatzki, J.-C. Buhl, S. Rahman, and L. Robben "Sonic analysis in cut-off grinding of concrete", in *Production Engineering*, vol. 2, no. 2, pp. 209218, 2008.

[3] http://www.baunormenlexikon.de/Normen/DIN/DIN%201045-1/1ce039fc-a3e6-4f97-a670-38950b599945.

[4] M. Grigoriu "Random field models for two-phase microstructures" in *Journal of Applied Physics*, vol. 94, no. 6, pp. 61-68, 2003.

[5] O. Hjelle, and D. Boehnke "Triangulations and Applications", New York, NY: Springer, 2006.

[6] J. Konstanty "Powder metallurgy Diamond Tools", Amsterdam: Elsevier, 2006.

[7] D. C. Montgomery, E. A. Peck, and G G. Vining "Introduction to Linear Regression Analysis, 3rd Edition [Hardcover]", New York, NY: Wiley, 2001.

[8] L. S. Pan "Diamond: Electronic Properties and Applications", Norwell, MA: Kluwer, 1995.

[9] N. Raabe, C. Rautert, M. Ferreira, and C. Weihs "Geometrical Process Modeling of Concrete Machining Based on Delaunay Tessellations", in *Proceedings of The World Congress on Engineering and Computer Science 2011*, vol. 2, 2011, pp. 991-996.

[10] http://www.statistik.tu-dortmund.de/sfb823-project_b4.html.

[11] H. K. Tönshoff, J. Peters, I. Inasaki, and T. Paul "Modelling and Simulation of Grinding Processes," in *Annals of the CIRP: Manufacturing Technology*, vol. 41, no. 2, pp. 677-688, 1992.

[12] H. Wackernagel "Multivariate Geostatistics", 3nd ed., Berlin: Springer , 2003.

Simulation of the Deflection of Thin Plates under the Action of Random Loads

Vitaliy Lukinov
Russian Academy of Sciences
ICM&MG SB RAS
Novosibirsk, Russia
Email: Vitaliy.Lukinov@gmail.com

***Abstract*—In this work, new results are obtained using constructed probabilistic representation of the first boundary value problem for polyharmonic equation. It is shown that corresponding solution is presented by the parametric derivative of a solution to the specially constructed Dirichlet problem for Helmholtz equation. On this base, new algorithms of 'random walk by spheres' for solving biharmonic equations are derived.**

***Index Terms*—plate bending; Monte Carlo methods; biharmonic equations; 'random walks by spheres'**

I. Introduction

The deflection of thin plates under the action of loads is satisfied an biharmonic equation with Dirichlet, Neumann, or mixed boundary conditions [1]. Despite the slow rate of convergence of statistical methods for low-dimensional spaces, in comparison with classical numerical methods, their use is advantageous in finding a solution to a small area or for calculating the statistical characteristics of the solutions with random right-hand sides. We can distinguish several approaches of Monte Carlo methods for solving above mentioned problems:

- Approaches based on probabilistic representation of the solution [3], [7], [5]
- Random walk by subdomains methods [8], [9], [10], [13] ("by spheres" is most known)
- Random walk on boundary methods [13]
- SVD-based approaches [14]

Let us consider the pros and cons of each approach. The methods based on probabilistic representation of the solution are often used to find the asymptotic properties of solutions. These methods are difficult to construct numerical algorithms directly and estimates of the simplified approach are used. This methods are more time-consuming in comparison with others. More economical methods are walk by subdomains and walk on boundary based on the reduction of the original equation to a special integral equation with generalized kernel. Walk on boundary methods are derived for a more restricted range of problems, but can solve problems with complex geometry boundaries. New SVD approach allows to construct the most efficient statistical methods for finding solutions of linear equations approximated the original problem.

In this paper, we consider the biharmonic equation with random inputs functional parameters. The walk by shperes vector estimates of covariance for the solutions were constructed in [6], the corresponding walk on boundary vector estimates were suggested in [13], SVD approach was presented in [14].

Here, a new scalar walk by spheres estimates of covariance for the solution were constructed by an parametric differentiation of well known estimates of solution to special constructed problems. First, this approach was proposed by G. Mikhailov [12]. Besides, Mikhailov and Tolstolytkin [6] proposed scalar estimates had been fully investigated: the finiteness of variances has been proved, absolute errors have been evaluated, laboriousness have been estimated, the problem of optimal choice of method parameters to achieve a given error level have been solved. Compared to [13][14] the offered method can solve a problems with a random spectral parameter. It seems that the approach by Sabelfeld and Mozartova [14] is less time consuming, but a special comparison of methods was not carried out.

We obtain estimates for the Dirichlet BVP and some special Neumann BVP. Further investigation is aimed at building a cost-effective methods for mixed and Neumann BVP.

This work is mostly theoretical. However, the proposed estimates is easy to extend to the real problems with the own geometry of the boundary.

A brief structure of the paper is presented below. Precise mathematical formulation of problems for theory of fluctuations of elastic systems and some auxiliary equations are presented in Section II. General theoretical results for polyharmonic equation, obtained with Mikhailov [4][5], are presented in Section III. New results and model calculations are considered in Section IV.

II. Boundary Value Problems

A. Helmholtz equation

Let us consider the Dirichlet problem for the Helmholtz equation in a domain $D \subset R^3$ with boundary Γ:

$$(\Delta + c)u = -g, \quad u|_\Gamma = \varphi. \tag{1}$$

Let us assume that the following conditions hold. The function g satisfied Holder condition [2] in $\overline{D}$, D is a bounded open set in R^3 with a regular boundary Γ, the function φ is continuous on Γ, $c < c^*$, where $-c^*$ is the first eigen value of Laplace

This work was supported by the Russian Foundation for Basic Research (grants N09-01-00639-a and N11-01-00252-a) and Novosibirsk State University

operator defined on the domain D. These conditions provide the existence and uniqueness of the solution to problem (1), the existence its probabilistic and integral representations in terms of the spherical Green's function [2]. We suppose that above conditions are fulfilled and after change of all parametric functions by their modules.

B. Elastic BVP

In a domain $\mathrm{D} \in \mathrm{R}^2$ with boundary Γ bending of thin elastic plate satisfies the biharmonic equation [1]

$$\Delta^2 u = f(x,y)/K. \tag{2}$$

In case a plate is lying on an elastic foundation, we have the following equation [1]

$$\Delta^2 u + cu = f(x,y)/K. \tag{3}$$

Here, $u(x,y)$ is normal flexure of a plate at the point (x,y); $f(x,y)$ is a strength of normal charge; $K = Eh^3/12(1-\sigma^2)$, where E is the elastic modulus; σ is the Poison constant for the stuff of the plate; $2h$ is plate thickness. Let us consider the following frequently occurring boundary conditions

- the edge of the plate is simply supported: $u|_\Gamma = 0$, $\Delta u - \frac{1-\sigma}{\rho}\frac{\partial u}{\partial n}\Big|_\Gamma = 0$;
- the edge of the plate is rigid: $u|_\Gamma = 0$, $\frac{\partial u}{\partial n}\Big|_\Gamma = 0$;
- the edge of the plate is elastically supported: $u|_\Gamma = 0$, $\Delta u + \left(\frac{1-\sigma}{\rho} + k_0\right)\frac{\partial u}{\partial n}\Big|_\Gamma = 0$.

Here, n is the external normal to the boundary Γ of the plate; ρ is curvature radius of Γ; k_0 is a value related to a rigidity of the edge fixity.

C. Metaharmonic BVP

Let us consider the general problem:

$$\begin{cases} (\Delta + c)^{p+1} u = -g, \\ (\Delta + c)^k u|_\Gamma = \varphi_k, \quad k = 0, \ldots, p. \end{cases} \tag{4}$$

In this work, the following results will be used [5].

Theorem 1. *Let conditions of the part A are satisfied then the p-th parametric derivative of solution u to the problem (1) with a functional parameters*

$$\varphi = \sum_{k=0}^{p} \frac{(-1)^k \lambda^{p-k}}{p!}\varphi_k, \quad g_1 = \frac{(-1)^p}{p!} g \tag{5}$$

is the solution to the problem (4).

III. Algorithms of 'random walks by spheres'

A. General algorithm

Further considered estimators of the Monte Carlo method are associated with a 'random walks by spheres' in the domain D [8]. For simplicity, we designate: $\overline{D}$ is a closure of domain D; $d(P)$ is a distance from the point P to the boundary Γ; $\Gamma_\varepsilon = \{P \in \overline{D} : d(P) < \varepsilon\}$ is the ε-neighborhood of the boundary; $S(P) = \{Q \in \overline{D} : |Q - P| = d(P)\}$ is a sphere of radius $d(P)$ with its center at the point P lying in $\overline{D}$. In the 'random walk by spheres' we chose the successive P_{k+1} uniformly on the sphere $S(P_k)$; the walk is terminated if the point P_{k+1} occurs in Γ_ε. Let $N = \min\{\mathrm{m} :\ r_m \in \Gamma_\varepsilon\}$.

It is well known [12] that solution to the problem (1) satisfies the following equation $u(r_0) = \mathsf{E}\,\eta_\varepsilon$ in R^n, where

$$\begin{aligned} \eta_\varepsilon = \sum_{i=0}^{N}\left[\prod_{j=0}^{i-1} s(c,d_j)\right] & \int\limits_{D(r_i)} G(\rho;c,d_i)g(\rho)d\rho \\ & + \left[\prod\nolimits_{j=0}^{N-1} s(c,d_j)\right] u(r_N). \end{aligned} \tag{6}$$

Here, $d_j = d(r_j)$, $D(r_i)$ is a ball of radius d_i with its center at the point r_i,

$$s(c,d) = \frac{(d\sqrt{c}/2)^{(n-2)/2}}{\Gamma(n/2)J_{(n-2)/2}(d\sqrt{c})}, \tag{7}$$

$G(\rho;c,d)$ is a spherical Green's function, $J(\cdot)$ is a Bessel function, $\Gamma(\cdot)$ is a Gamma function.

Therefore, using Theorem 1, we have following assertion (all derivatives are considered at the point $c = c_0$) [4].

Theorem 2. *Under the conditions of Theorem 1 the following representation holds true for the solution to the problem (3)* $u = \mathsf{E}(\frac{\partial \eta_{1,\varepsilon}}{\partial c^p}) = \mathsf{E}(\eta_{1,\varepsilon}^{(p)}) \quad \forall p \geq 0$, *where* $\eta_{1,\varepsilon}$ *is derived from* η_ε *by the substitute*

$$g \longrightarrow g_1 = \frac{(-1)^p}{p!} g,$$

$$u(r_N) \longrightarrow \varphi(r_N, c) = \sum_{k=0}^{p} \frac{(-1)^k (c - c_0)^{p-k}}{p!} u_k(r_N).$$

Here, u_k *is a solution to the problem (4) with* $\varphi \equiv \varphi_k$, $k = 0, \ldots, p$, *function g is equal to zero for* $k = 0, \ldots, p-1$.

Theorem 3. *If* $c < c^*/2$ *then* $\mathsf{D}(\eta_{1,\varepsilon}^{(p)}) < C_p < +\infty \quad \forall p \geq 0$

B. Practical estimators

Let us consider now practically realizing estimate $\tilde{\eta}_{1,\varepsilon}^{(p)}$ which is obtained after substitute of variables $u_k(r_N)$ to $\varphi_k(P)$, where P is a nearest to r_N point of boundary. From (6), we obtain that $\tilde{\eta}_{1,\varepsilon}^{(p)} = \sum\limits_{i=1}^{N} [Q_i(c) \int\limits_{D(r_i)} G(\rho;c,d_i)g_1(\rho)d\rho]^{(p)} + [Q_N(c)\varphi(P,c)]^{(p)}$. The following theorems hold true [4].

Theorem 4. *If* $c < c^*$ *and first spatial derivatives of the function* $\{u_k^{(i)}\}$, $i = 1, \ldots, p+1$, *are uniformly bounded in* $\bar{\mathrm{D}}$ *then* $|u(r) - \mathsf{E}\tilde{\eta}_{1,\varepsilon}^{(p)}| \leq C_p\varepsilon, \quad r \in D, \varepsilon > 0$.

Theorem 5. *Under conditions of Theorem 4, for* $c < c^*/2$ *it holds, that*

$$\mathsf{D}\tilde{\eta}_{1,\varepsilon}^{(p)} < C_p < +\infty, \quad \forall \varepsilon > 0.$$

In the expression of $\tilde{\eta}_{1,\varepsilon}^{(p)}$, we may to estimate the integrals by one random point as follows

$$\begin{aligned} & \int\limits_{D(r)} G(\rho;c,d)g(\rho)d\rho = \\ & = \tfrac{d^2}{2n} \int\limits_{D(r)} p_0(r,p)[G(\rho;c,d)/G(\rho;0,d)]g(\rho)d\rho = \\ & = \tfrac{d^2}{2n} \mathsf{E}\left\{ \tfrac{G(\rho;c,d)}{G(\rho;0,d)} g(\rho) \right\}, \end{aligned} \tag{8}$$

where ρ is a random point in $D(r)$, its distribution function is equal to (under $n > 2$)

$$\begin{aligned} p_0(r,p) &= 2nd^{-2}G(\rho;0,d) = \\ &= \tfrac{2n}{(n-2)d^2\omega_n}\left(\tfrac{1}{(\rho-r)^{n-2}} - \tfrac{1}{d^{n-2}}\right), \quad |\rho - r| \le d \end{aligned} \tag{9}$$

It is clearly that $\int p_0(r,\rho)d\rho = 1$. Under n=2, we obtain

$$p_0(r,p) = 4d^{-2}G(\rho;0,d) = \frac{2}{\pi d^2}\ln\frac{d}{|\rho - r|}, |\rho - r| \le d$$

Since under various values c the Green's functions have at the point $\rho = 0$ a same order of poles the ratio of the function G is bounded in (8) Then

$$\begin{aligned} \tilde{\tilde{\eta}}^{(p)}_{1,\varepsilon} &= \sum_{i=0}^{N}\left\{\left[\prod_{j=0}^{i-1} s(c,d_j)\right] g_1(\rho_i)\tfrac{d_i^2 G(\rho;c,d_i)}{2nG(\rho;0,d_i)}\right\}^{(p)} \\ &+ \left\{\left[\prod_{j=0}^{N-1} s(c,d_j)\right]\varphi(r_N,c)\right\}^{(p)}, \end{aligned} \tag{10}$$

besides $\mathsf{E}\tilde{\tilde{\eta}}^{(p)}_{1,\varepsilon} = \mathsf{E}\tilde{\eta}^{(p)}_{1,\varepsilon}$. It is clear that proof of the Theorem 6 still is valid after substitution of $\tilde{\eta}$ by $\tilde{\tilde{\eta}}$, i.e. $\mathsf{D}\tilde{\tilde{\eta}}^{(p)}_{1,\varepsilon} < C_p < +\infty, \quad \forall\varepsilon > 0$.

IV. Calculations

A. Biharmonic equation solving

Consider the first boundary value problem for the inhomogeneous biharmonic equation

$$\Delta^2 u = -g, \quad u|_\Gamma = \varphi_0, \quad \Delta u|_\Gamma = \varphi_1. \tag{11}$$

in a domain $D \subset R^n$.

Under $n = 3$, the corresponding estimators $\tilde{\tilde{\eta}}^{(1)}_{1,\varepsilon}$ has the following form

$$\begin{aligned} \tilde{\tilde{\eta}}^{(1)}_{1,\varepsilon} &= \tfrac{1}{36}\sum_{i=0}^{N}\left[-\sum_{j=0}^{i} d_j^2 + (d_i - \nu_i)^2\right] d_i^2 g(\rho_i) \\ &- \tfrac{1}{6}\left(\sum_{j=0}^{N-1} d_j^2\right)\varphi_1(r_N) + \varphi_0(r_N). \end{aligned} \tag{12}$$

The random variable ν_i distributed in interval $(0, d_i)$ with probability density $6x(1 - x/d_i)d_i^{-2}$ and isotropic unit vector ω_i are simulated by well known formulas [11].

Under $n = 2$, we obtain that the estimator to the solution to problem(14) has form

$$\begin{aligned} \tilde{\tilde{\eta}}^{(1)}_{1,\varepsilon} &= \tfrac{1}{16}\sum_{i=0}^{N}\left[-\sum_{j=0}^{i-1} d_j^2 - \tfrac{d_i^2 - \nu_i^2 - \nu_i^2\ln(d/\nu_i)}{\ln(d/\nu_i)}\right] d_i^2 \times \\ &\times g(\nu_i,\omega_i) - \tfrac{1}{4}\left(\sum_{j=0}^{N-1} d_j^2\right)\varphi_1(r_N) + \varphi_0(r_N) = \\ &= \sum_{i=0}^{N} Q_i g(\rho_i) + \widehat{Q}_N\varphi_1(r_N) + \varphi_0(r_N), \end{aligned} \tag{13}$$

where ω_i is a isotropic unit vector, ν_i/d_i is a random variable is distributed in interval $(0, 1)$ with a density $-4x\ln x$.

In case $g \equiv 0$, the representation (12) may to get from known estimate [13].

1) Numerical results:

Let us consider the first boundary value problem for the inhomogeneous biharmonic equation

$$\Delta^2 u = 9\exp(x)\exp(y)\exp(z),$$

$$u|_\Gamma = \exp(x)\exp(y)\exp(z), \Delta u|_\Gamma = 3\exp(x)\exp(y)\exp(z).$$

in the unit cube $D = [0,1] \times [0,1] \times [0,1] \subset R^3$. The solution to problem is $u = \exp(x)\exp(y)\exp(z)$. The solution to problem is calculated numerically by formula (12). The numerical results are given in the table 1.

TABLE I
CALCULATIONS FOR THREE-DIMENSIONAL BIHARMONIC EQUATION

ε	$S \cdot 10^{-4}$	$u(r)$	$\widetilde{u}(r)$	$\lvert u(r) - \widetilde{u}(r)\rvert$ $\pm\sqrt{\frac{\sigma^2}{N}}$
10^{-2}	1	4.48169	4.5120	0.030 ± 0.025
10^{-2}	4	4.48169	4.4875	0.006 ± 0.012
10^{-2}	16	4.48169	4.4882	0.0066 ± 0.0063
10^{-3}	16	4.48169	4.4837	0.002 ± 0.063
10^{-3}	256	4.48169	4.4815	0.0001 ± 0.0016

In the Table 1, we assume: $r = (0.5; 0.5; 0.5)$ are the coordinates of the point, S is the number of the modelling trajectories, $\widetilde{u}(r)$ is the numerical solution, σ^2 is the variance of the random estimate $u(r) - \widetilde{u}(r)$.

B. Metaharmonic equation solving

Let us consider the following problem

$$\Delta^2 u + cu = -g, \quad u\Big|_\Gamma = \varphi_0, \quad \Delta u\Big|_\Gamma = \varphi_1. \tag{14}$$

in a domain $D \subset R^2$.

Suppose c is a random variable such that

$$\mathsf{E}c = 0, \quad \mathbf{D}c \ll 1, \quad c < c^*,$$

g is a random field, φ_0 and φ_1 are random functions.

The aim of this subsection is to estimate covariance function $cov(r_1, r_2) = \mathsf{E}u(r_1)u(r_2)$. Using series expansion of $u(r,c)$ at the point $c = 0$ we change

$$\begin{aligned} u(r,c,g,\varphi_0,\varphi_1) &\approx u(r,0,g,\varphi_0,\varphi_1) \\ &+ cu^{(1)}(r,0,g,\varphi_0,\varphi_1). \end{aligned} \tag{15}$$

We may assume that corresponding error δ is equal to

$$\frac{1}{2}u^{(2)}(r,0,g,\varphi_0,\varphi_1)c^2.$$

Then

$$\begin{aligned} &cov(u(r_1,c,g,\varphi_0,\varphi_1), u(r_2,c,g,\varphi_0,\varphi_1)) \approx \\ &\approx \mathsf{E}u(r_1,0,g,\varphi_0,\varphi_1), u(r_2,0,g,\varphi_0,\varphi_1) + \\ &+\mathsf{E}\left[u^{(1)}(r_1,0,g,\varphi_0,\varphi_1)u^{(1)}(r_2,0,g,\varphi_0,\varphi_1)\right]\mathbf{D}c, \\ &2\delta = \mathsf{E}[u^{(2)}(r,0,g,\varphi_0,\varphi_1)u(r,0,g,\varphi_0,\varphi_1)]\mathbf{D}c+ \\ &+\mathsf{E}[u^{(2)}(r,0,g,\varphi_0,\varphi_1)u^{(1)}(r,0,g,\varphi_0,\varphi_1)]\mathsf{E}[c^3] \\ &+\mathsf{E}[u^{(2)}(r,0,g,\varphi_0,\varphi_1)u^{(2)}(r,0,g,\varphi_0,\varphi_1)]\mathsf{E}[c^4] \end{aligned}$$

ISBN: 978-1-61208-234-9

The parametric derivative $u^{(1)}(r,0,g,\varphi_0,\varphi_1)$ is a solution to the following problem

$$\left\{ \begin{array}{l} \Delta^4 u = -g, \quad u\Big|_\Gamma = 0, \quad \Delta u\Big|_\Gamma = 0, \\ \Delta^2 u\Big|_\Gamma = -\varphi_0, \quad \Delta^3 u\Big|_\Gamma = -\varphi_1. \end{array} \right. \tag{16}$$

The parametric derivative $u^{(2)}(r,0,g,\varphi_0,\varphi_1)$ is a solution to the following problem

$$\left\{ \begin{array}{l} \triangle^6 u = g, \quad \triangle^k u\Big|_\Gamma = 0, \quad k = 0,\dots,3, \\ \triangle^4 u\Big|_\Gamma = \varphi_0, \quad \triangle^5 u\Big|_\Gamma = \varphi_1 \end{array} \right. \tag{17}$$

The corresponding estimates of the solutions to the problems (16), (17) has form

$$\tilde{\tilde{\eta}}^{(3)}_{1,\varepsilon} = \sum_{i=0}^{N} \left\{ \sum_{k=0}^{3} \mathrm{C}_3^{\mathrm{k}} S_i^{(3-k)}(0) \frac{G^{(k)}(\rho;0,d_i)}{G(\rho;0,d_i)} \right\} \frac{[-d_i^2 g(\rho_i)]}{24} +$$

$$+ S_N^{(3)}(0)\frac{\varphi_1(r_N)}{6} - S_N^{(2)}(0)\frac{\varphi_0(r_N)}{2},$$

$$\tilde{\tilde{\eta}}^{(5)}_{1,\varepsilon} = \left\{ \sum_{i=0}^{N} \sum_{k=0}^{5} \mathrm{C}_5^{\mathrm{k}} S_i^{(5-k)}(0) \frac{G^{(k)}(\rho;0,d_i)}{G(\rho;0,d_i)} \right\} \frac{d_i^2 g(\rho_i)}{480} -$$

$$- S_N^{(5)}(0)\frac{\varphi_1(r_N)}{120} + S_N^{(4)}(0)\frac{\varphi_0(r_N)}{24},$$

where $S_i(c) = \prod_{j=0}^{i-1} s(c,d_j)$.

1) Numerical results: Here we consider the following problem

$$\Delta^2 u + cu = g, \quad u\Big|_\Gamma = 0, \quad \Delta u\Big|_\Gamma = 0,$$

in the $\mathsf{D} = \{x_1, x_2 : 0 \le x_1, x_2 \le 1\}$.

Suppose c is uniformly distributed in the $(-1/2;1/2)$, g is a homogeneous, isotropic Gaussian field with the spectral density

$$\rho(\lambda) = \frac{1}{2\pi\alpha^2}(1 + |\lambda|^2/\alpha^2)^{-3/2}, \quad \mathsf{E}g = 0,$$

Corresponding covariance function of g is equal to $e^{-\alpha|x|}$, where $|x| = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2}$. In the Table 2, we assume: $v(r,r')$ is a covariance function of solution at the point $r = (0.5;0.5)$ and $r' = (0.5+\delta;0.5)$, σ_v^2 is the variance of the random estimate for v, S is the number of the modelling trajectories, $\triangle$ is the error of approximation, $\sigma^2_\triangle$ is the variance of the random estimate for $\triangle$.

TABLE II
CALCULATIONS FOR THE COVARIANCE ESTIMATOR

δ	ε	S	$(v \pm \sqrt{\frac{\sigma_v^2}{N}})*$ $*10^{-5}$	$(\triangle \pm \sqrt{\frac{\sigma_\triangle^2}{N}})*$ $*10^{-11}$
0.0	10^{-2}	10^3	1.078±0.041	1.08±0.12
0.0	10^{-2}	10^5	1.085 ±0.005	1.26±0.03
0.1	10^{-2}	10^3	1.026±0.071	1.26±0.03
0.1	10^{-4}	10^5	1.010±0.007	1.40±0.00
0.2	10^{-2}	10^5	0.885 ±0.006	1.04±0.03
0.3	10^{-2}	10^5	0.654 ±0.006	0.81±0.04
0.4	10^{-2}	10^5	0.343 ±0.004	0.41±0.18

V. Conclusion and future work

In this paper the parametric differentiation approach has been considered as an efficient method for constructing scalar 'random walk by spheres' estimates. This method is based on parameter differentiation standard estimates of the solution to the special constructed boundary value problem. Using this approach and partial averaging method, 'random walk by spheres' estimates of covariance function were obtained for the Dirichlet problems of the biharmonic equation. For testing efficiency of this method, two examples for different types of equations have been considered. The constructed algorithms are particularly useful for estimating the covariance function in the local domain. Another improvement is the need to store only one trajectory of a random walk.

The developed algorithms find practical application in the theory of elasticity, discussed in Section 2. Problems with random functional parameters are suitable in the study of vibrations of plates under the action of random forces such as the wind or earthquakes.

Future work will focus on the development of similar algorithms for boundary value problems of the second and third type. Additionally, it is supposed to develop 'random walk by spheres' algorithms for biharmonic equations where spectral parameter is a random value with high dispersion and non-zero expectation.

References

[1] Bolotin V.V. Random fluctuations of elastic systems. (Moscow.: Nauka, 1979).

[2] Gilbarg, D., Trudinger, Neil. *Elliptic Partial Differential Equations of Second Order.* (New York: Springer, 1983).

[3] Ventzel A.D. *Course of the random walk's procees* . – Moskow.: Nauka, 1975.

[4] Lukinov V.L., Mikhailov G.A.*The probabilistic representation and Monte Carlo methods for the first boundary value problem for a polyharmonic equation.Rus. J. Num. Anal. and Math. Modell. 2004, v. 19, N.5, p. 434-449* .

[5] Mikhailov G.A., Lukinov V.L. *The probabilistic representation and Monte Carlo methods for solving polyharmonic equations.* // Dokl. Akad. Nauk – 2003. – V. 390, No 6. – P. 1–3.

[6] Mikhailov G.A., Tolstolytkin D.V. *A new Monte Carlo method for calculating the covariance function of the solutions of the general biharmonic equation.* // Dokl. Akad. Nauk – 1994. – V. 338, No 5. – P. 601–603.

[7] Ermakov S.M., Nekrutkin V.V., Sipin A.S. *A random procees for solving a classic equations of mathematical physics.* – Moskow.: Nauka, 1984.

[8] Muller M.E. *Some continuous Monte Carlo methods for the Dirichlet problem* // Ann. Math. Stat. – 1956. – Vol. 27, No 3. – P. 569–589.

[9] Elepov B.S., Mikhailov G.A. *About solution to the Dirihlet problem for equation by constructed 'walk by spheres'.* // J. Num. Math. and Math. Physics. – 1969. – No 3. – P. 647–654.

[10] Motoo M. *Some evalutions for continuous Monte Carlo method by using Brownian hitting process.* // Ann. Math. Stat. – 1959. – Vol. 11. – P. 49–54.

[11] Ermakov S.M., Mikhailov G.A. *Statistical simulation.* – Moskow.: Nauka, 1982.

[12] Mikhailow G.A. *Parametric Estimates by the Monte Carlo method* - Utrecht, The Nederlands,1999.

[13] Sabelfeld K.K. *Monte Carlo Methods for boundary problems.* – Moskow.: Nauka, 1989[Engl. transl.: Springer–Verlag, 1991].

[14] Sabelfeld K.K., Mozartova N. *Sparsified Randomization Algorithms for low rank approximations and applications to integral equations and inhomogeneous random field simulation.* – Monte Carlo Methods Appl. Vol. 15 No. 3 (2009), pp. 116.

Simulation of an Order Picking System in a Pharmaceutical Warehouse

João Pedro Jorge[1], Zafeiris Kokkinogenis[3,4], Rosaldo J. F. Rossetti[2,3], Manuel A. P. Marques[1]

[1] Department of Industrial Engineering and Management, Faculty of Engineering, University of Porto, Portugal
[2] Department of Informatics Engineering, Faculty of Engineering, University of Porto, Portugal
[3] Artificial Intelligence and Computer Science Laboratory (LIACC), Faculty of Engineering, University of Porto, Portugal
[4] Institute of Mechanical Engineering (IDMEC), Faculty of Engineering, University of Porto, Portugal
{deg11008, pro08017, rossetti, pmarques}@fe.up.pt

Abstract - **The paper presents an agent-based simulation model of an order picking system. A pharmaceutical warehouse is used as case study with the purpose of improving the implemented picking processes. Warehousing activities affect the total logistic costs of a company or supply chain. The optimization of the required picking operations is one of the most important objectives when attempting to reduce the operative costs. This study intends to provide a proof-of-concept agent-based model for scheduling the number of human resources required for picking activities. The improvement in the service and the planning of the manpower used in the warehouse, thus achieved, leads to operation-cost reductions. The goal is accomplished by using the NetLogo agent-based simulation framework. The simulation outcomes suggest that dimensioning human resources is a means to satisfy the desirable level of customer's service.**

Keywords - Warehouse Simulation; Order Picking system; Agent-Based model; NetLogo.

I. INTRODUCTION

Recent trends in the warehouse planning have resulted in order picking design and management being more important and complex. Order picking operation is one of the logistic warehouse's processes. It comports the retrieval and collection of articles from a storage location in a specified quantity into a box to satisfy a customer's order.

Customers tend to order more frequently, in smaller quantities, and they require customized service. On the other hand, companies tend to accept late-arrival orders while they need to provide rapid and timely delivery within tight time windows [1]. In general, lead times are under pressure. This is particularly true for pharmaceutical distribution centres.

In this business, pharmacies can order at the click of a button and expect inexpensive, rapid and accurate delivery. Obviously, managing order picking operations effectively and efficiently is a challenging process for the warehouse function. A key objective is to shorten throughput times for order picking, and to guarantee the meeting of due times for shipment departures. In order to offer high customer service level and to achieve economies of scale in transportation to support the related costs, these small size, late-arrival orders need to meet the tight shipment time fence. Hence the time available for picking orders at warehouses becomes shorter, which imposes higher requirements on order processing time at warehouses [2].

Order picking operations often consume a large part of the total labour activities in the warehouse ([3], even claims up to 60%), and for a typical warehouse, order picking may account for 55% of all operating costs [4]. Most of the warehouses employ humans for order picking.

According to the movement of human and products, order picking is organized into picker-to-parts and parts-to-picker systems. In a picker-to-parts system the picker (the person that performs the order picking operation) walks along the aisles to pick items. In this system is used the pick by order. During a pick cycle, pick information is communicated by a handheld terminal or a voice picking system. No paper pick lists are needed. The parts-to-picker systems are usually implemented by the usage of "Automated Storage and Retrieval Systems" (AS/RS).

In the present case study we use a pharmaceutical warehouse that has four different storage areas depending on the type of product stored: products with low rotation rate, products with high rotation rate, big and fragile products and special products (inflammable). The maximum number of pickers is 15.

To simulate the order picking operations, an agent-based model of the warehouse is used. The agent-based model represents the real order picking entities and simulates the customer service indicators. For this work, we used the NetLogo modelling framework [5] to rapidly prototype simple, yet realistic, "what-if" order picking scenarios and analyse the system performance under different real set-ups. NetLogo is a free open-source programmable modelling and simulation platform, appropriate for modelling complex systems. One of its main advantages is the ease of programming. The language is very intuitive and specifically designed for agent-based modelling, thus the user needs only to program agent behaviour, not the agents themselves. Moreover, the researcher community extensively supports the modelling platform and regularly develop a number of tools useful to the modeller.

This paper has the following research objectives: (i) to assign the correct number of pickers for a certain average of served orders; (ii) to provide a tool based on a simulation model to analyse the performance of the order picking process; (iii) to calculate the demand of each type of product based on real data; (iv) to calculate the orders rate that enter in the warehouse; (v) to create a framework with the capacity of generating orders randomly.

This paper is organized as follows. In Section II the existing literature is reviewed, the real system is presented in Section III. A modelling framework, the NetLogo implementation and the validation experiments are shown in Sections IV, V and VI. Conclusion and future work follow in Section VII.

II. Related Work

The two major types of order picking systems can be distinguished into parts-to-picker and picker-to-parts systems. De Koster et al. [1] have provided an extensive literature review of the order picking operations and their implementations. One of their conclusions was the lack of attention from the researcher community for the pickers-to-parts order picking systems despite them being the dominant implemented approach.

Picker-to-parts systems occur in two types: pick by order and pick by article (batch picking). It is also possible to distinguish picker-to-parts systems by the order arrival and release. This can be either deterministic or planned [6] or stochastic [7].

A polling model can describe the order arrivals and processing; a system of multiple queues accessed in a specified sequence by a multiple servers [8]. Hwang et al. [9] use clustering-based heuristic algorithms for the batching of orders for order picking in a single-aisle automated storage and retrieval system. Daniels et al. [10] consider the warehouse in which goods are stored at multiple locations and the pick location of a product can be selected dynamically.

There are also relevant applications in operation management. For instance, Koenigsberg and Mamer [11] consider an operator who serves a number of storage locations on a rotating carousel conveyor. Bozer and Park [12] presented a single-device polling-based material handling systems. Although these systems have been widely researched, they have not yet received systematic treatment and application in the picking process organized in a picker-to-parts system. The same situation occurs with the agent-based models. The literature mainly presents agent-based approaches to solve order picking problems where goods are stored at multiple locations or warehouse control solutions [13], [14]. The study described in this paper is applied to a real case, which the picking is organized in a picker-to-product system.

This work contributes to the literature by exploring the agent-based metaphor to simulate an order picking system in a realistic scenario using real data of a pharmaceutical warehouse.

III. REAL SYSTEM DESCRIPTION

Figure 1 shows the layout of the order picking workstation under study. This system can be classified as a picker-to-product system. Pickers work to fulfil orders. The number of order lines in an order is referred to as order size. Order sizes may vary significantly.

The pharmaceutical distribution center has four different storage areas depending on the type of product stored: products with low rotation rate, products with high rotation rate, big and fragile products and special products (inflammable) and a maximum number of 15 pickers.

The storage areas are arranged in a pre-defined layout and there is a common conveyor to transports the order boxes between them. A customer order may require products from one or more storage areas and the time to collect the products is different for each case.

At the order picking workstation, orders arrive sequentially using boxes on the conveyors. Once the picker and the required order product box are available, the picker moves to the product rack, then picks a number of required items and place it on the conveyor to be transported to the position where the order product box waits. The Figure 1 shows a scheme of this process. The picker works on one order at a time until all lines of the order have been picked and the order is said to be finished. When an order is finished, the system moves the finished order box to the dispatch area.

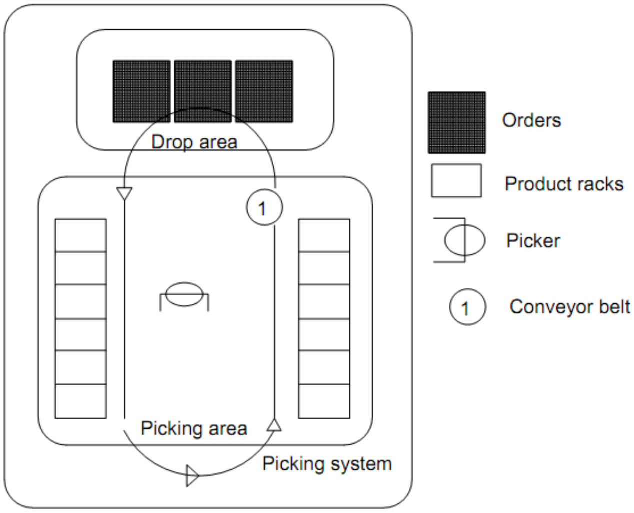


Figure 1. A scheme of order picking process

In this type of system, the picker can only pick items from the product racks that belong to the order currently being processed.

If for any reason the product is not available, the picking time is above a limit or some error occurs the order is deviated and need a special procedure is used to finish it.

IV. SIMULATION MODEL DESCRIPTION

In the following section we discuss the concepts of a pharmaceutical order picking system focusing on the picker-

Copyright (c) IARIA, 2012.   ISBN: 978-1-61208-234-9

to-parts system. We describe the processes taking place during the operation of the warehouse as well as the data used to model the system.

The workstation is modelled using an agent-based approach. Several pickers, products stored in racks, boxes with the products to complete the orders and a server-based management information system, compose such a system.

The user can adjust this model changing the number of pickers. The orders arrive randomly at the workstation with a rate of 7, 8 orders per minute and they have random lines of products based on the real data. This value is obtained by the analysis of the data of one entire month (65 000 records). The result is depicted in Figure 2 in a Poisson distribution. Orders are created based in this real rate.

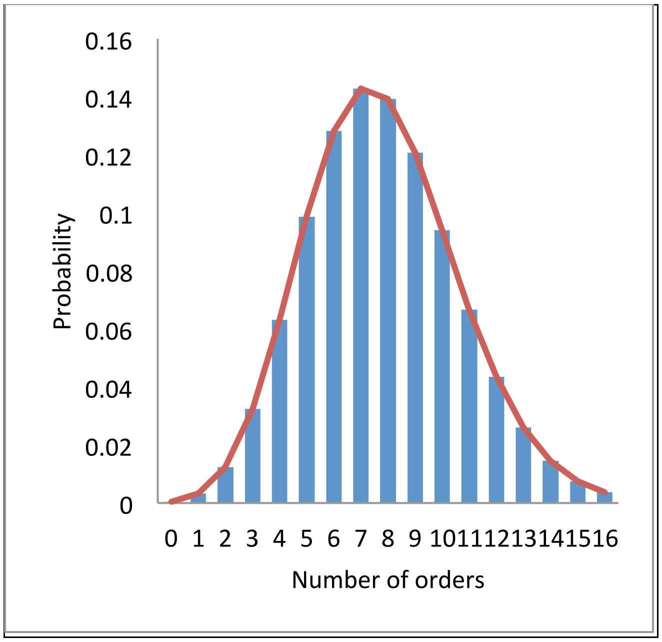


Figure 2. Poisson distribution of the real data

In this model, customer orders can require one or more product lines from one or more storage areas. To describe this situation it was assumed four types of final product:

- if an order requires one or more products lines from one area, we call it a PROD1ZONE;
- if an order requires one or more products lines from two areas, we call it a PROD2ZONE;
- if an order requires one or more products lines from three areas, we call it a PROD3ZONE;
- if an order requires one or more products lines from four areas, we call it a PROD4ZONE.

The frequency and the average preparation time per each type of order are calculated using the real data from entire days. And the result is depicted in Table I.

Table I. Demand production and time preparation

Type of order	Frequency per Type of order (%)	Average Preparation Time (min)
PROD1ZONE	30.46%	5'
PROD2ZONE	27.89%	15'
PROD3ZONE	22.91%	25'
PROD4ZONE	18.74%	43'

This means that in 10 orders the amount of each type of final product is:

- PROD1ZONE: 3.1 orders
- PROD2ZONE: 2.7 orders
- PROD3ZONE: 2.3 orders
- PROD4ZONE: 1.9 orders

For this model it was assumed that products are always available in the warehouse, the pickers are equally skilled (homogeneous agents) for the order picking operations, the workspace is considered an open space and the time to pick a product from the shelf is standard and do not vary with the product.

The orders can have many different statuses: “arriving”, “queuing”, “placing”, “preparation”, “finishing” and “leaving” as it can be seen in Figure 3. The process time used in this paper represents an aggregation of all components that contribute to the processing time at the order picking workstation, the Effective Process Time (EPT). Initially orders move towards the warehouse entrance and stand in a sequential order until a place is chosen. If a free place exists, then the order must navigate to that place. Once the order has reached the place the status is changed to “waiting” and a counter is started. After the pre-defined finishing time has passed, the order changes its status to “leaving”.

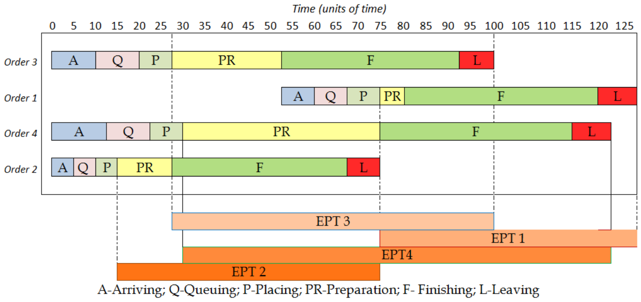


Figure 3. Orders status

Jacobs et al. [15] presented an algorithm to compute the EPT realizations directly from arrival and departure events.

An order picking workstation is characterized by several process time components (see Figure 3). At the core of the process is the time required for picking items (preparing and finish time), that is the raw pick time. Next to the raw pick time, pickers may require some setup between processing of orders. Conveyor systems may break down, causing unavoidable delays. Picker availability is also an issue since it is likely that a picker is sometimes not present at the workstation. These components are aggregated into a single EPT. The idea is then to reconstruct the EPT directly from order arrival and departure times registered at the operating order picking workstation under consideration with the obvious advantage that it´s not need to quantify each component that contribute to the process time.

Figure 3 shows an example of arrivals and departures of four orders at an order picking workstation. An arrival *Ai* occurs at the moment an order *i* is prepared at the order picking workstation. A departure *Di* occur when the picking has been finished and the respective order *i* is finished.

EPT realizations are calculated using the following equation:

$$EPT_i = D_i - A_i \quad (1)$$

where D_i denotes the time of i^{th} departing order and A_i denotes the arrival of the corresponding i^{th} order. The bottom part of Figure 3 illustrates how EPT realizations are obtained using Equation (1).

The picker agent has various statuses: “selecting the product”, “getting products from the rack”, “going to the workspace”, “preparing the product” and “sending the product”.

V. NETLOGO IMPLEMENTATION

In this paper, we describe the use of NetLogo as a rapid prototyping tool for an agent-based simulation framework to evaluate the setup of a pharmaceutical warehouse order picking system.

We present how our problem was modelled and which abstractions were used to achieve the outlined objectives. Furthermore, complexities and constraints inherent to this problem were identified. From that, a simplified model of an abstraction of the application domain was created without losing key aspects. Our purpose is to simulate the activities and operation taking place in an order picking system in way (i) to assign the correct number of pickers for a certain average of served orders; (ii) to simulate the orders behaviour: served and diverted; (iii) to calculate the orders rate that enter in the warehouse.

There are several concepts and agents involved in this model:

1. The Orders:

Orders are randomly generated to “arrive” in the warehouse following a probability distribution according to the historical data distribution. A preparation place is assigned (“placing”) to each order. Here, the order assumes the state “waiting” until the order picking operations are finished (“finishing”). Upon conclusion, the order assumes the state “leaving” and is forwarded to the dispatching area. The demand for each type of product is based on data from a pharmaceutical company.

If the waiting time is too long, (for any reason: product not found, place not available) the order is diverted.

2. The picker:

The user before setup can define the number of pickers and their initial location is randomly generated in the workspace. Simple reactive agents based on simple “if-Then” rules implement the pickers. The picker collects all the products to finish the order. There are four types of final products: PROD1ZONE, PROD2ZONE, PROD3ZONE and PROD4ZONE. Each final product has different picking time to be prepared (defined in the source code). Once the products have been collected, the picker moves to the sending area and places it on the conveyor. The picker restarts the cycle. In the proposed model, pickers are represented by agents.

3. The server:

The server is an agent responsible for the managing and dispatching of the orders.

If all positions for preparing the orders are full, the server does not allow orders to enter into the system to do the order picking operation; this causes a sequential order (“queuing” of orders).

The only interface variable that the user must set before the model runs is the number of pickers. All the other variables can be changed, allowing a dynamic observation.

The user can change also the following variables:

- The demand for each type of final product;
- The chance that orders are generated;
- The speed of the conveyor.

Various monitors and plots allow the user to display the result of these dynamics:

- The total number of orders;
- The number of free places;
- The number of instances of each type of product;
- A chart plotting the number of orders served every 100 ticks (NetLogo unit of time);
- A chart plotting the number of diverted orders every 100 ticks;
- An average number of orders served;
- An average number of diverted orders;

The user can experiment to change the values and seeing the result through the monitors and charts. It is possible to observe the visual phenomena that are developed such as bottlenecks, queues and the spatial distribution of diverted orders.

There are important aspects in this model:

- Queuing at the warehouse entrance;
- Bottlenecks at the exit;
- Congestion on the conveyor;
- The stochastic aspects are inherent in the model;
- Agents don't always move in exactly the same way;
- Demand of different final products may vary naturally.

Although NetLogo is a simple simulation framework, it proves to be a very useful tool for creating this type of agent-based model of real scenario. With respect to other simulation paradigms, the agent-based approach offers the users the possibility to observe not only the dynamic of the system but also the interaction of the situated entities in the system. One key point is also the agent movement. Specifically creating a realistic system where agents (e.g., pickers) can move having specific goals and destinations. The limitation in allowing agents to move dynamically in the NetLogo environment is that they are constrained moving discretely on a grid-based space patch-by-patch rather having smooth trajectories.

VI. VALIDATION EXPERIMENT

In this section the simulation experiments are discussed to validate the proposed model. First, a simulation scenario has been created to use as a test case representing the "real life" operating order picking workstation. Then, the model has been simulated at a real utilization level (using different number of picker-agents, the real demand and the real order picking times) to generate order arrival and departure events. Subsequently, these events are used as variables for the global model. Next, the model was simulated at various utilization levels (varying the number of pickers and conveyor belt speed) to measure the number of orders served versus the number of diverted orders.

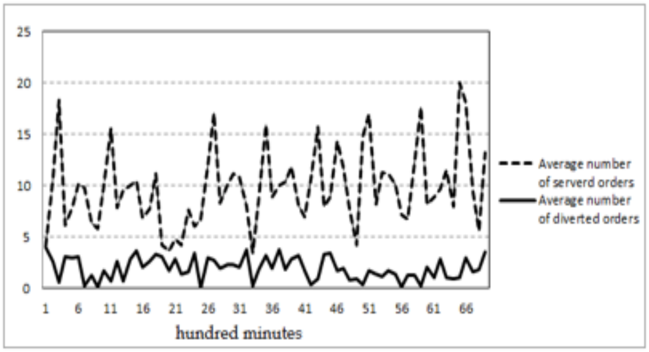


Figure 4. Orders behaviour in real system

For the aforementioned experimental set-up, a comparison has been made over the average number of served and diverted orders of this simulation model with those of the real system. In this way, it´s possible to assess the accuracy of the predictions of the average number of served orders. The real system's data are depicted in Figure 4 (in green the average number of served orders) while the simulation obtained results are depicted in Figure 5. The word "tick" in the graph of Figure 5 is a measure for the time that is used by NetLogo.

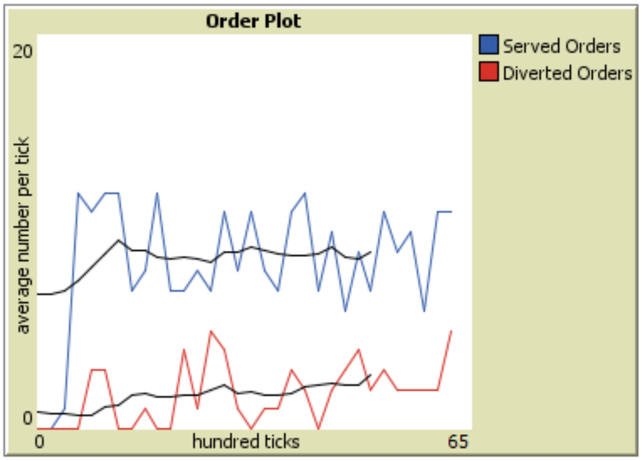


Figure 5. Order behaviour in simulated system

Analysing the order plot over a complete working day's operations, we get a good idea of the order average of served orders and diverted orders. By looking at the historical company's data, we can see that the diverted orders are in line with the current observations.

VII. CONCLUSION AND FUTURE WORK

This paper presented an order picking model implemented in the NetLogo agent-based platform. Although the model may appear to be simplistic, its conceptualisation encompasses many aspects of the observed system in the real world. The model manages to predict the average number of served orders using a certain amount of human resources by means of an agent-based simulation model with real data from a pharmaceutical distribution center. Arrival and departure time data are the only input required to calculate the time to complete the orders. The validation of the simulation study demonstrated that the data used are adequate to the required results. It was found that the proposed model accurately predicts the defined goal in a satisfactory degree.

In practice, the actual pick rate does not deviate from the expected rate. With this regard, the EPT represents the actual pick time of an order picking workstation. This will allow the identification of possible improvements for order picking activities.

The proposed model has practical use because collecting arrival and departure data of orders is relatively simple in warehouses and the output is easily perceived.

In future work, the model's layout will be improved and demonstrate how the presented approach in this paper can be applied to a more detailed order picking workstation (i.e., heterogeneous agents to represent differently skilled pickers, automation mechanism in the warehouse, etc.).

Besides that, the current advancements in order picking technology have allowed multiple orders to be processed simultaneously by a picker, which is often the case in large-scale warehouses. Thus, orders routings in large-scale warehouses would not be processed in a FIFO sequence at the workstation. Performance analysis, in such a context, of order picking workstation will be also subject to future work. Therefore, it is important to improve and enhance the attributes of the agent-based model to tackle these issues.

ACKNOWLEDGMENT

This project has been supported by the FEUP project “Support system for availability and location of products for picking” and FCT (Fundação para a Ciência e a Tecnologia), the Portuguese Agency for R&D, under the grant, SFRH/BD/67202/2009

REFERENCES

1. R. de Koster, T. Le-Duc, and K. J. Roodbergen, “Design and control of warehouse order picking: A literature review,” European Journal of Operational Research, vol. 182, issue 2, pp.68–73, October 2007.
2. M. Yu, “Enhancing warehouse performance by efficient order picking,” Netherlands Research School for Transport, Infrastructure and Logistics, Doctoral Thesis, October 2008.
3. J. Drury, “Towards more efficient order picking,” IMM Monograph No. 1. Report, The Institute of Materials Management Cranfield, U.K, 1988.
4. J. A. Tompkins, J. A. White, Y. A. Bozer, and J. M. A. Tanchoco, ”Facilities planning,” New York, John Wiley and Sons, 2003.
5. U. Wilensky, “NetLogo,” Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, http://ccl.northwestern.edu/netlogo/, (accessed September 2012).
6. N. Gademann and S. Van de Velde, “Order batching to minimize total travel time in a parallel-aisle warehouse,” IIE Transactions vol. 37, issue 1, pp. 63-75, 2005.
7. T. Le-Duc, “Design and control of efficient order picking process,” Rotterdam School of Management, Erasmus University Rotterdam, the Netherlands, Ph.D. dissertation, 2005.
8. M. M. Srinivasan, “Nondeterministic polling systems,” Management Science, vol. 37 no. 6, pp. 667–681, June 1991.
9. H. Hwang, W. Baek, and M. Lee, “Clustering algorithms for order picking in an automated storage and retrieval system,” International Journal of Production Research, vol. 26, issue 2, pp. 189–201, 1988.
10. R. L. Daniels, J. Rummel, and R. Schantz, “A model for warehouse order picking,” European Journal of Operational Research, vol. 105, pp. 1–17, February 1998.
11. E. Koenigsberg and J. Mamer, "The analysis of production systems," International Journal of Production Research, vol. 20, issue 1, pp. 1-16, 1982
12. Y.A. Bozer and J.H. Park, “Expected move request waiting times in single-device, polling-based material handling systems,” IIE Trans vol. 33, issue 1, pp. 55–63, 2001.
13. B. Kim, J. G. Robert, and S. H. Sunderesh, “Intelligent agent modeling of an industrial warehousing problem,” IIE Transactions, vol. 34, issue 7, pp. 601-612, 2002.
14. B. Kim, S. H. Sunderesh, J. G. Robert, and A. St. Onge, “Intelligent agent based framework for warehouse control,” Proceedings of the 37th Hawaii International Conference on System Sciences, January 2004.
15. J. H. Jacobs, L. F. P. Etman, E. J. J. van Campen, and J. E. Rooda, “Characterization of operational time variability using effective processing times,” IEEE Transactions on Semiconductor Manufacturing, vol. 16, no. 3, pp. 511–520, 2003.

Monte Carlo Simulation of an Optical Differential Phase-Shift Keying Communication System with Direct Detection Impaired by In-Band Crosstalk

Genádio Martins, Luís Cancela and João Rebola
Instituto de Telecomunicações, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal.
Department of Information Science and Technology, Instituto Universitário de Lisboa (ISCTE-IUL), Av. das Forças Armadas, Edifício II, 1649-026 Lisboa, Portugal.
genadio.martins@gmail.com, luis.cancela@iscte.pt, joao.rebola@iscte.pt

Abstract— **A Monte Carlo (MC) simulation is used to evaluate the performance of an optical differential phase-shift keying (DPSK) communication system impaired by in-band crosstalk. The MC simulation results are used to study the validity range of an analytical work proposed in the literature addressing the same problem. It is shown that the performance estimates obtained using the analytical work become incorrect, whenever the effect of inter-symbol interference (ISI) is enhanced, for example, for optical filter bandwidths below twice the DPSK signal bandwidth. When ISI is negligible, a slight discrepancy is observed between the performance estimates of both methods, because the two formalisms consider different models for the random phase noise.**

Keywords – optical communication systems; Monte Carlo simulation; differential phase-shift keying; in-band crosstalk.

I. INTRODUCTION

In the last decade, differential phase-shift keying (DPSK) modulation [1] has attracted much attention in optical communication systems, mainly due to its ~3 dB advantage in receiver sensitivity when balanced detection is used, in comparison with the conventional on-off keying (OOK) modulation. Additionally, the DPSK outperforms the conventional OOK in its robustness to transmission impairments and tolerance to signal power fluctuations. Regarding optical networking, it has been experimentally found that the DPSK signal with balanced detection has ~6 dB higher tolerance to in-band crosstalk than the OOK signal [2].

Crosstalk due to the imperfect isolation of optical components, such as optical filters, (de)multiplexers and optical switches, used in optical network nodes is considered one of the most important physical layer limitations in optical networks. The crosstalk signals may arise from distinct sources or from the same source of the original signal. When the crosstalk signal arises from distinct sources, it may have the same wavelength as the original signal or have different wavelengths, giving rise to in-band crosstalk and inter-band crosstalk, respectively [3]. In-band crosstalk is the most severe form of crosstalk, because the signal-crosstalk beating terms originated at the receiver cannot be removed by filtering [4].

In this work, the influence of in-band crosstalk on the performance of an optical DPSK receiver is analyzed by Monte Carlo (MC) simulation. One major goal is to study the validity range of an analytical work proposed in the literature [4], [5], that addresses the same issue, keeping in mind that this work was developed considering an isolated DPSK symbol. The MC simulation, on the other way, considers a sequence of DPSK symbols, and takes into account the inter-symbol interference (ISI) effect on the performance. The impact of in-band crosstalk on the receiver performance is investigated for different receiver filters configurations, for an increasing crosstalk level, by taking into account the delay between the DPSK signal and the crosstalk signal, for different sequences of bits on the crosstalk signal, by considering receiver imperfections and for multiple interfering terms.

This work is organized as follows: Section II describes the model used to characterize the optical DPSK receiver and the MC simulation implementation. Numerical results are obtained in Section III. The conclusions are outlined in Section IV.

II. SYSTEM DESCRIPTION

A. Optical DPSK receiver

The structure of a typical direct detection DPSK receiver using balanced detection is shown in Fig. 1. It consists of an optical pre-amplifier with a constant power gain *G* over the amplifier bandwidth; an optical filter with –3 dB bandwidth B_o; a delay interferometer (DI) with a differential delay equal to the bit period T_b; a balanced photodetector consisting of two photodetectors with responsivities R_λ^+ and R_λ^-; and a post-detection electrical filter with –3 dB bandwidth B_e.

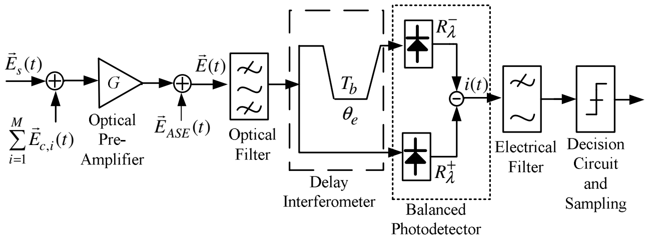


Figure 1. Balanced DPSK receiver schematics.

The electrical fields, represented in Fig. 1, are described with more detail in [4]. The electrical field at the amplifier output, $\vec{E}(t)$, can be expressed as [4]

ISBN: 978-1-61208-234-9

$$\vec{E}(t)=\sqrt{G}\cdot\vec{E}_s(t)+\sqrt{G}\cdot\sum_{i=1}^{M}\vec{E}_{c,i}(t)+\vec{E}_{ASE}(t), \qquad (1)$$

where the first term, $\vec{E}_s(t)$, corresponds to the electrical field of the incoming DPSK signal, described as $\vec{E}_s(t)=\sqrt{P_s}\,\exp\left[j\theta_s(t)\right]\vec{e}_s$, where P_s is the average signal power at the optical pre-amplifier input; $\theta_s(t)$ is the signal phase that carries the binary DPSK information; and $\vec{e}_s$ is the signal polarization unit vector. The second term of (1), $\sum_{i=1}^{M}\vec{E}_{c,i}(t)$, corresponds to the electrical field of the in-band crosstalk, described by M different DPSK signals with the same wavelength and the same bit rate of the original signal. The complex envelope of the i-th crosstalk signal field can be represented as $\vec{E}_{c,i}(t)=\sqrt{P_{c,i}}\,\exp\left[j\left(\theta_{c,i}(t)+\phi_{c,i}(t)\right)\right]\vec{e}_c$, where $P_{c,i}$ is the crosstalk power; $\phi_{c,i}(t)$ is a random phase noise; $\theta_{c,i}(t)$ is the crosstalk phase that carries the binary information; and $\vec{e}_c$ is the crosstalk signal polarization unit vector. The crosstalk level of the i-th interferer is defined as the ratio between the crosstalk power, $P_{c,i}$, and the signal power, P_s. The total crosstalk level is the sum of the crosstalk levels of all the interferers. In this work, we assume that signal and crosstalk fields are co-polarised, $\vec{e}_s=\vec{e}_c$ [5]. Finally, the third term of (1), $\vec{E}_{ASE}(t)$, corresponds to the electrical field of the amplified spontaneous emission (ASE) noise originated at the optical pre-amplifier. The ASE noise is assumed as a zero mean white stationary Gaussian noise, co-polarised with the signal, and with single-sided power spectral density described by $N_o=h\nu_s(G-1)F/2$, where $h\nu_s$ is the photon energy at the signal wavelength, and F is the pre-amplifier noise figure. The DI is modeled as in [4] and the photodetectors are assumed as ideal square-law detectors.

Ideally, a DPSK optical receiver should have $R_\lambda^+=R_\lambda^-$ and $\theta_e=0$. However, practical optical DPSK receivers may exhibit imperfections, such as: responsivity imbalance, which is quantified by $K=10\log_{10}(R_\lambda^+/R_\lambda^-)$; and an offset between the transmitting laser frequency and the frequency leading to perfect constructive/destructive interference at the DI output. This last effect can be characterized by the phase error θ_e and modeled by the DI detuning $\Delta f=\theta_e/(2\pi T_b)$ [6].

B. Implementation of the MC simulation

The main goal of this work is to develop a MC simulation tool capable of evaluating the performance of an optical DPSK receiver impaired by in-band crosstalk. The MC simulation is used to study the validity range of the analytical work proposed in [4], which was developed considering an isolated DPSK symbol. In the simulation, a pseudorandom deBruijn binary sequence with length N_b bits is generated, which allows to study the effect of ISI on the DPSK receiver performance. Then, to obtain the DPSK format, the signal is encoded differentially, considering that for each bit 'one' the optical phase does not change, and for each bit 'zero' a π-phase change is introduced. The bits sequence on the DPSK crosstalk signal is generated randomly and the random phase noise on the crosstalk signal is generated considering a Brownian motion model [7]. Notice that in the analytical work [4], the random phase noise is assumed constant over the bit period and with a uniform distribution. In the simulation, the ASE noise is generated using a random number generator, which follows a Gaussian distribution with zero mean and variance N_oB_{sim}, where B_{sim} is the bandwidth used in the MC simulation [8].

At each MC iteration, in accordance with (1), a sample function of the ASE noise and a sample function of the DPSK crosstalk signal with random phase noise are added to the DPSK original signal obtained from the deBruijn sequence. Then, the current at the decision circuit input is computed using the MC simulator, by passing the field described in (1) through the receiver model (using the frequency domain description of the optical and electrical filtering), and it is sampled. After sampling, each received bit, that is corrupted by noise and crosstalk, is compared to the corresponding transmitted bit to find out if an error has occurred. The MC iterations are repeated, until a specified N_e number of counted errors is attained. Then, the bit error probability (BEP) is estimated using $\text{BEP}=N_e/\left[N_{it}(N_b-1)\right]$, where N_{it} is the number of iterations of the MC simulator.

III. RESULTS

In this section, the results corresponding to the performance of the optical DPSK receiver obtained using MC simulation are presented and compared with analytical results [4]. The performance of the optical DPSK direct detection receiver is assessed in presence of in-band crosstalk, considering: 1) different optical and electrical filters with different –3 dB bandwidths; 2) different crosstalk levels; 3) different sequence of bits on the DPSK crosstalk signal; 4) different delays between the crosstalk signal and the original signal; 5) two types of receiver imperfections, responsivity imbalance and DI detuning; and 6) multiple interfering terms. All results have been obtained considering the parameters shown in Table 1, and unless otherwise stated, these parameters are used throughout this work.

TABLE I. SIMULATION PARAMETERS

Parameter	Value
Number of bits (N_b)	64
Gain (G)	30 dB
Amplifier noise figure (F)	5 dB
Bit period (T_b)	0.1 ns
Responsivity (R_λ^+, R_λ^-)	1 A/W
Phase error of the interferometer (θ_e)	0
Total number of counted errors (N_e)	100
Laser spectral linewidth	10 MHz

In our results, we consider two optical receivers filters configurations: 1) ideal receiver configuration based on a rectangular optical filter and an integrate-and-dump electrical filter, which considers $B_eT_b=1$; and 2) Gaussian receiver configuration based on a Gaussian optical filter and a Gaussian electrical filter, which considers $B_e T_b = 0.7$.

A. *Optical DPSK receiver without in-band crosstalk*

In this subsection, the MC simulation is used to evaluate the optical DPSK receiver performance impaired only by ASE noise. The MC simulation results are compared to the results of the analytical formalism [4]. The study is performed for different optical signal powers and for different optical filter bandwidths for the ideal receiver configuration. In-band crosstalk and receiver imperfections are neglected in this study.

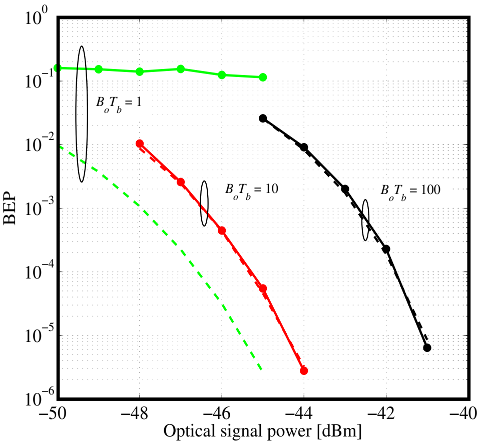


Figure 2. BEP as a function of the optical signal power, for the ideal receiver configuration with B_oT_b= 1, 10 and 100. Solid lines: MC simulation; dashed lines: analytical results.

Fig. 2 shows the BEP estimates obtained using MC simulation and using the analytical formalism [4] as a function of the optical signal power, P_s, for different optical filter –3 dB bandwidths,. As the MC simulation is obtained with a sequence of bits, it takes into account the ISI effect on the DPSK receiver performance. The analytical formulation neglects this effect, and due to this difference, for smaller normalized optical filter bandwidths $(B_oT_b = 1)$, the BEPs estimated from the MC simulation and the analytical formulation are very discrepant. For larger optical filter bandwidths $(B_oT_b = 10, 100)$, the MC simulated results are very similar to the results obtained with the analytical formulation, and ensure us that the implementation of the MC simulator is correct. Fig. 2 also shows that, for the same optical signal power, there is a severe increase of the BEP with the optical filter bandwidth enlargement. This occurs because with the increase of the optical filter bandwidth, the filtered ASE noise power is higher, and the DPSK receiver performance becomes degraded.

Fig. 3 depicts the BEPs obtained with the MC simulation and with the analytical formalism [4], for normalized optical filter bandwidths where the ISI effect on the performance is relevant, $B_oT_b \le 2$. As shown in Fig. 3, it can be assumed that the simulated results become sufficiently approximated to the analytical ones for B_oT_b = 2. This means that the ISI starts to lose its influence as the dominant source of performance degradation and provides a reference for the optical filter bandwidth above which, the accuracy of the analytical formalism, that neglects ISI, is ensured.

Other receiver filters configurations have been studied and a good agreement was found between the BEPs obtained using MC simulation and analytically. A good agreement with the results presented in [9] was also found, which further ensure us that the MC simulation is well implemented.

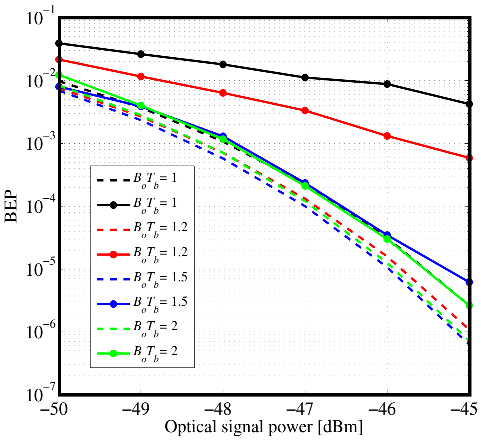


Figure 3. BEP as a function of the optical signal power, for the ideal receiver configuration, considering $B_oT_b \le 2$. Solid lines: MC simulation; dashed lines: analytical results.

B. *Optical DPSK receiver impaired by in-band crosstalk due to a single interferer*

In this subsection, the performance of the optical DPSK receiver is analyzed in presence of ASE noise and in-band crosstalk, considering a single interfering term. The estimates obtained through MC simulation are compared to the estimates obtained with the analytical formalism [4]. Firstly, the BEP is evaluated for different crosstalk levels and optical filter bandwidths. Then, the influence of a delay applied on the crosstalk signal in relation to the original signal, of different bits sequences on the interferer and of receiver imperfections on the DPSK receiver performance is studied.

Fig. 4 shows the BEP estimates from the MC simulation and from the analytical formalism [4] as a function of the optical signal power, considering the ideal receiver configuration with B_oT_b = 1, 10 and 100, and a crosstalk level equal to –12 dB. The BEPs estimated from the MC simulation, which are represented by the solid green line, are obtained assuming that the random phase noise has a uniform distribution with constant phase along the bit period, similarly to the assumption considered in the analytical work [4]. The other BEPs obtained using MC simulation presented in Fig. 4, are estimated considering that the random phase noise follows the Brownian motion model [7].

Regarding the methods comparison, Fig. 4 shows that for smaller optical filter bandwidths, as the ISI is dominant, the difference between the BEPs obtained using the MC simulation and the analytical formalism is enhanced. For higher normalized optical filter bandwidths (B_oT_b = 10 and 100), the BEPs obtained using the MC simulation with the Brownian motion assumption and in-band crosstalk are slightly discrepant to the analytical BEPs. This difference is attributed to the Brownian motion model assumed for the random phase noise, since when the MC simulation assumes a uniform distribution for the random phase noise, the simulated and analytical curves become very similar. Random phase noise with uniform distribution is a good model to describe the crosstalk that impairs an isolated symbol. However, this model loses reality for a sequence of symbols, due to the phase discontinuity introduced between adjacent symbols. The Brownian motion model provides a physical description that is suitable to describe the influence of random phase noise in a sequence of symbols, due to its temporal continuity.

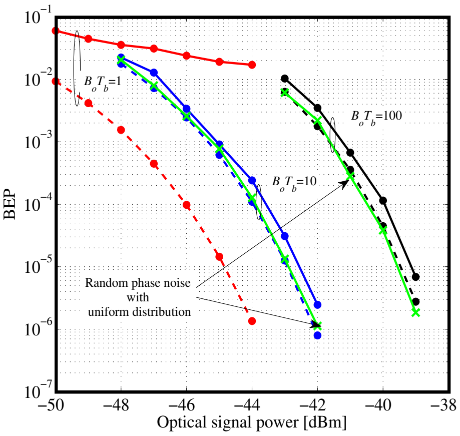


Figure 4. BEP as a function of the optical signal power, considering a crosstalk level equal to –12 dB, and an ideal receiver configuration with B_oT_b = 1, 10 and 100. Solid red, blue and black lines: MC simulation with Brownian motion assumption; solid green lines: MC simulation with uniform assumption; dashed lines: analytical results.

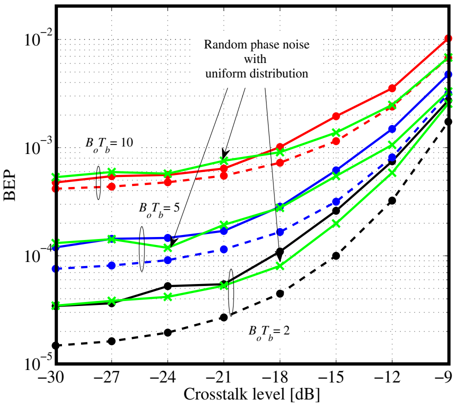


Figure 5. BEP as a function of the crosstalk level, for an optical signal power equal to –46 dBm, and the ideal receiver configuration with different optical filter bandwidths. Solid red, blue and black lines: MC simulation with Brownian motion assumption; solid green lines: MC simulation with uniform assumption; dashed lines: analytical results.

Fig. 5 shows the BEPs obtained by MC simulation and analytically, as a function of the crosstalk level with B_oT_b as a parameter considering the ideal receiver configuration and an optical signal power equal to –46 dBm. Fig. 5 shows that, whenever the ISI influence on the performance is dominant, the BEPs estimated using MC simulation using both phase noise models are discrepant with the BEPs obtained analytically (see, for example, the curves corresponding to $B_oT_b = 2$ in Fig. 5). When ISI is not relevant ($B_oT_b = 10$) and the crosstalk level is above –18 dB, there is a slight discrepancy between the analytical results and the MC simulation results obtained with the Brownian motion model. When considering the uniform distribution, for a high crosstalk level (above –15 dB), for $B_oT_b = 10$, the analytical results tend to the simulated results.

In the remainder of this subsection, all the MC simulated results are obtained considering the Brownian motion model.

In the following, the performance of the optical DPSK receiver is investigated considering different bits sequences on the DPSK crosstalk signal. Until now, the bits sequence on the DPSK crosstalk signal was always assumed to be random. The next study aims to analyze and compare the performance of the optical DPSK receiver when the bits sequence on the DPSK crosstalk signal is: 1) equal to the sequence of bits on the original DPSK signal; 2) the negation of the sequence of bits on the original DPSK signal; 3) a sequence with only bits ‘1’ and 4) a sequence with only bits ‘0’.

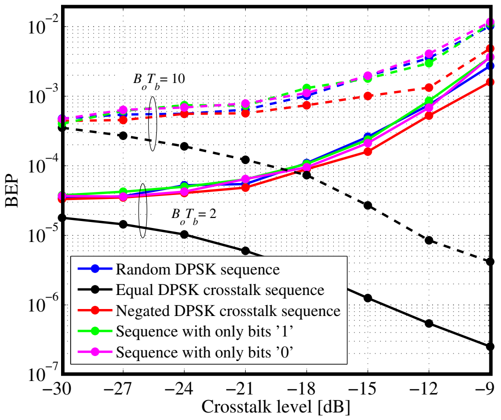


Figure 6. BEP as a function of the crosstalk level, considering an optical signal power of –46 dBm, for different sequences of bits on the DPSK crosstalk signal and for the ideal receiver configuration with B_oT_b = 2 and 10.

Fig. 6 shows the BEP as a function of the crosstalk level, for different bits sequences on the DPSK crosstalk signal considering an optical signal power of –46 dBm, and the ideal receiver configuration with B_oT_b = 2 and 10. As shown in Fig. 6, except for the case where the sequence of bits on the DPSK crosstalk signal is equal to the original DPSK signal, the BEPs estimated using MC simulation, when impaired by other DPSK crosstalk signals with different bits sequences are approximated to the ones obtained considering the random bits sequence. When the bits sequence on the DPSK crosstalk signal is equal to the original DPSK signal, an increase of the crosstalk level results in an improvement of the BEP, because as the DPSK crosstalk signal is added to the original signal, there is a signal power reinforcement.

In order to understand the influence of the bits sequences on the DPSK crosstalk signal, the eye diagrams of the original signal impaired by different bits sequences on the crosstalk signal are shown in Fig. 7, considering: a random bits sequence (left); a bits sequence equal to the original signal (middle) and the negated bits sequence (right). The eye diagrams are obtained for the ideal configuration with B_oT_b = 2, for P_s = –45 dBm and a crosstalk level of –12 dB.

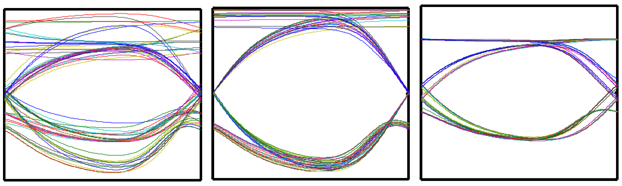


Figure 7. Eye diagrams at the decision circuit input, for the ideal configuration, B_oT_b = 2, an optical signal power equal to –45 dBm, a crosstalk level of –12 dB and considering a random bits sequence on the DPSK crosstalk signal (left), an equal bits sequence (middle) and the negated bits sequence (right).

As can be observed in Fig. 7, it can be concluded that the eye pattern shows a significant enlargement in the eye opening when the bits sequence on the DPSK crosstalk signal is equal to the bits sequence of the original DPSK signal. As a consequence, the BEP achieved through the MC simulation, shown in Fig. 6 is considerably lower.

In the next study, the influence of a delay between the crosstalk signal and the original signal on the optical DPSK receiver performance is analyzed.

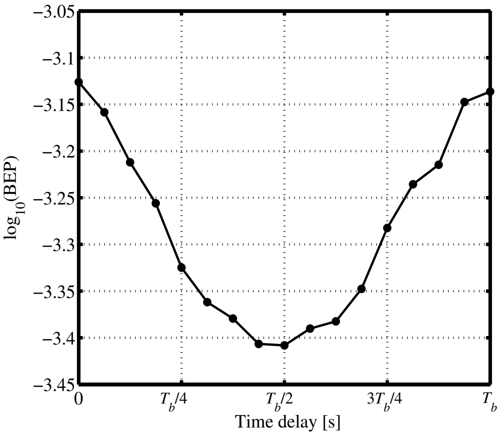


Figure 8. BEP as a function of the delay between the original and crosstalk signals, for the ideal receiver configuration with B_oT_b = 2, an optical signal power equal to –46 dBm, and a crosstalk level of –12 dB.

Fig. 8 shows the BEP as a function of the delay between the original and crosstalk signals, for the ideal receiver configuration with B_oT_b = 2, for P_s = –46 dBm, and a crosstalk level of –12 dB. To achieve an improved accuracy, a value of N_e = 1000 erroneous bits is assumed in the MC simulation. Fig. 8 shows that, the influence of the delay between the crosstalk signal and the original signal on the performance of the optical DPSK receiver is small, since the BEP variation is not much significant. Nevertheless, Fig. 8 shows that the best BEP is achieved for a delay of half the bit period, and that the BEP has a symmetric behavior around this point. The worst BEP is obtained when the crosstalk and the original signals are aligned, which is in agreement with the worst-case situation usually assumed in the literature [4].

Afterwards, the performance of the optical DPSK receiver is analyzed considering the effect of two receiver imperfections: the responsivity imbalance and the DI detuning. The MC simulation of an optical DPSK receiver impaired by imperfections without in-band crosstalk has been validated by comparison of its estimates with the analytical results of [6].

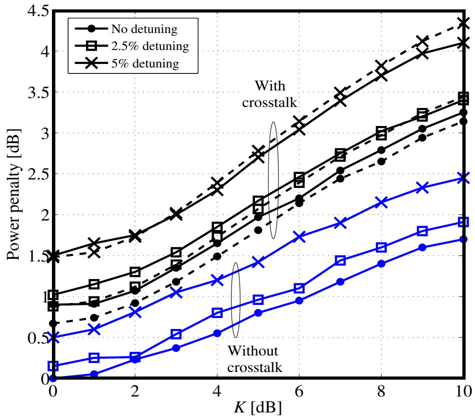


Figure 9. Power penalty as a function of the responsivity imbalance with different interferometer detunings, for the Gaussian receiver configuration with B_oT_b = 5, with a crosstalk level of -15 dB,. Solid lines: MC simulation; dashed lines: analytical results.

Fig. 9 depicts the power penalty as a function of the responsivity imbalance with different interferometer detunings, considering a Gaussian configuration with B_oT_b = 5 and a crosstalk level of –15 dB. The reference for the power penalty is obtained with no detuning, K = 0 dB and without crosstalk, which corresponds to a BEP of 10^{-3} obtained for an optical signal power of –46.5 dBm. Accordingly with Fig. 9, it can be concluded that there is a good approximation between the power penalties obtained using the analytical formalism and the MC simulation. The small differences between the performances might be related with the random phase noise models. Without imperfections, the performance degradation due to in-band crosstalk is about 0.9 dB (taken from the MC simulation results). For K = 10 dB and no detuning, the performance degradation is enhanced to about 1.5 dB. Fig. 9 also shows that the performance degradation due to receiver imperfections is not enhanced with the crosstalk influence.

C. *Optical DPSK receiver impaired by in-band crosstalk due to multiple interferers*

In this subsection, the accuracy of the MC simulation is analyzed in presence of multiple interfering terms in the crosstalk signal.

Fig. 10 shows the BEP as a function of the optical signal power with the number of interferers as a parameter, considering the ideal receiver configuration with $B_oT_b = 2$

(above) and $B_oT_b = 10$ (below) and a total crosstalk level of –12 dB. Fig. 10 shows that with the increase of the optical signal power, the performance degradation due to a higher number of interfering terms is enhanced. For $B_oT_b = 2$, the performance degradation is more noticeable due to the lower ASE noise power. In this case, the differences between the analytical estimates and simulated results are attributed to ISI and to the difference between the random phase noise models.

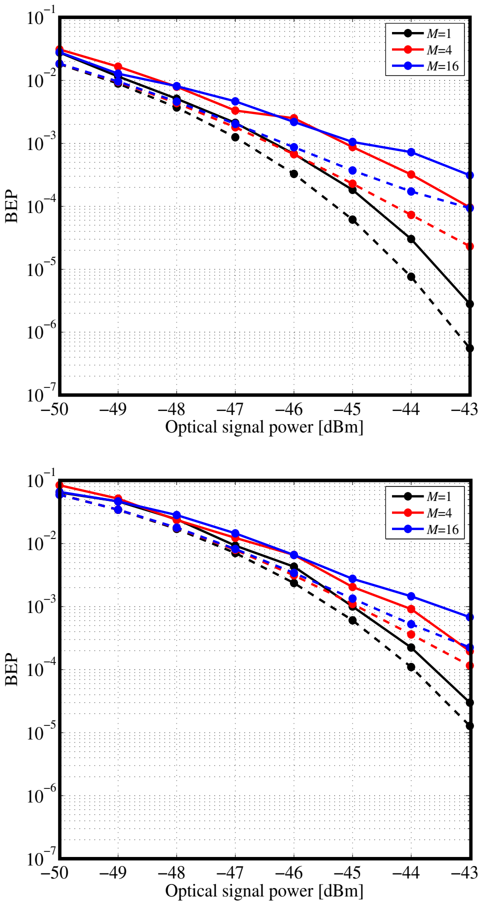


Figure 10. BEP as a function of the optical signal power, considering the ideal receiver configuration with $B_oT_b = 2$ (above) and $B_oT_b = 10$ (below), a total crosstalk level of –12 dB and multiple interfering terms. Solid lines: MC simulation; dashed lines: analytical results.

IV. Conclusions

In this work, a MC simulator has been developed to evaluate the impact of in-band crosstalk in an optically pre-amplified DPSK direct detection receiver and to study the validity range and the limitations of the analytical work proposed in [4]. This study has been performed for the scenario without in-band crosstalk and for the scenario with in-band crosstalk originated from multiple interferers.

It was shown that when the ISI plays a key role as the dominant source of performance degradation, for $B_oT_b < 2$, the BEPs predicted by the analytical work became very discrepant from the ones obtained with MC simulation. As a rule of thumb, $B_oT_b = 2$ provides a good bound for the normalized optical filter bandwidth, where the accuracy of the analytical work is still guaranteed. In presence of in-band crosstalk, it was concluded that even for large optical filter bandwidths, a slight discrepancy is observed between the BEPs obtained using both formalisms due to the fact that the analytical formalism considers that random phase noise has a uniform distribution constant over the bit period, whereas the MC simulation assumes a Brownian motion model for the random phase noise. It was also shown that a delay applied in the crosstalk signal (in relation to the original signal) does not have a relevant influence on the DPSK receiver performance. Furthermore, it was shown that, a DPSK crosstalk bits sequence equal to the original bits sequence leads to a performance improvement due to signal power reinforcement. We have also observed that, the performance degradation due to DPSK receiver imperfections is not enhanced with the presence of crosstalk.

The influence of multiple interfering terms on the DPSK receiver performance was also analyzed, and it was seen that, with the increase of the optical signal power, the performance degradation induced by a higher number of interfering terms is enhanced. The discrepancies between analytical and simulated results, found for multiple interfering terms are similar to the ones obtained with one single interfering term.

References

[1] A. H. Gnauck and P. J. Winzer, “Optical Phase-Shift-Keyed Transmission“, J. Lightw. Technol., vol. 23, no. 1, pp. 115-130, Jan. 2005.

[2] X. Liu, Y.-H. Kao, M. Movassaghi and R. C. Giles, “Tolerance to In-Band Coherent Crosstalk of Differential Phase-Shift-Keyed Signal with Balanced Detection and FEC”, IEEE Photon. Technol. Lett., vol. 16, no. 4, pp. 1209-1211, Apr. 2004.

[3] T. Y. Chai, T. H. Cheng, Y. Ye and Q. Liu, “Inband Crosstalk Analysis of Optical Cross-Connect Architectures”, J. Lightw. Technol., vol. 23, no. 2, pp. 688-701, Feb. 2005.

[4] J. Pires and L. Cancela, “Estimating the Performance of Direct-Detection DPSK in Optical Networking Environments Using Eigenfunction Expansion Techniques”, J. Lightw. Technol., vol. 28, no. 13, pp. 1994-2003, Jul. 2010.

[5] L. Cancela and J. Pires, “Crosstalk Tolerance of Direct Detection DPSK Optical Systems in Presence of Receiver Imperfections”, IET Optoelectron., vol. 6, no. 2, pp. 94-101, Apr. 2012.

[6] G. Bosco and P. Poggiolini, “The Impact of Receiver Imperfections on the Performance of Optical Direct-Detection DPSK”, J. Lightw. Technol., vol. 23, no. 2, pp. 842-848, Feb. 2005.

[7] M. Azizoglu and P. A. Humblet, “Optical DPSK with Generalized Phase Noise Model and Narrowband Reception”, in Proc. IEEE ICC 93, Geneva, vol. 3, pp. 1591-1596, May 1993.

[8] M. C. Jeruchim, P. Balaban and K. S. Shanmugan, Simulation of Communication Systems – modeling, methodology, and techniques, 2nd ed., Kluwer Academic Publishers, 2002, ch. 11.

[9] Q. Zhang, C. R. Menyuk, R. Bajracharya, H. W. Huang, A. Miner, “On the Gaussian Approximation and Margin Measurements for Optical DPSK Systems With Balanced Detection”, J. Lightw. Technol., vol. 28, no. 12, pp. 1752-1760, Jun. 2010.

A Flexible Analytic Model for the Design Space Exploration of Many-Core Network-on-Chips Based on Queueing Theory

Erik Fischer, Albrecht Fehske, and Gerhard P. Fettweis
Vodafone Chair Mobile Communications Systems
Technische Universität Dresden
Dresden, Germany
Email: {erik.fischer, albrecht.fehske, fettweis}@ifn.et.tu-dresden.de

***Abstract*—A continuing technology scaling and the increasing requirements of modern embedded applications are most likely forcing a current multi-processor system-on-chip design to scale to a many-core system-on-chip with thousands of cores on a single chip. Network-on-chip emerged as flexible and high-performance solution for the interconnection problem. There will be an urgent need for fast, flexible and accurate simulation models to guide the design process of many-core system-on-chip. In this paper, we introduce a novel analytic approach for modeling on-chip networks to fulfill these requirements. The model is based on queueing theory and very flexible in terms of supported topology, routing scheme and traffic pattern. The approach overcomes the limitations of the mean value analysis introduced in the existing work. Instead, it provides information about a steady-state distribution of the network routers. This allows to dimension network resources, such as buffers, links, etc. We show the high accuracy of the model by comparison with a cycle-accurate simulation. The model is able to estimate the mean network latency with an accuracy of about 3%.**

Keywords-network-on-chip; noc; queueing theory; analytic model.

I. Introduction

In embedded computing, today's applications show a common trend towards a continuously increasing computational effort and reliability. This is especially true in the area of multi-media and mobile communication. These requirements can only be fulfilled by massively exploiting parallelism. Taking also emerging technologies like 3D chip stacking [1] into account, today's multi-processor system-on-chips (MPSoCs) soon scale to many-core SoCs with thousands of processors on a single chip [2]. Already in 2015, we may have 1000 or more cores on a chip [3].

If we assume such a large number of cores, the interconnection problem becomes a serious challenge. Classical interconnection architectures, such as busses or crossbar switches, cannot offer the necessary flexibility and scaling with respect to throughput or area overhead. Network-on-chip (NoC) evolved as a flexible and high-performance solution for the interconnection problem during the last decade [4]. NoC is a packet-switched on-chip network where packets are forwarded from a source to a destination via several intermediate router nodes. We call the processing nodes that are connected to the NoC cores, modules or processing elements (PEs). Their functionality is thereby transparent to the NoC, i.e., this could be a processors, memory or an external interfaces. The smallest unit, to be transmitted over a NoC, is called the *flit* (flow control digit).

Finding an optimal NoC interconnect for many-core SoCs is a very challenging task, since many different design objectives and constraints have to be considered, like choosing routing and switching methods, selecting topology, application mapping, etc. [5]. This leads to a huge design space. Therefore, fast and accurate NoC models will be required that give an insight into the system and enable us to reduce the design space already in early design stages. Cycle-accurate simulation based approaches are too slow for this purpose. Simple high-level system models (e.g. only considering the propagation latency and ignoring queueing delays), on the other hand, are able to provide results in very short time. Due to the high abstraction, however, these models loose quite some accuracy. Analytic models provide a good trade-off between both approaches and are thus well suited for the NoC exploration of a many-core SoC.

In this paper, we propose an analytic NoC model based on queueing theory [6] that provides a high degree of flexibility regarding topology, routing and traffic scheme. In contrast to existing models, it is not restricted to mean value analysis but provides information about the state distribution functions of the routers. It enables us to easily derive arbitrary performance metrics, such as mean latency, buffer usage or blocking probability, and makes the model a very flexible tool for NoC performance analysis.

The remainder of this paper is structured as follows. In Section II, we discuss related work. Section III shows the system model and its assumptions. Then, Section IV introduces the analytic NoC model on network level (IV-A) and router level (IV-B). We evaluate the accuracy of the proposed approach against cycle-accurate simulation in Section V. Finally, Section VI concludes the work.

II. Related Work

Much effort has been spent for more than two decades for finding adequate traffic models for the analysis of off-chip and (later) on-chip networks. In 1990, Dally [7] developed analytic tools for investigating latency and throughput in networks, but restricting to k-ary n-cube topologies. Recent approaches focus on the mean value analysis of latency, throughput and energy consumption. Kiasari et al. presented an M/G/1 queueing model for wormhole switched two-dimensional (2D) torus NoC topologies, assuming deterministic routing [8]. A different approach has been published in 2009 in [9], which introduces an empirical model to estimate contention delays for constant service time routers. Thereby, the hybrid router model takes into account Poisson input flows as well as output flows from preceding constant service time routers. Ogras et al. presented a fast and flexible analytic approach in 2010 [10] for the mean value performance analysis of virtual channel first-come first-serve (FCFS) input buffered routers for arbitrary topology and service time

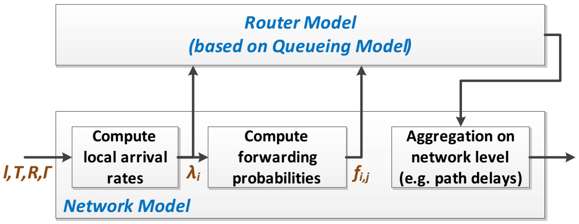

Router Model
(based on Queueing Model)
Compute local arrival rates
Compute forwarding probabilities
Aggregation on network level (e.g. path delays)
l,T,R,Γ
λi
fi,j
Network Model

Figure 1. The Hierarchical structure of the proposed analytic model.

distribution. Other recent approaches for modeling on-chip networks [11] focus on the theory of the Network Calculus [12]. This theory provides a powerful tool for an estimation of performance bounds in NoCs, which is essential for giving statements about the realtime capabilities of a network in early design stages. However, for the exploration of the average network behavior, other methods, like stochastic models, are more expedient.

III. System Model

We assume the routers to be arranged in an arbitrary topology. An arbitrary number of cores is allowed tobe connected to a single router. Due to the low buffer requirements, wormhole switching is the most favored switching technique for realizing best-effort services in on-chip networks today [5]. Therefore, we restrict our model to this technique. The routing protocol, on the other hand, shall not be restricted. Concerning the arbitration scheme, we restrict to the first-come first-serve method. Extensions to other arbitration schemes, like the popular round-robin, are left for future work. Routers consist of an arbitrary number of buffered input ports and an arbitrary number of (unbuffered) output ports. We assume infinite buffer size.

Furthermore, we assume external packet arrivals from PEs to possess Poisson characteristic [6], i.e., they have exponentially distributed inter-arrival times with known mean values. This assumption is often made to approximate real network traffic while reducing the model complexity at the same time. The router service times include processing delay for arbitration as well as forwarding delay for the packet and are assumed to be exponentially distributed. Furthermore, knowledge of the mean router service rate and router service latency is required. We assume it w.l.o.g. to be equal for all routers in the network. Finally, we imply a common clock for all routers.

IV. An analytic Model for Networks-on-Chip

To provide a flexible as well as a fast analytic model we propose to follow a hierarchical approach as depicted in Figure 1. We split the NoC model into an analysis on network level and on router level. By performing the analysis on router level and combining the results on network level, we thus reduce the complexity.

The network model receives multiple inputs that have to be specified by the user. The traffic scenario is described by the *traffic characterization matrix* $\boldsymbol{T}$ and the *external arrival rate vector* $\boldsymbol{l}$. The topology and interconnection is specified via the *connectivity matrix* $\boldsymbol{\Gamma}$. Finally, information about the applied routing scheme is provided via the *routing matrix* $\boldsymbol{R}$. An overview of the notation and a more detailed explanation is given in Table I.

Based on this information, the network model is able to compute local parameters for each router node individually, i.e., the inputs for the router model. The local parameters comprise the local arrival rates λ_i that is the accumulated arrival rate over all traffic flows that cross router input i. Furthermore, the forwarding probabilities $f_{i,j}$ are computed. $f_{i,j}$ defines the probability that a packet arriving at a router input i is forwarded to a router output j (please note that the indices i and j correspond to the unique identifier of the link that is connected to router input or output). The computation of the local arrival rates and forwarding probabilities is discussed in more detail in Section IV-A.

The local parameters can now be applied to a queueing model on router level. It is responsible for deriving the compound distribution for the number of packets in the input queues, which represent the router state. Consequently, the knowledge of the compound distribution enables a computation of key performance indicators, such as average buffer usage, blocking probabilities or mean queueing delays. The queueing model on router level is introduced in Section IV-B.

Finally, the performance metrics, computed on router level, have to be combined on network level, e.g., to derive path delays by summing up the queueing delays and the fix router propagation latencies.

A. Analysis on Network Level

We can derive the vector of local arrival rates $\boldsymbol{\lambda}$, with elements λ_i $(1 \leq i \leq N_E)$, by summing up all traffic flows that cross a specific link (and router input queue, respectively). Therein, N_E is the number of links in the network. The traffic characterization matrix $\boldsymbol{T}$ provides information about a pairwise traffic flow probability between each module s and d. By weighting $\boldsymbol{T}$ with the external arrival rates $\boldsymbol{l}$, we get the traffic intensities (in packets/cycle) for each pair of modules. Finally, we multiply the traffic intensities with the probability that the flow will pass link i (given by routing matrix $\boldsymbol{R}$) and sum up the fractions of the contributing traffic flows:

$$\lambda_i = \sum_{s=1}^{N_M} \sum_{d=1}^{N_M} l_s \cdot t_{s,d} \cdot r_{s,d,i}, 1 \leq i \leq N_E. \quad (1)$$

The notation is given in Table I. By applying the definition of the Frobenius inner product [13], we can rewrite (1) as matrix equation as follows:

Table I: Model parameters and notation

N_M	Number of modules
N_R	Number of router nodes
N_E	Number of edges
$\boldsymbol{T} = [t_{s,d}]$	Traffic characterization matrix (of size $N_M \times N_M$) with elements $t_{s,d}$ that specify the send probability from module s to module d
$\boldsymbol{l} = [l_s]$	External arrival rate vector (of size $N_M \times 1$) with elements l_s representing the arrival rate (packets/cycle) from source module s
$\boldsymbol{\Gamma} = [\gamma_{s,d}]$	Connectivity matrix (of size $(N_M+N_R)\times(N_M+N_R)$) with elements $\gamma_{s,d}$; $\gamma_{s,d} > 0$, if there is a directed connection from s to d; the value $\gamma_{s,d}$ represents the link ID for this connection (sgn($\boldsymbol{\Gamma}$) ≡topology matrix)
$\boldsymbol{R} = [r_{s,d,i}]$	Routing matrix (of size $N_M \times N_M \times N_E$) with elements $r_{s,d,i}$ defines the probability that link i is occupied for routing a packet from source module s to target module t ($\sum_{\forall i} r_{s,d,i} = 1$)
$\overline{x}$	Average router service time

$$\lambda_i = \mathrm{tr}\left(\left(\boldsymbol{T} \cdot \boldsymbol{L^D}\right)^T \boldsymbol{R_i}\right). \tag{2}$$

In (2), tr represents the trace of the matrix, $\boldsymbol{L^D}$ is the $N_M \times N_M$ diagonal matrix representation of vector $\boldsymbol{l}$:

$$\boldsymbol{L^D} := \mathrm{diag}(\boldsymbol{l}),$$

and $\boldsymbol{R_i}$ the corresponding submatrix of $\boldsymbol{R}$ that consists of all elements $r_{s,d,i}$ with $1 \leq s, d \leq N_M$. We can select the set of local arrival rates Λ^r for a single router node r by exploiting the knowledge of the topology that is contained in the connectivity matrix $\boldsymbol{\Gamma}$. I.e. we collect all λ_i where i is the ID of an input edge of router r:

$$\Lambda^r := \{\lambda_i \mid \exists s; 1 \leq s \leq N_M + N_R; \gamma_{s,r} = i\}. \tag{3}$$

We continue to compute the forwarding probability matrix $\boldsymbol{F}$. The matrix element $f_{i,j}$ $(1 \leq i, j \leq N_E)$ can be defined as traffic intensity between router input i and router output j normalized to the total arrival rate at input i, i.e., λ_i:

$$f_{i,j} := \frac{\sum_{s=1}^{N_M} \sum_{d=1}^{N_M} l_s \cdot t_{s,d} \cdot r_{s,d,i} \cdot r_{s,d,j} \cdot \delta_{i,j}}{\lambda_i}, 1 \leq i, j \leq N_E. \tag{4}$$

We call the term $\delta_{i,j}$ the *link selector matrix*. It ensures that there is only a forwarding probability $f_{i,j} > 0$, if (i, j) represents an input/output link pair of the same router:

$$\delta_{i,j} := \begin{cases} 1, & if\ \exists s, r, d\ with\ \gamma_{s,r} = i\ \wedge \gamma_{r,d} = j \\ 0, & otherwise \end{cases}.$$

Therein, $\gamma_{s,r}$ and $\gamma_{r,d}$ are corresponding elements of the connectivity matrix $\boldsymbol{\Gamma}$. Equation (4) can be rewritten in matrix form:

$$f_{i,j} := \frac{\delta_{i,j}}{\lambda_i} \cdot \mathrm{tr}\left(\left(\boldsymbol{T} \cdot \boldsymbol{L^D}\right)^T \left(\boldsymbol{R_i} \circ \boldsymbol{R_j}\right)\right), \tag{5}$$

where $\circ$ represents the entry-wise multiplication (i.e., the Hadamard product) of two matrices. Finally, we also restrict the set of forwarding probabilities F^r to a single router node r, similar to the approach in (3), and come to (6):

$$F^r := \{f_{i,j} \mid \exists s, d; 1 \leq s, d \leq N_M + N_R; \gamma_{s,r} = i \wedge \gamma_{r,d} = j\}. \tag{6}$$

B. An Analytic Router Model based on Queueing Theory

Based on the assumptions that we made in Section III, an M/M/1 queueing system [6] with exponential interarrival and service times will be appropriate to model the router behavior. However, in reality, the traffic situation within a router looks more complicated, as the example in Figure 2 (left) shows.

Therein, we find splitting and merging of traffic flows that contend with other input queues for multiple output ports. Furthermore, each input has different probabilities of being forwarded to a specific output. To be able to represent the router system by a queueing model, we propose using a simplified equivalent system, as depicted in Figure 2 (right). The idea is to include the contention delays into the service times and thereby receiving port specific service times. In fact, if a packet in front of a (FIFO) queue is blocked due to a contending queue, this is nothing else than a delayed service. Therefore, it is reasonable to consider the contention delay as a service time increase. Consequently, we come to a reduced equivalent system that now only consists of a single output server and multiple input queues, each having an individual service time (and service rate μ_i respectively).

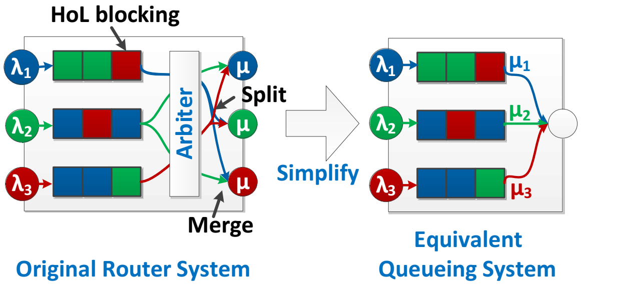


Figure 2. The equivalent queueing system simplifies a view onto the traffic situation in a router and can easily be expressed as a Markov model.

Due to the memoryless property of the exponentially distributed arrival and service processes, the state of the equivalent router system can now solely be defined by the number of flits contained in the input queues. If we represent the state by a vector where each element represents the fill level of a single input queue, we can model the system by means of a multidimensional Markov chain. This is illustrated in Figure 3 for the case of a router with two inputs (please ignore the depicted macro states for now). Therein, the transition rates are defined by the arrival rate λ_i and service rate μ_i for each input independently. Let $\boldsymbol{x}$ be the current state vector of the router. Then, a transition from state $\boldsymbol{x} \rightarrow \boldsymbol{x} + e_i$ (where e_i is the unit vector for dimension i) has an intensity of λ_i. On the other hand, a transition $\boldsymbol{x} \rightarrow \boldsymbol{x} - e_i$ has an intensity of μ_i. The boundaries of the Markov chain are an exception to that rule (first column/row in Fig. 3). There, we find a different contention situation. In the case of two inputs, we have no contention caused by the second input anymore. Therefore, the transition rates for $\boldsymbol{x} \rightarrow \boldsymbol{x} - e_i$ change to μ, i.e., the basic router service rate without contention delay.

For solving the Markov chain, we still need to know the

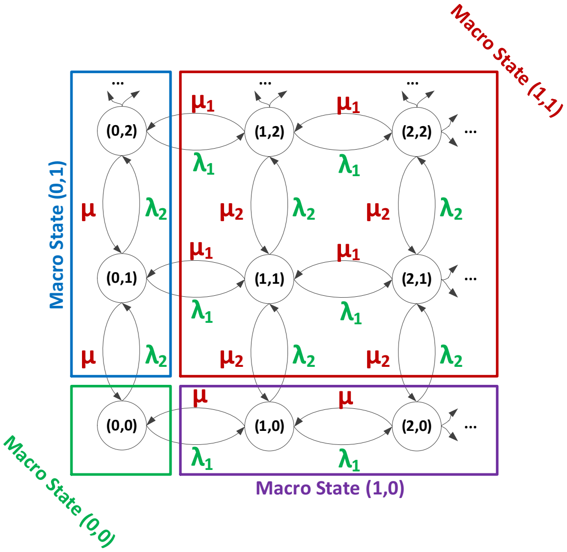


Figure 3. Example of a two-dimensional Markov model for a router with two inputs and the decomposition into reversible sub-chains.

service rates μ_i that include the contention delay to be able to define the transition rates. For this purpose, we apply an idea that was proposed in [10] to determine the mean waiting time for a similar input buffered router model assuming an FCFS arbiter. We modify this approach to find an estimation for the mean service time, i.e., the waiting time of the flit in front of the queue. Similar to [10], we first compute the pairwise contention probability $c_{i,j}$ for all inputs pairs (i, j) of a router with P inputs based on the forwarding probabilities $\boldsymbol{F}$ that can be derived according to (5):

$$c_{i,j} = \sum_{k=1}^{P} f_{i,k} f_{j,k}, i \neq j, 1 \leq i, j \leq P. \tag{7}$$

From (7), an equivalent matrix equation can be derived

$$\boldsymbol{C} = \boldsymbol{F} \cdot \boldsymbol{F}^T. \tag{8}$$

Note that the main diagonal of the contention probability matrix $\boldsymbol{C}$ in (8) is set to "1" which makes the following computation more convenient. Based on the contention probabilities, we can derive an expression to estimate the mean service times $\overline{x}_i(\boldsymbol{y})$ under contention:

$$\overline{x}_i(\boldsymbol{y}) := \overline{x} + \overline{x} \sum_{j=1, j\neq i}^{P} c_{i,j} y_j, 1 \leq i \leq P. \tag{9}$$

The first summand $\overline{x}$ of (9) represents the mean router service time for the packet in front of queue i. The second summand considers the contention delay. Therein, the vector $\boldsymbol{y}$ represents the instantaneous fill state of each input queue, i.e., $y_i = 0$, if input queue i is empty and does not contribute to the contention delay and $y_i = 1$ otherwise. We will call $\boldsymbol{y}$ the *router macro state* in the following and can directly derive it from the router state $\boldsymbol{x}$:

$$y_i = \begin{cases} 0, & if\ x_i = 0 \\ 1, & if\ x_i > 0 \end{cases},$$

or rather informally: $\boldsymbol{y} = \operatorname{sgn}(\boldsymbol{x})$.

We can still condense (9) somewhat by exploiting the convenient definition of contention probability matrix $\boldsymbol{C}$ and provide a short form matrix equation for the mean service rates $\mu_i(\boldsymbol{y})$ (i.e. the inverse of the mean service times):

$$\mu_i(\boldsymbol{y}) := \left[\frac{1}{\mu} \boldsymbol{C}_i{}^T \boldsymbol{y}\right]^{-1}, 1 \leq i \leq P. \tag{10}$$

With the definition for the mean service rates $\mu_i(\boldsymbol{y})$ in (10) we have now all necessary inputs to solve the Markov chain in order to obtain the steady-state probability distribution. However, in trying to do so, we are confronted with another challenge. If we apply the Kolmogorov criterion for reversibility of Markov chains, we soon realize that it does not hold for some cases in the peripheral region of our Markov chain. Accordingly, the chain is not time reversible; see Fig. 3 and examine the following state transitions: $(0, 0) \rightarrow (1, 0) \rightarrow (1, 1) \rightarrow (0, 1) \rightarrow (0, 0)$, and the corresponding return path. We notice that the product of the transition rates is not equal for both directions, and thus, it does not fulfill the Kolmogorov criterion [14]:

$$\lambda_1 \cdot \lambda_2 \cdot \mu_1 \cdot \mu \neq \lambda_2 \cdot \lambda_1 \cdot \mu_2 \cdot \mu.$$

Consequently, we are not allowed to apply local balance equations to solve the chain. Unfortunately, we are not able to find a closed-form solution for the infinite Markov chain solely based on the global balance equations. Fehske and Fettweis [15] recently encountered exactly the same problem when trying to solve an equivalent Markov chain. They proposed an approximation to find a solution for the stationary distribution. The approach is based on the concept of *aggregation of variables* that is well known by economics for quite some years [16]. The proposed algorithm consists of the following steps.

We start decomposing our Markov chain into reversible sub-chains. This is done by collecting all states $\boldsymbol{x}$ that belong to the same macro state (or aggregate state) $\boldsymbol{y} = \operatorname{sgn}(\boldsymbol{x})$ in a common set $\mathcal{S}(\boldsymbol{y})$:

$$\mathcal{S}(\boldsymbol{y}) := \left\{\boldsymbol{x} \in \mathbb{N}_0^P \mid \operatorname{sgn}(\boldsymbol{x}) = \boldsymbol{y}\right\}.$$

The idea behind the definition is that all states are collected in the a macro state where we find a similar contention situation. If we consider a contending queue, it doesn't matter how many packets it contains, only if it contains at least one packet or not. Consequently, the mean service rates are homogeneous within each macro state. An example for the Markov chain decomposition for the case of two input ports is provided in Figure 3. Therein, we decompose the two-dimensional Markov chain into four macro states. Macro state $(0, 0)$ contains all states where both input queues are empty (which is only a single router state $(0, 0)$). Macro states $(1, 0)$ and $(0, 1)$ collecting the states where only one of the two queues is empty. Hence, we have no contention within these two macro states. Macro state $(1, 1)$ represents all router states where both queues are not empty. In this example, this is the only macro state where contention occurs.

Since the transition rates are homogeneous within each macro state, the sub-chains are reversible and can be solved. This leads to a product form solution for the stationary probability distribution of the number of customers (i.e. packets) $\widetilde{\pi}$ in an M/M/1 queueing system that is well known from classical queueing theory [6][15]:

$$\widetilde{\pi}(\boldsymbol{x}) = \begin{cases} \prod_{i\in N_1(\boldsymbol{y})} (1 - \rho_i(\boldsymbol{y}))\rho_i^{x_i-1}(\boldsymbol{y})\sigma(\boldsymbol{y}), & \text{for } \boldsymbol{y} \neq \boldsymbol{0} \\ \sigma(0), & \text{for } \boldsymbol{y} = \boldsymbol{0} \end{cases} \tag{11}$$

with utilization $\rho_i(\boldsymbol{y})$ of input queue i defined as

$$\rho_i(\boldsymbol{y}) := \frac{\lambda_i}{\mu_i(\boldsymbol{y})}.$$

Note that (11) only yields an estimate for the solution of the stationary probability distribution. This is because we omit the transitions between the macro states at this consideration. Also, note that (11) is conditioned on the probabilities of the corresponding macro state $\sigma(\boldsymbol{y})$ to ensure that $\sum_{\boldsymbol{x}} \widetilde{\pi}(\boldsymbol{x}) = 1$.

So far, we have no knowledge about the macro state probabilities $\sigma(\boldsymbol{y})$. We can compute $\sigma(\boldsymbol{y})$ by solving the (now finite) Markov chain on macro state level. Figure 4 shows a solution for the transition rate $p(\boldsymbol{y}, \boldsymbol{y}')$ from macro state $\boldsymbol{y}$ to macro state $\boldsymbol{y}'$, as provided by [15]:

$$p(\boldsymbol{y}, \boldsymbol{y}') = \begin{cases} \lambda_i, & \text{for } \boldsymbol{y}' = \boldsymbol{y} + e_i \\ \mu_i(\boldsymbol{y}) - \lambda_i, & \text{for } \boldsymbol{y}' = \boldsymbol{y} - e_i \\ 0, & \text{else} \end{cases}, \tag{12}$$

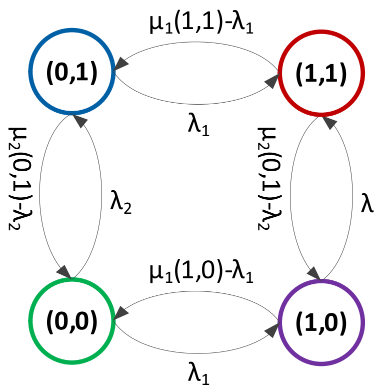


Figure 4. Example: Markov chain on macro state level assuming a router with two inputs.

where e_i again represents the unit vector for dimension i. Based on (12), we can now define the transition probability matrix $P = \left[p_{ij}\right]$ with $p_{ij} := p\left(y_i, y_j\right)$. With the definition of $p_{ii} := -\sum_{j=1}^{2^P} p_{ij}$ we normalize the row sum to 0.

Finally, we can follow the usual approach and solve the equation system for the vector of macro state probabilities σ based on the transition probability matrix P:

$$\sigma P = 0,$$

under the side condition $\sum_{\boldsymbol{y}} \sigma(\boldsymbol{y}) = 1$.

Based on (11), we can now compute the approximates for the state probabilities $\widetilde{\pi}(\boldsymbol{x})$. We can derive several key performance indicators, such as the mean number of packets in the queue $\mathbb{E}[x_i]$:

$$\mathbb{E}[x_i] \approx \sum_{\boldsymbol{x}} \widetilde{\pi}(\boldsymbol{x}) x_i = \sum_{\boldsymbol{y}} \frac{\rho_i(\boldsymbol{y})}{1 - \rho_i(\boldsymbol{y})} \sigma(\boldsymbol{y}),$$

or the mean queueing delay W_i for input queue i by applying Little's law [6]:

$$W_i = \frac{\mathbb{E}[x_i]}{\lambda_i}.$$

V. Performance Evaluation

We show the accuracy of the proposed NoC model by comparing it against cycle-accurate NoC simulation. Due to the similar system model assumptions we decided to compare our approach against the model proposed in [10] as well as the NoC simulation tool that has been used therein [17].

We assumed following common simulation parameters:

- deterministic, dimension-ordered XY-routing,
- flit traffic, i.e., packet size = 1,
- input buffered routers with FCFS arbiter and service rates of $\mu = 0.5$,
- large buffer size (256 flits) to approximate the infinite buffer model and
- simulation run time of 10^5 cycles with a warm-up period of 10^4 cycles.

We investigate the following two topology/traffic scenarios under different load conditions (defined by number of injected packets/cycle) and compare the average packet transmission latency in the network.

First, we choose a very simple scenario to investigate the model behavior under a clear contention situation. Therefore, we consider a simple chain of four routers where a single PE is connected to each router. The PEs at routers 1 and 4 are sending their packets to PE 2 and 3 with a uniform distribution. PEs 2 and 3 do not send any packets. Hence, we find at router 2 and 3 a contention situation with the following forwarding probability matrix $\boldsymbol{F}$:

$$\boldsymbol{F} = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 0 & 0.5 \\ 1 & 0 & 0 \end{pmatrix}.$$

The result under different load conditions is shown in Figure 5. We find that the latency estimation for our proposed approach (red curve with + marker) follows very well the cycle-accurate simulation results (black curve with point marker) under a low and medium load condition. However, it significantly underestimates the network saturation limit where latency tends to infinity (0.66 packets/cycle in our model compared to 0.8 packets/cycle in the cycle-accurate simulation). The reference mean value model from [10] (blue curve with circle marker) shows a slight overestimation of the latencies under mid load conditions but estimates the network saturation point quite well.

The reason for the poor estimation of the network saturation point of our model is the applied aggregation approach for approximating the solution of a Markov chain. Therein, the stability of the overall solution is determined by the stability of the "worst-case" aggregate, i.e., the aggregate with the highest contention. If the solution for the "worst-case" aggregate tends to infinity the overall solution tends to infinity as well. To avoid this behavior, we propose to determine an average service time $\overline{\overline{x}}_i$ over all macro states for every router input. This is done by computing the expectation of the mean service times $\overline{x}_i(\boldsymbol{y})$ over all macro states based on the known macro state probabilities $\sigma(\boldsymbol{y})$:

$$\overline{\overline{x}}_i = \sum_{\boldsymbol{y}\in\{0,1\}^P} \overline{x}_i(\boldsymbol{y})\sigma(\boldsymbol{y})y_i. \qquad (13)$$

Therein, y_i constrains the expectation to those macro states where queue i is not empty. We compute the average waiting time W_i for input queue i based on (13):

$$W_i = \frac{\overline{\overline{x}}_i}{1 - \lambda_i \overline{x}_i}.$$

The result of the refined approach is also depicted in Figure 5 (green curve with square marker). It shows a very good match compared to the cycle-accurate simulation. The latencies under low/mid load conditions, as well as the network saturation point, are estimated very accurately by this approach. The average estimation error is less than 3%.

Finally, we choose a 4x4 2D-mesh topology using a more diverse traffic pattern of the generic multimedia application from [10]. We target to compare the estimation quality of the average latencies under more complex contention situations. The results are plotted in Figure 6 and confirm the accurate results of the first scenario. Again, the average estimation error is around 3% (9% for the reference model). However, we still notice a slight underestimation of the network saturation limit of about 2.5% for that case. The reference mean value model shows a better accuracy in this region.

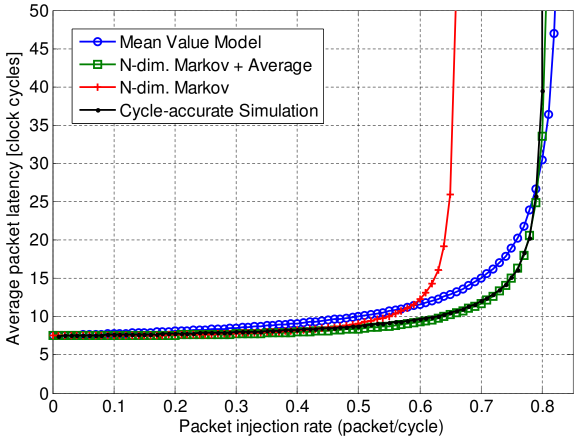


Figure 5. Performance results for 4x1 chain analyzing the average packet latency in comparison to cycle-accurate simulation.

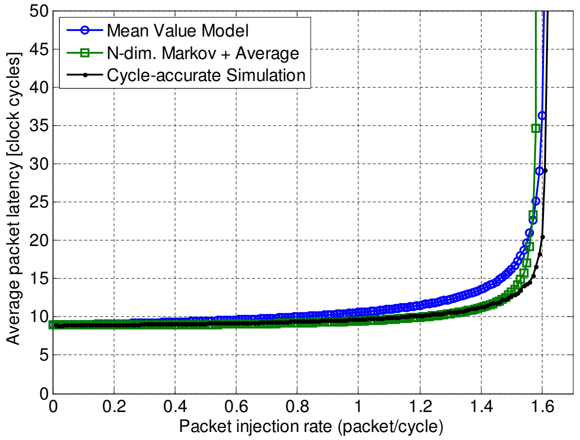


Figure 6. Performance results for 4x4 2D-mesh with generic multimedia application traffic analyzing the average packet latency in comparison to cycle-accurate simulation.

Note that the presented results only serve as proof of concept and easily scale to larger networks. The relative accuracy of the latency estimation is expected to stay in the same range under similar contention situations, independent of the NoC size. This is because the analysis of the queueing delay is done on router level and only accumulated on network level.

VI. Conclusion and Future Work

In this paper, we presented a novel analytic approach for modeling on-chip networks for many-core SoC based on queueing theory. In contrast to many existing models, the approach is very flexible in terms of supported topology, routing scheme and traffic pattern. The approach overcomes the limitations of the mean value analysis introduced in the existing work. Instead, it provides information about a steady-state distribution of the network routers. This allows to derive arbitrary key performance indicators, such as blocking probabilities or average queueing delays, which is very important information for dimensioning network resources, such as buffers, links, etc. We demonstrated the very high accuracy of the approach by comparison to a cycle-accurate simulation. The average estimation error for the mean latencies in a 4x4 2D-mesh is only 3%.

Many extensions of the NoC model are planned. We target to consider different arbitration schemes, such as the popular round-robin method. A finite buffer model extension would be interesting in order to model network acceptance behavior and back pressure effects. A generalization towards an arbitrary service time distribution is also desirable. Finally, supporting multiple clock domains (i.e., globally asynchronous locally synchronous systems) and frequency scaling is another open topic in order to explore a many-core NoC more accurately.

References

[1] T. Dresden, “Esf young investigators group; 3d chip stack intraconnects - 3dcsi,” last visited on 15/10/2012. [Online]. Available: http://tu-dresden.de/die_tu_dresden/fakultaeten/fakultaet_elektrotechnik_und_informationstechnik/3dcsi

[2] J. Manferdelli, N. Govindaraju, and C. Crall, “Challenges and opportunities in many-core computing,” *Proc. of IEEE*, vol. 96, no. 5, pp. 808 –815, May 2008.

[3] S. Borkar, “Thousand core chipsa technology perspective,” in *Proc. of DAC*, 2007.

[4] L. Benini and G. De Micheli, “Networks on chips: a new soc paradigm,” *Computer*, vol. 35, no. 1, pp. 70 –78, Jan 2002.

[5] R. Marculescu, U. Ogras, L.-S. Peh, N. Jerger, and Y. Hoskote, “Outstanding research problems in noc design: System, microarchitecture, and circuit perspectives,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 28, no. 1, pp. 3 –21, Jan. 2009.

[6] L. Kleinrock, *Queueing systems - 1 : Theory*. New York: Wiley, 1975.

[7] W. Dally, “Performance analysis of k-ary n-cube interconnection networks,” *Computers, IEEE Transactions on*, vol. 39, no. 6, pp. 775 –785, Jun 1990.

[8] A. Kiasari, D. Rahmati, H. Sarbazi-Azad, and S. Hessabi, “A markovian performance model for networks-on-chip,” in *Proc. of Euromicro PDP*, 2008.

[9] N. Nikitin and J. Cortadella, “A performance analytical model for network-on-chip with constant service time routers,” in *Proc. of ICCAD*, 2009.

[10] U. Ogras, P. Bogdan, and R. Marculescu, “An analytical approach for network-on-chip performance analysis,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 29, no. 12, pp. 2001 –2013, Dec. 2010.

[11] M. Bakhouya, S. Suboh, J. Gaber, and T. El-Ghazawi, “Analytical modeling and evaluation of on-chip interconnects using network calculus,” in *Proc. of NoCS*, 2009.

[12] J.-Y. Le Boudec and P. Thiran, *Network calculus: a theory of deterministic queueing systems for the internet*. Berlin, Heidelberg: Springer-Verlag, 2001.

[13] Seber and A. F. George, *A Matrix Handbook for Statisticians*. John Wiley & Sons, Inc., 2008.

[14] R. Nelson, *Probability, stochastic processes, and queueing theory / the mathematics of computer performance modeling*. New York ; Heidelberg [u.a.]: Springer, 1995.

[15] A. J. Fehske and G. P. Fettweis, “Aggregation of variables in load models for interference-coupled cellular data networks,” in *Proc. of ICC*, 2012.

[16] H. A. Simon and A. Ando, “Aggregation of variables in dynamic systems,” *Econometrica*, vol. 29, no. 2, pp. 111–138, Apr 1961.

[17] worm_sim, “Cycle-accurate noc simulator,” last visited on 15/10/2012. [Online]. Available: http://www.ece.cmu.edu/~sld/software/index.php

Multi-objective Linear Programming Optimization for Waste Management Simulation

Eric Solano
RTI International
RTP, North Carolina, USA
solano@rti.org

Abstract—**Municipal Solid Waste Management (MSWM) is a very complex problem present in many communities around the world. Decision makers need to formulate solutions that consider multiple goals and strategies. Strategies include multiple options for waste collection, transportation, transfer, treatment and disposal. The most appropriate choice, however, is often not clear. Given the large number of available options for MSWM and the interrelationships among these options, identifying MSWM strategies that satisfy economic or environmental objectives is a complex task. The main objective of this work is to use MSWM simulation and multi-objective linear programming to support decision makers in the process of selecting MSWM strategies from a very large decision space and in the presence of multiple objectives. Three competing objectives are considered: least cost, minimization of carbon dioxide emissions and minimization of energy consumption. A multi-objective fuzzy linear programming method is proposed to manage imprecision and uncertainty in the objectives via fuzzy membership functions.**

Keywords-simulation model; linear programming; fuzzy multiobjective optimization; Pareto optimal set; waste management.

I. INTRODUCTION

Municipal Solid Waste Management (MSWM) is getting increased attention at national and local levels. The specific goals of each community for implementing MSWM plans depend on site-specific conditions and issues. For instance, a community facing a landfill space crisis may set a goal to reduce the amount of waste sent to landfill disposal and may consider source reduction, waste diversion through recycling, and volume reduction alternatives such as converting waste to energy. The most appropriate choice, however, is often not clear. For instance, recycling is known to reduce consumption of natural resources and save some processing activities at manufacturing facilities. These savings will avoid the emissions of some associated greenhouse gases and pollutants. However, if the market prices of recyclable materials are low, then a recycling program may not be as economical as one of the other options. To add complexity to this problem, landfill space may be very limited, making recycling an attractive option regardless of low market prices for recyclable materials. Each step in waste management (collection, recycling, treatment, disposal) could be accomplished through different technological options, and an overall MSWM overall strategy should include in most cases at least one technological option for each step. Decision makers are then faced with a problem of multiple dimensions: they need to select from multiple technological options to manage municipal waste from the generation point to the final disposal point to create an overall strategy, and they must evaluate each overall strategy for the competing objectives of cost effectiveness and environmental impact reduction. Given the large number of available options for MSWM and the interrelationships among these options, identifying MSWM strategies that satisfy economic or environmental objectives is a complex task. Simulation of MSWM is used in this work to help the decision makers with the screening and identification of MSWM strategies.

The main objective of this work is to use MSWM simulation and multi-objective linear programming to select MSWM strategies from a very large decision space. Multiple objectives are considered: least cost, minimization of carbon dioxide emissions and minimization of energy consumption. Least cost solutions will tend to select MSWM technological options that do not necessarily reduce emissions or energy consumption. Conversely, minimum energy consumption scenarios and minimum pollutant emission scenarios will tend to select more advanced and expensive MSWM technological options. A number of methodologies have been proposed for multi-objective linear programming, such as the ones reported in [4], [5], [6], [7], [8], [9], and [10]. None of these methodologies offers a simple way to evaluate the relative importance of objective functions with very dissimilar measurement units. A new methodology using MSWM simulation and multi-objective fuzzy linear programming is proposed to improve the traditional weighted sum methodology. It is used to select noninferior solutions and to quantify the degree of achievement for each of the competing objectives.

II. METHODOLOGY

A. *Municipal Solid Waste Management Simulation*

Simulation of MSWM can be performed by using a systemic representation of the different MSWM options. Each MSWM option or node has a specific purpose (e.g.

waste collection, waste transfer or waste treatment) and is inter-related with a number of other MSWM options. Waste processes are simulated as mass balances at each management option, where there is incoming waste from other nodes and outgoing waste to other nodes. This simulation includes a number of sub-models, one for each MSWM option. These sub-models use the Life Cycle Inventory methodology to calculate annualized cost, energy consumption and emissions of different pollutants. The mathematical formulation is linear programming (LP) in which the objective functions are to minimize cost, energy consumption or a number of pollutant emissions. The simulation includes a life cycle inventory of different pollutants and the user can select to minimize the emissions of any of these pollutants. It also includes a complex mathematical formulation via constraints to represent waste mass flows from process units to other process units. For more details on the model simulation design and mathematical formulation see references [1], [2] and [3]. Table 1 shows a list of MSWM options used in this study.

TABLE I. MUNICIPAL SOLID WASTE MANAGEMENT OPTIONS

MSWM step	*Management Option*
Collection	Residential Collection of Yardwaste
Collection	Residential Collection of Mixed MSW
Collection	Residential Collection of Commingled Recyclables Sorted by Crew
Collection	Residential Collection of Presorted Recyclables
Collection	Residential Collection of Commingled Recyclables Sorted at Materials Recovery Facility
MRF	Materials Recovery Facility (MRF) for Mixed MSW
MRF	Materials Recovery Facility (MRF) for Presorted Recyclables
MRF	Materials Recovery Facility (MRF) for Commingled Recyclables
Treatment	Treatment at Yardwaste Compost Plant
Treatment	Treatment at Mixed MSW Combustion Plant
Treatment	Treatment at Refuse Derived Fuel (RDF) Plant
Disposal	Disposal at Landfill
Disposal	Disposal at Ash-landfill

B. Generation of Baseline Solutions

A hypothetical case representing a residential urban region of medium size will be defined. The residential population is 400,000 people with a solid waste generation rate of 4.5 lbs/person-day. The MSWM system definition requires specification of many input parameters, e.g., waste composition, distances between waste processing facilities, collection frequencies, etc. Most of these input parameters use national average values. The model was applied to an illustrative problem scenario where 15% of overall generated waste was banned from being disposed at the landfill. Diverting waste to be disposed at the local landfill can be achieved via yardwaste composting and recycling programs. The diversion rate will be enforced by adding a diversion constraint in addition to the mass flow constraints. Single objective simulations were executed to obtain the minimum-cost solution, the minimum carbon dioxide emissions solution and the minimum energy consumption solution. The resulting MSWM options selected from the decision space were reported for each case. The resulting cost or environmental objective values selected from the objective space were reported for each case.

C. Generation of Pareto Optimal Set

The next step was to generate additional MSWM solutions using multi-objective programming. The three solutions obtained previously were generated by minimizing on one of the main objectives at a time. Since these objectives can be conflicting with one other, we are now interested in formulating a single aggregate objective function that incorporates all three objectives. This is done by creating a weighted linear sum of the objectives.

Equation 1 shows the weighted linear sum of the three objectives: cost, energy consumption and carbon dioxide emissions. This objective function is optimized subject to the mass flows constraints and the diversion constraint. The objective function weights w_{cost}, w_{CO2} and w_{enegy} are supplied to obtain different optional solutions. They dictate how much of one objective must be sacrificed for the benefit of the other objectives. All solutions obtained with this method constitute non-inferior solutions necessary to generate the Pareto Optimal Set.

$$\text{Min } \{w_{cost} \text{ x } Z_1 + w_{energy} \text{ x } Z_2 + w_{CO2} \text{ x } Z_3\}. \qquad (1)$$

Subject to: mass flow and waste diversion constraints.

where: Z_1 is Cost, Z_2 is Energy Consumption and Z_3 is CO_2 emissions. Weights w_{cost}, w_{CO2} and w_{enegy} are greater than zero and their sum is equal to 1.

D. Fuzzy Linear Programming

The last step was to use fuzzy linear programming to represent the objective functions as fuzzy sets. The use of fuzzy sets tries to capture the imprecision and uncertainty of competing objectives. A number of methodologies have been proposed for fuzzy multi-objective linear programming, such as the ones reported in [4], [5], [6], [7], [8], [9] and [10]. The method proposed here is a modification of the method proposed by Raju and Kumar [4]. The relative importance of the objectives is measured by the membership functions. The membership function is a measure of the degree of achievement for any given objective and is represented by $\mu_i(X)$ in Equation 2.

$$\mu_i(X) = \{ \begin{array}{l} 0, \text{ for } Z_i < Z_{L,i} \\ (Z_i - Z_{L,i})/(Z_{U,i} - Z_{L,i}), \text{ for } Z_i < Z_{L,i} \\ 1, \text{ for } Z_i > Z_{U,i} \end{array} \qquad (2)$$

ISBN: 978-1-61208-234-9

where:
$Z_{L,i}$: less desirable value for objective i
$Z_{U,i}$: most desirable value for objective i
Z_i: objective value linked to degree of achievement μ_i
X: decision variables vector

Equation 1 can be re-written as:

$$\text{Min } \lambda \tag{3}$$

Subject to:

$$\lambda = w_{cost} \text{ x } \lambda_1 + w_{energy} \text{ x } \lambda_2 + w_{CO2} \text{ x } \lambda_3$$
$$(Z_{cost} - Z_{L,cost})/(Z_{U,cost} - Z_{L,cost}) \leq \lambda_1$$
$$(Z_{energy} - Z_{L,energy})/(Z_{U,energy} - Z_{L,energy}) \leq \lambda_2$$
$$(Z_{CO2} - Z_{L,CO2})/(Z_{U,CO2} - Z_{L,CO2}) \leq \lambda_3$$
$$0 \leq \lambda_1 \leq 1,\ 0 \leq \lambda_2 \leq 1,\ 0 \leq \lambda_3 \leq 1$$

mass flow and waste diversion constraints

A value of 0 for λ_1 means that there was a perfect achievement to minimize the cost objective, and a value of 1 means that the worst achievement for cost was obtained. The same applies for λ_2 and the energy objective and for λ_3 and the CO_2 objective. The degree of achievement can be then defined as 1- λ_i. Under this modified formulation, weights w_{cost}, w_{CO2} and w_{enegy} have values greater than zero but do not need to add up to 1. They measure the objectives relative importance between one another.

III. RESULTS AND ANALYSIS

A. *Minimum Cost Solution*

The model simulation resulted in a minimum cost solution that includes yard waste collection and composting for a sector of the residential population (23,800 tons). It also includes collection of mixed MSW taken to a mixed MSW MRF for separation of recyclable materials (260,000 tons).

Recyclable materials sorted and processed (25,500 tons) at the mixed waste MRF include among others: old newspaper (10,500 tons), corrugated cardboard (3,280 tons), ferrous cans (2,350), plastic (782 tons), clear glass (6,100) and brown glass (2,500 tons). This amount of composted and recycled material helped comply with the mandatory 15% diversion policy goal. Fig. 1 shows the selection of technologies for the minimum cost solution. The minimum cost obtained was \$33 million, with -40 million pounds of CO_2 emissions and -0.18 trillion BTU of energy consumption. A negative value in CO_2 emissions indicates that CO_2 emissions generated by unit processes (35 million lbs) are offset by the remanufacturing of recyclable materials into other usable materials (avoided 32 million lbs). These remanufacturing processes use recyclable material instead of virgin materials. Avoided emissions from the extraction and processing of raw materials are accounted for in the life cycle inventory. Additional emission offsets are accounted for at the landfill where CO_2 is sequestered and avoided from entering the atmosphere (43 million lbs). Similarly, a negative value in the energy consumption means that energy consumed by the unit processes (0.27 trillion BTU) is offset by the remanufacturing process (avoided 0.29 trillion BTU). Avoided energy consumption from the extraction and processing of raw materials is accounted for in the life cycle inventory.

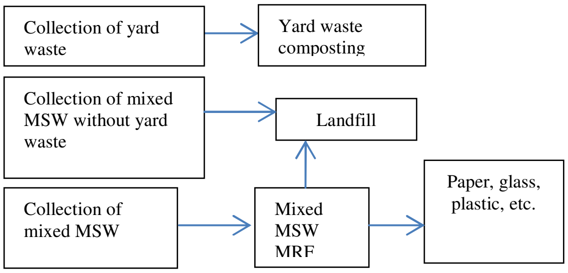


Figure 1. Management options for minimum cost.

B. *Minimum Carbon Dioxide Emissions Solution*

The minimum carbon dioxide emissions solution includes quite different options from the minimum cost solution. It includes collection of recyclables sorted by the collection crew and taken to a presorted recyclables MRF (29,200 tons). It also includes collection of mixed MSW taken to a mixed MSW MRF (299,000 tons). Residuals from the mixed MSW MRF are then taken to a waste to energy facility (280,000 tons). Recyclable materials sorted and processed at the mixed MSW MRF (18,800 tons) include among others: old newspaper (6,830 tons) and clear glass (4,160 tons); and recyclable materials sorted at the presorted recyclables MRF (29,200 tons) include among others: old newspaper (9,590 tons) and clear glass (5,250 tons). Fig. 2 shows the selection of technologies for the minimum carbon dioxide emissions solution. The minimum carbon dioxide obtained was -202 million pounds of CO_2 (offset), with a cost of \$57.6 million and -2.2 trillion BTU of energy consumption (offset).

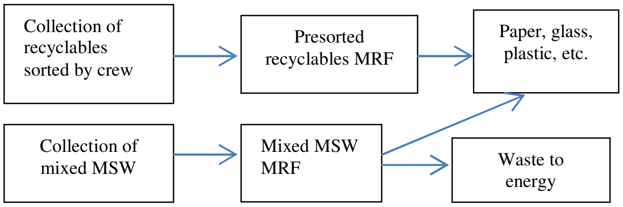

Collection of recyclables sorted by crew
Presorted recyclables MRF
Paper, glass, plastic, etc.
Collection of mixed MSW
Mixed MSW MRF
Waste to energy

Figure 2. Management options for minimum carbon dioxide emissions.

C. *Minimum Energy Consumption Solution*

The minimum energy consumption solution includes collection of commingled recyclables taken to a commingled recyclables MRF (27,600 tons). It also includes collection of mixed MSW taken to a mixed MSW MRF (301,000 tons). Residuals from the mixed MSW MRF are then taken to a waste to energy facility (280,000 tons). Recyclable materials sorted and processed at the commingled materials MRF (27,600 tons) include among others: old newspaper (9,590 tons) and clear glass (5,250 tons); and recyclable materials sorted at the mixed waste MRF (21,000 tons) include among others: old newspaper (6,830 tons) and clear glass (4,160 tons). Fig. 3 shows the selection of technologies for the minimum energy consumption solution. The minimum energy consumption obtained was -2.27 trillion BTU (offset), with a cost of $60.3 million and -198 million pounds of CO_2 emissions (offset).

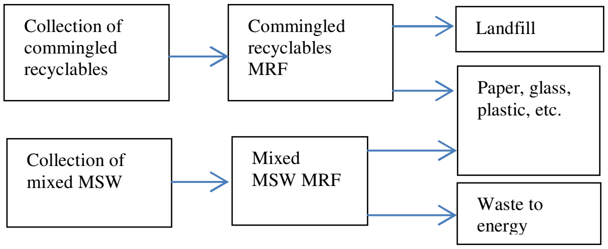

Collection of commingled recyclables
Commingled recyclables MRF
Landfill
Paper, glass, plastic, etc.
Collection of mixed MSW
Mixed MSW MRF
Waste to energy

Figure 3. Management Options for minimum energy consumption.

D. *Multi-objective Optimization*

Additional solutions were found by using multi-objective optimization and the weighted sum method. Table 2 shows the weight values used for each multi-objective formulation, and summarizes the values of the three objectives for each of the noninferior solutions. Weights were varied arbitrarily to try to capture different solutions. The optimal solution obtained depended on the relative values of the used weights. For example, if the specified weight for the cost objective was greater than the specified weight for the energy consumption objective, the solution favored lower cost over lower energy consumption.

Solution A corresponds to the least cost solution (Z1*); solution E to the minimum CO_2 emissions solution (Z3*); and solution F to the minimum energy solution (Z2*). The other solutions represent noninferior points in the multi-objective solution space.

TABLE II. SUMMARY OF SOLUTIONS FROM MULTI-OBJECTIVE OPTIMIZATION

Sol.	w_{cost}	w_{energy}	w_{CO2}	***Cost (10^6 $)***	***Energy (trillion BTU)***	***CO_2 (10^6 lbs)***
A	0.999	5e-4	5e-4	33.1 (Z1*)	0.018	39.8
B	0.8	0.1	0.1	36.4	1.207	137.0
C	0.6	0.2	0.2	45.6	2.070	187.0
D	0.4	0.3	0.3	52.1	2.203	196.0
E	0.2	0.4	0.4	57.6	2.205	202.4 (Z3*)
F	5e-5	0.9999	5e-5	59.9	2.270 (Z2*)	195.9

Fig. 4 shows a plot of the multi-objective solutions as a tradeoff between the conflicting objectives of least cost and minimum CO_2 emissions. Similarly, Fig. 5 shows a plot of the multi-objective solutions as a tradeoff between the conflicting objectives of least cost and minimum energy consumption. These tradeoff curves represent the Pareto optimal sets or frontiers for the conflicting objectives. Any point in the Pareto optimal set represents a noninferior solution, for which an improvement in one objective requires a degradation of the other. Energy and emissions are non-conflicting objectives.

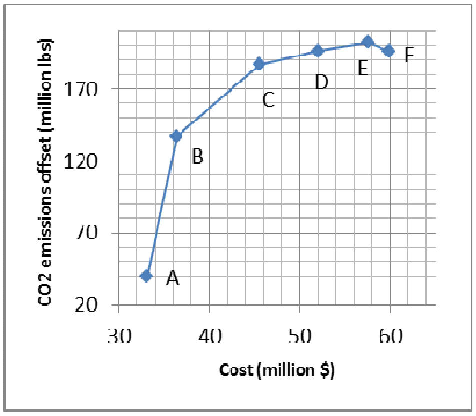


Figure 4. Cost and CO_2 emissions tradeoff curve.

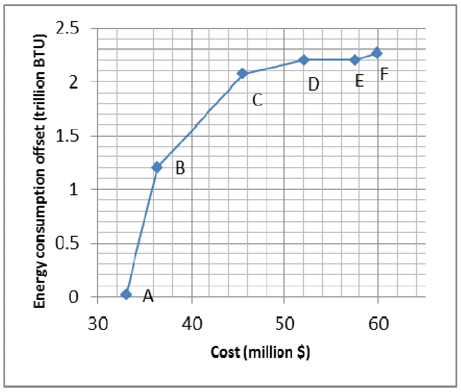


Figure 5. Cost and energy consumption tradeoff curve.

Solutions B, C and D are noninferior solutions in which none of the three objectives reaches its minimum possible value. These "intermediate" solutions try to accommodate all three objectives based on the provided objective weights. The combinations of MSWM options for these intermediate solutions are also different from the least cost, minimum energy and minimum CO_2 solutions. Fig. 6 shows the selected MSWM options for solution B. This solution is trying to depart from the least cost solution A and provides a greater weight to both the energy and emissions objectives. Therefore, the selection of technologies includes, in addition to the mandatory recycling, Refuse Derived Fuel (RDF) to try to offset energy consumption via the generation of energy from waste, and to avoid CO_2 emissions by reducing the amount of waste to be disposed at landfills. Fig. 7 shows the selected MSWM options for solution C, which increments the weight values again for both the energy and emissions objectives. The selection of technologies includes now both

RDF and waste to energy to try to improve the environmental objectives at the expense of cost. Finally, and following the same reasoning, Fig. 8 shows the selected options for solution D, which now includes only waste to energy and RDF, and excludes yardwaste composting.

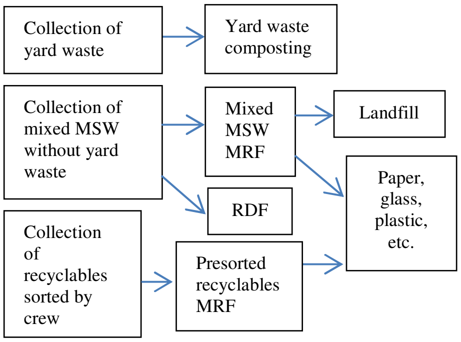

Collection of yard waste
Yard waste composting
Collection of mixed MSW without yard waste
Mixed MSW MRF
Landfill
RDF
Paper, glass, plastic, etc.
Collection of recyclables sorted by crew
Presorted recyclables MRF

Figure 6. Management options for Solution B

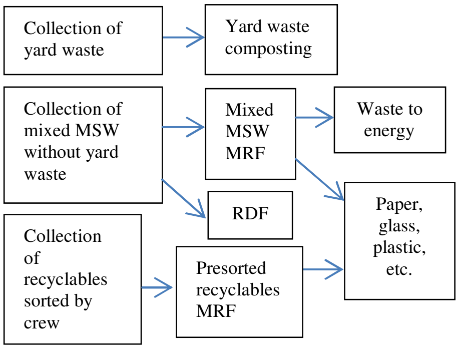

Collection of yard waste
Yard waste composting
Collection of mixed MSW without yard waste
Mixed MSW MRF
Waste to energy
RDF
Paper, glass, plastic, etc.
Collection of recyclables sorted by crew
Presorted recyclables MRF

Figure 7. Management options for Solution C

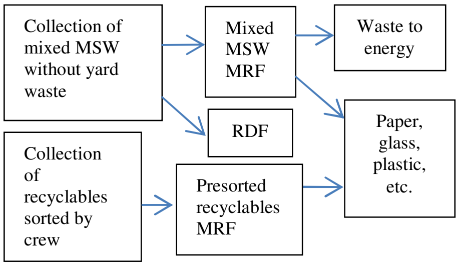

Collection of mixed MSW without yard waste
Mixed MSW MRF
Waste to energy
RDF
Paper, glass, plastic, etc.
Collection of recyclables sorted by crew
Presorted recyclables MRF

Figure 8. Management options for Solution D

E. Multi-objective Fuzzy Optimization

The multi-objective fuzzy optimization method simplified the task of choosing a unique solution among possible solutions from the noninferior solution set determined previously. Because the different objectives use very different measuring units, the weights used in the traditional weighted sum method were counter-intuitive. By using the membership functions in the multi-objective fuzzy optimization method, the objective values are normalized to comparable dimensionless units.

Table 3 shows the results obtained after running a number of scenarios by varying the relative importance given to the objectives. The relative importance given to the objectives through the weights had a direct effect on their reported degree of achievement. The greater the relative importance given to an objective, the greater the degree of achievement was for that objective in the reported solution.

Scenario 1 was obtained by assigning five times more importance to cost than to the other objectives. Cost obtained a degree of achievement of 87% while the other two objectives obtained degrees of achievement below 60%. Scenario 2 assigned five times more importance to energy than to cost, and the degree of achievement obtained for energy was 97% in detriment of cost. Since energy and CO_2 are non-competing objectives, the degree of achievement for CO_2 was also high. Scenario 3 assigned five times more importance to CO_2 than to cost, and the degree of achievement obtained for CO_2 was 97% in detriment of cost. Again, since energy and CO_2 are non-competing objectives, the degree of achievement for energy was high.

The combinations of MSWM options corresponding to the solutions obtained in these 3 scenarios are also different from the least cost, minimum energy and minimum CO_2 solutions, and from the solutions B, C and D obtained in the previous section. The fuzzy multi-objective linear programming method presented here allows for the exploration of the decision space, as different combinations of management options are likely to be selected when new noninferior solutions are chosen.

TABLE III. SUMMARY OF SOLUTIONS FROM FUZZY MULTI-OBJECTIVE OPTIMIZATION

Scenario	Relative importance	Objective values (1=cost, 2=energy, 3=CO2)	Degree of achievement (1- λ_i)
1	w1 =5	Z1=36.4 M$	λ1=87%
	w2=1	Z2=1.21 trillion BTU	λ2=53%
	w3=1	Z3=137.0 million lbs	λ3=60%
2	w1 =1	Z1=51.1 M$	λ1=33%
	w2=5	Z2=2.20 trillion BTU	λ2=97%
	w3=1	Z3=194.8 million lbs	λ3=95%
3	w1 =1	Z1= 53.3 M$	λ1=25%
	w2=1	Z2= 2.20 trillion BTU	λ2=97%
	w3=5	Z3= 197.6 million lbs	λ3=97%

IV. DISCUSSION

Municipal Solid Waste Management (MSWM) is a difficult task when a number of optional technologies and management processes are available to choose from. MSWM simulation is crucial to help generate and analyze multiple and different waste management scenarios. It allows the analyst to choose from a number of optional objectives. Initially, optimal solutions were found for a waste management scenario under three different objectives: minimum cost, minimum carbon dioxide emissions and minimum energy consumption. The combination of unit processes associated with these optimal solutions was very different from one another. While the minimum cost solution included collection of mixed waste and processing of recyclables in a mixed waste MRF, the other two scenarios included additional recyclable collection and processing options. Only the minimum cost solution included yardwaste composting. Both minimum energy and minimum carbon dioxide emissions included waste-to-energy as a means of offsetting energy consumption with the electricity generated at the waste-to-energy plant. The mandated diversion rate of 15% was satisfied by all three scenarios in different ways. The minimum cost solution diverted waste by composting yardwaste and recycling. The other two scenarios relied on recycling only. Both minimum energy and minimum carbon dioxide used waste-to-energy to offset energy consumption and emissions from regular unit processes.

Decision makers may also want to find MSWM options while satisfying multiple objectives at once. The MSWM simulation allowed us to perform multiple-objective optimization to find interesting and different solutions from those obtained by minimizing on one objective at a time. The weighted sum methodology is a simple and reliable way to obtain non-inferior solutions from the Pareto optimal set. Each individual non-inferior solution represents a compromise between the competing objectives, and will favor one objective with respect to the other depending on the relative weights used. The convexity of the Pareto frontier means that, when moving from one non-inferior solution to another, an improvement in one of the objectives will represent degradation in the other objective. Non-inferior solutions represent interesting multi-objective scenarios for the decision maker to consider. They represent different points in the objective space, combining multiple objectives by means of the weighted sum methodology. They also represent potential different points in the decision space, which can provide valuable information about a variety of MSWM strategies. A drawback of the traditional weighted sum methodology is the potential disparity in the values of used weights, due to the different measurement units used by the objectives. The fuzzy multi-objective methodology presented here provides the modeler with the possibility to select a noninferior solution and quantify the degree of achievement for each of the competing objectives. By normalizing the objective values via the membership functions, it eliminates the inconvenience of having dissimilar measurement units. It offers a method to manage imprecision and uncertainty in the objectives via fuzzy membership functions defined based on the worst achievement level and the best achievement level for each objective. By changing the relative level of importance on the objectives, the user can explore the noninferior set and have a quantifiable means to rank the degree of optimization obtained for each individual objective. The user may also experience different combinations of waste management options, as changes in the solution space may imply changes in the decision space.

REFERENCES

[1] E. Solano, S. R. Ranjithan, M. A. Barlaz, and E. D. Brill Jr., "Life-cycle-based solid waste management: 1. Model development," ASCE Journal of Environmental Engineering, 128(10): 981–992, 2002.

[2] E. Solano, R. D. Dumas, K.W. Harrison, S. R. Ranjithan, M. A. Barlaz, and E. D. Brill Jr., "Lifecycle based solid waste management: 2. Illustrative applications," ASCE Journal of Environmental Engi-neering, 128(10): 993–1005, 2002.

[3] K. W. Harrison, R. D. Dumas, E. Solano, M. A. Barlaz, E.D. Brill, and S. Ranjithan, "Decision support tool for life-cycle-based solid waste management," Journal of Computing in Civil Engineering, 15(1), 44–58, 2001.

[4] K. S. Raju, and D. N. Kumar, "Application of multi-objective fuzzy and stochastic linear programming to Sri Ram Sagar Irrigation Planning Project of Andhra Pradesh", ISH Journal of Hydraulic Engineering, Volume 6, Issue 1, 2000.

[5] J. M. Cadenas, and J. L. Verdegay, "Using ranking functions in multiobjective fuzzy linear programming", Fuzzy Sets and Systems, 111, 4-53, 2000.

[6] R. Minciardi, M. Paolucci, M. Robba, and R. Sacile. "Multi-objective optimization of solid waste flows: Environmentally sustainable strategies for municipalities", Waste Management, 28, 2202-2212, 2008.

[7] P. A. Thakre, D. S. Shelar, and S. P. Thakre, "Solving fuzzy linear programming problem as multi objective linear programming problem", Journal of Engineering and Technology Research, Vol 2(5), pp. 82-85, 2009.

[8] H. Katagiri, M. Sakawa, K. Kato, and I. Nishizaki, "Interactive multiobjective fuzzy random linear programming: maximization of possibility and probability", European Journal of Operational Research, Vol. 188, Issue 2, pp. 530-539, 2008.

[9] C. Stanciulescu, P. Fortemps, M. Installe, and V. Wertz, "Multiobjective fuzzy linear programming problems with fuzzy decision variables", European Journal of Operational Research, Vol. 149, Issue 3, pp. 654-675, 2003.

[10] A. Amid, S. H. Ghodsypour, and C. O'Brien, "Fuzzy multiobjective linear model for supplier selection in a supply chain", International Journal of Production Economics, Vol. 104, Issue 2, pp. 394-407, 2006.

A Markov Random Field Approach for Modeling Correlated Failures in Distributed Systems

Jorge E. Pezoa
Electrical Engineering Department and The Center for Optics and Photonics (CEFOP)
Universidad de Concepción, Concepción, Chile E-Mail: jpezoa@udec.cl

***Abstract*—In this paper, logically and spatially correlated failures affecting a distributed-computing system (DCS) have been modeled in a stochastic manner by means of a Markov random field (MRF) approach. The MRF is induced by the topology of the communication network, and is specified locally by the reliability of each node and the degree of interaction between a node and its nearest neighbors. Thus, the MRF introduces a global probability distribution function for the failure patterns of nodes in the DCS, which is parameterized using n values per node, where n is the number of nodes in the DCS. The statistical analysis conducted on test networks has shown that, compared to independent failures, correlated failures increase: (i) the average number of failed nodes due to failures propagate among the nodes; and (ii) the probability of observing a large fraction of failed computing nodes.**

***Keywords*-Distributed computing; Reliability; Markov Random Fields**

I. Introduction

Distributed computing (DC) is computing paradigm that allows to process computationally and data-intensive workloads in a parallel and cooperative fashion using a large number of computing nodes. Unlike other parallel computing environments, in a distributed-computing system (DCS) the memory is not shared, and furthermore, is geographically distributed. Consequently, computing nodes must exchanging data and control messages using an communication network that is usually bandwidth constrained [1].

A DCS is very complex system due to the heterogeneous processing capabilities of the nodes, the large number of elements in the system, the tight coupling of the nodes, the data dependencies in the workloads, and the concurrent dynamics of the nodes, workloads, and network, among other factors. In spite of this complexity, there exists a deep understanding on the computing performance and resource utilization of DCS [2]. The understanding, however, of DCSs's reliability and availability is not as deep as in the case of the aforementioned subjects.

Reliability and availability in DCSs are indeed extremely complicated to model and analyze [2]. To obtain results and achieve conclusions, researchers have simplified the problem assuming that the components in a DCS may fail independently. Under this (on occasions oversimplified) assumption, the reliability of DCSs has been vastly assessed, and studies have been conducted regarding either the computing nodes, the communication network, the application being processed by the DCS, or DCS's management software as the basic component in the analysis. Examples of such results are the works by Ravi *et al.* [3] as well as the work by Srinivasan and Jha [4] where the system reliability of complex DCSs was maximized using task reallocation, or the work by Vidyarthi and Tripathi [5] where both the safety and reliability of a DCS were jointly maximized.

The assumption on the independent node failure in a DCS is a popular because it simplifies the analysis. However, it is clear that such an assumption is not realistic for the type of failures occurring in DCS scenarios, because a DCS is a highly heterogeneous computing environment that imposes a significant communication latency [2]. Furthermore, a DCS becomes highly dynamic due to the communication network and the nodes are affected by a wide class of anomalies that change the topology of the system in a random fashion [6], [7]. These anomalies do exhibit some type of spatial and/or temporal correlation when they result, for instance, from wide-area power or network outages, communication network failures, or missed data dependencies. In addition, these correlated failures may also induce further failures in other nodes, as a result of the lack of reliable communication between the components of the DCS. For instance, by analyzing DCSs' logs stored at The Failure Trace Archive [8] we noticed that the first failure triggered by a power outage indeed produced a correlated failure in two nodes of the high-performance computing (HPC) system at Los Alamos National Laboratory. In fact, other authors have thoroughly analyzed failure logs in large-scale systems and concluded that: (i) correlated node failures are frequent events affecting such systems; and (ii) correlated failures reduce the reliability of a DCS [9], [10]. Finally, we add that it has been quantified that correlated failures may reduce the system unavailability by orders of magnitude [11], [12].

In this paper, we have tackled the problem of modeling the reliability of DCSs affected by correlated component failures. To model correlated failures, a Markov random field (MRF) [13] approach has been undertaken to derive a Gibbs probability distribution [13] for the patterns of correlated failures affecting the DCS. The Gibbs distribution is induced by the underlying network topology of the DCS, which has been abstracted using graph theory. In addition, the Gibbs distribution has been parameterized by individual and group node parameters, such as the reliability of each node and the degree of interaction between a node and its nearest neighbors. Equipped with this global Gibbs distribution function, patterns of correlated-failure can be sampled using an algorithm whose inputs are: the graph modeling the topology of the DCS and n parameters per node, where n is the number of nodes in the DCS.

The rest of this paper is organized as follows. In Section III, we build the model for correlated failures. In Section IV, we present simulations results on the impact of correlated failures in the reliability of DCSs and, in Section V, the conclusions of this work are outlined.

II. Related Work

Under scenarios of application different from DC, correlated failures have been extensively modeled. For instance, in computer networks correlated failures have been modeled using naive yet effective methods, such as regarding clusters of nodes, whose joint probability of failure is “large enough,” as prone to fail in a correlated manner [14]. Jiang and Cybenko used hidden Markov models to detect temporally and spatially correlated failures in a network security system [15]. Fu and Xu reported in [16] a proactive management system for node failures using a failure predictor based on spatial and temporal correlations. Other researchers have developed models for correlated failures triggered by massive natural and/or man-made events [17]–[19]. These events occur within a certain geographical region and physically damage several nodes. For example, correlated failures were modeled using geographical distances and failure probabilities in [17], while the concept of probabilistic shared-risk groups was used in [18], and a Strauss spatial point process was employed in [19] to capture the aforementioned failures.

In the context of storage systems, Bakkaloglu *et al.* [20] modeled the availability of a storage system in the presence of correlated failures introducing using two representations. The first representation used the so-called correlation level parameter, which was defined as the conditional likelihood of failure at a unit, given that another system unit has already failed. The second representation relies on the capability of the Beta-Binomial distribution to capture correlation among interconnected storage units. Later, Nath *et al.* modeled correlated failures in wide-area storage systems by fitting a bi-exponential distribution for the number nodes failing in a correlated manner [21]. In software development, Goseva *et al.* [22] and by Dai *et al.* [23] modeled correlation in software reliability using a Markov renewal process which incorporated the dependencies among successive software runs. In the context of system monitoring, Fiondella and Gokhale derived analytical expressions, based on pairwise component correlations, for the reliability in an on-demand system exhibiting correlated failures [24].

Some models for correlated failures in DCSs have been also proposed in the literature. To the best of our knowledge, the work by Tang and Iyer is the first paper on modeling correlated failures in multicomputer systems [6]. In this pioneering work, the authors tackled the modeling problem by analyzing traces from real systems and proposed a two-phased hyperexponential model for the time between failures. It is noteworthy to mention that correlation was modeled in the time domain assuming that failures propagate among nodes. Following the same ideas, Nath *et al.* studied the effects of failure patterns on the availability of DCSs using traces from real-world systems [25]. Dai *et al.* evaluated the reliability of a grid computing system by modeling the failure correlation appearing in the different subtasks executed by the grid. Chen *et al.* reported in [7] a model for temporally correlated failures in HPC systems which captures cyclic dependencies among the tasks executed by the nodes. Gallet *et al.* created a database of system logs and modeled correlated failures in a probabilistic fashion using parametric models with time-varying parameters [9].

III. Model for correlated failures

The key idea is to develop a model for correlated failures capturing the logical and spatial interaction among the nodes in a DCS. To do so, we first abstract the logical and geographical connections between the nodes in a DCS by means of the underlying topology of the network connecting the nodes. Next, the ability of MRFs to model correlated phenomena has been exploited by defining meaningful local interactions that are simple to specify. These interactions in turn, define a global Gibbs distribution of logically and spatially correlated failures. The technical details of the model are provided next.

A. Markov random fields approach for modeling spatially correlated failures

Suppose that the undirected graph $G = (V, E)$ represents the topology of a DCS, where $V = \{1, \ldots, n\}$ is the set of nodes and $E \subset V \times V$ represents the underlying topology of the communication network connecting the nodes. In order to capture both logical as well as spatial correlations in a MRF setting, the following neighborhood system is introduced:

$$\mathcal{N}_v \triangleq \{u : d_W(v,u) \leq D_{\max} \vee d_L(v,u) = 1,\ u, v \in V\}. \tag{1}$$

In words, two nodes are neighbors if their Euclidean (geographical) distance is within the range $D_{\max}$ or if they have a direct connection with each other. From this definition of neighborhood, the graph G induces the neighborhood system $\mathcal{N}$.

Suppose now that X_i is a binary random variable representing if a node has failed (“1”) or not (“0”). The definition of neighborhood-system in conjunction with the collection of binary random variables $\mathbf{X} = \{X_i,\ i \in V\}$ taking values on the configuration space $\Omega = \{0,1\}^n$ is employed here to introduce a MRF. The definition of the MRF is complete when the Markovian condition is specified, that is, the MRF is completely determined when the likelihood of failure of a node, conditional on the failed or working state of its neighbor nodes is specified.

Requirements. It is of interest here analyzing the performance of DCSs in scenarios where the failure of a node induces failures in other functioning nodes, for instance, due to the inability of the working nodes to exchange data and information with a failed node. It is also of interest to this work to model situations where the geographical or logical proximity of a functioning node to a failed node increases the probability of failure on the functioning node and its neighbor nodes. To fulfill all these requirements, the following local specification for the probability of failure of the node, say, v, given the failed or working state of its neighbor nodes is proposed:

$$\begin{aligned}&\mathsf{P}\{X_v = x_v | \mathbf{X}(\mathcal{N}_v) = \mathbf{x}(\mathcal{N}_v)\}\\&\quad= \frac{\exp\left(-T^{-1}\, x_v (r_v - \sum_{u\in\mathcal{N}_v} s_{v,u}\ x_u)\right)}{1+\exp\left(-T^{-1}(r_v - \sum_{u\in\mathcal{N}_v} s_{v,u}\ x_u)\right)}, \qquad (2)\end{aligned}$$

where T is a constant, r_v is a non-negative parameter modeling the resilience of the vth node to failures and $s_{v,u}$ is a non-negative parameter modeling the strength of interaction between the nodes u and v.

Note that, due to the summation in (2), the likelihood of node v of being in a failed state, $x_v = 1$, effectively increases when one or more of its neighboring nodes are also in a failed state, $x_u = 1$. Moreover, consider the following definition for $s_{v,u}$, the strength of interaction parameter:

$$s_{v,u} = \begin{cases} \frac{D_{\max}}{d_W(v,u)} + s_L & \text{, if } u\in\mathcal{N}_v \wedge d_W(v,u) \le D_{\max} \\ s_L & \text{, if } u\in\mathcal{N}_v \wedge d_W(v,u) > D_{\max} \\ 0 & \text{, if } u\notin\mathcal{N}_v \end{cases}, \qquad (3)$$

where s_L is a non-negative parameter modeling the logical strength of interaction between nodes v and u. This inhomogeneous definition for $s_{v,u}$ clearly increases the likelihood of failure of nodes when they are geographically or logically close to failed nodes.

The equivalence between MRFs and Gibbs fields can be exploited to determine the energy function. By invoking the law of total probability, the definition (2) and recalling the Markovian condition it is straightforward to obtain the energy function in terms of second-order Gibbs potentials:

$$\begin{aligned}\mathcal{E}(\mathbf{x}) &= \sum_{v\in V} r_v\ x_v - \sum_{v\in V}\sum_{u\in\mathcal{N}_v} s_{v,u}\ x_v x_u,\\ &= \mathbf{x}^T\mathbf{r} - \mathbf{x}^T\mathbf{A}\mathbf{x}, \qquad (4)\end{aligned}$$

where $\mathbf{x} = (x_1 \dots x_n)^T$, $\mathbf{r} = (r_1 \dots r_n)^T$, and $\mathbf{A} = (s_{v,u})_{n\times n}$. Thus, the Gibbs distribution associated with this energy function is

$$\pi_{\mathbf{X}}(\mathbf{x}) = \frac{1}{Z_T}\exp\left(-\frac{\mathbf{x}^T\mathbf{r} - \mathbf{x}^T\mathbf{A}\mathbf{x}}{T}\right). \qquad (5)$$

Note that the local specification (2) is independent of the normalizing constant Z_T, while the Gibbs distribution depends on it. Note also that, when the strength of interaction parameters are equal to zero, the Gibbs distribution reduces to the case of independent failures.

B. A Gibbs sampler for generating correlated failures

Realizations of spatially correlated failures following a Gibbs distribution can be sampled, in theory, from (5). Unfortunately, the normalizing constant T is usually hard to compute since due to the large dimension of the configuration space. In order to circumvent this problem, sampling algorithms such as Gibbs or Metropolis samplers can be employed to generate realizations of (5). These sampling algorithms yield realizations of MRFs by constructing a field-valued, homogeneous Markov chain that has as its stationary distribution, the desired Gibbs distribution. The idea of the algorithm is to generate a realization of a Markov chain that, at a large number of iterations, will be close to (5). A key result in MRFs theory proves that a Markov chain having as a stationary distribution (5) can be constructed using the local specifications [13].

From (2), the local specifications for the vth node are:

$$p_0 = \pi\left(0|\mathbf{x}(\mathcal{N}_v)\right) = \frac{1}{1+\exp\left(-T^{-1}(r_v - \sum_{u\in\mathcal{N}_v} s_{v,u}\ x_u)\right)}, \qquad (6)$$

$$p_1 = \pi\left(1|\mathbf{x}(\mathcal{N}_v)\right) = \frac{\exp\left(-T^{-1}\,(r_v - \sum_{u\in\mathcal{N}_v} s_{v,u}\ x_u)\right)}{1+\exp\left(-T^{-1}(r_v - \sum_{u\in\mathcal{N}_v} s_{v,u}\ x_u)\right)}. \qquad (7)$$

Once these expressions are known, the Gibbs sampler can be implemented. Algorithm 1 shows the details of the sampling process, whose main idea is the following: Starting with an initial random configuration, at each iteration of the algorithm a node is randomly picked, say the vth node. The value of the realization x_v associated with the random variable X_v, is updated according to either p_0 or p_1. This process is repeated a large number of times, K, and as a result of these K iterations a sample from (5) is obtained.

IV. Simulation results

To demonstrate the ability of the MRF-based model for generating correlated failures, DCSs with representative network topologies have been considered. In the examples a nationwide DCS has been considered, where nodes are located at several cities in the USA, as shown in Fig. 1. The network topology of the first DCS considered is a realization from the class of the so-called random networks. In the second and third DCSs considered, the underlying communication networks correspond to modified versions

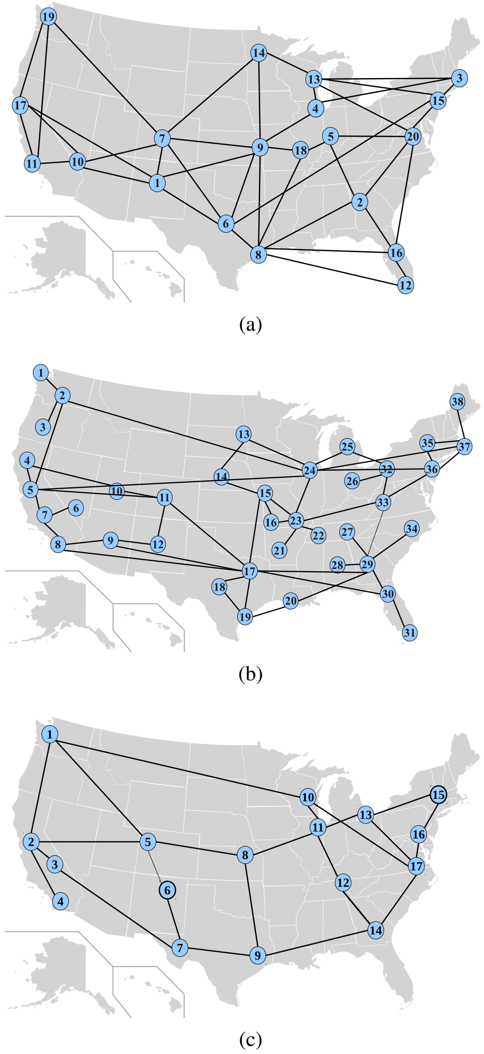
Algorithm 1 Gibbs sampler for the distribution (5). Algorithm taken from [26].

```
Require: $G = (V, E)$, $T$, $r_v$, $s_L$, $D_{\max}$, $\mathcal{N}_v$, and $K$
Ensure: $\mathbf{x}$
  Set $\mathbf{x}^0$ to any random value in $\Lambda^V$
  Set $k = 0$
  while $k \leq K$ do
    $\mathbf{x}^{k+1} \leftarrow \mathbf{x}^k$
    Randomly pick $v \in V$
    Compute $p_0$ using (6)
    Generate a random number $\alpha \sim U[0, 1]$
    if $\alpha < p_0$ then
      Set $x_v^{k+1} = 0$
    else
      Set $x_v^{k+1} = 1$
    end if
    $k \leftarrow k + 1$
  end while
  $\mathbf{x} \leftarrow \mathbf{x}^k$
```

of the AT&T IP backbone network 2Q2000 [27]. The DCSs studied in this section comprise 20, 38 and 17 nodes, and the Fiedler connectivity of the communication networks are 0.47, 0.45 and 0.23, respectively. (The Fiedler connectivity is defined as the second smallest eigenvalue associated with the Laplacian matrix of the graph G modeling the topology of the network [28].)

Samples of correlated failures have been drawn using the Gibbs sampler shown in Algorithm 1. To model the resilience of the nodes to failures, the r_i parameters were set, for simplicity, to be homogeneous for all the nodes. The coupling or strength of interaction parameters, $s_{i,j}$, can be defined as homogeneous ($s_{i,j} = s$ for all $i, j \in V$) or heterogeneous ($s_{i,j}$). Here, such parameters are mainly heterogeneous because they depend on the geographical distance of the nodes; however, for simplicity the parameter s_L modeling the logical strength of interaction between neighboring nodes was defined to be homogeneous. Thus, unless otherwise stated, the following parameters have been used to generate patterns of correlated failures on the DCS: $T = 1$, $r_i = 2$, and $s^L = 1$. Additional parameters employed are: (i) the Gibbs sampler iterates $K = 50000$ times before yielding a sample of the MRF; and (ii) covariance matrices were estimated using 2000 realizations of the MRF. For comparison, the case of independent failures has been also simulated by setting all the strength of interaction parameters to zero, due to when $s_{i,j} = 0$ for all i and j in (5), the Gibbs distribution reduces to a product of exponential distributions, which corresponds in fact to a probability distribution for independent failures.

Correlated failure patterns have been tested by generating a total of 2000 failure realizations, and sampled covariance matrices have been computed. Each off-diagonal element of such matrices was statistically tested for correlation using a t-test for the hypothesis of no correlation with a confidence of 99%. The results of these tests and a sample realization of correlated failures in the DCS were used to construct the



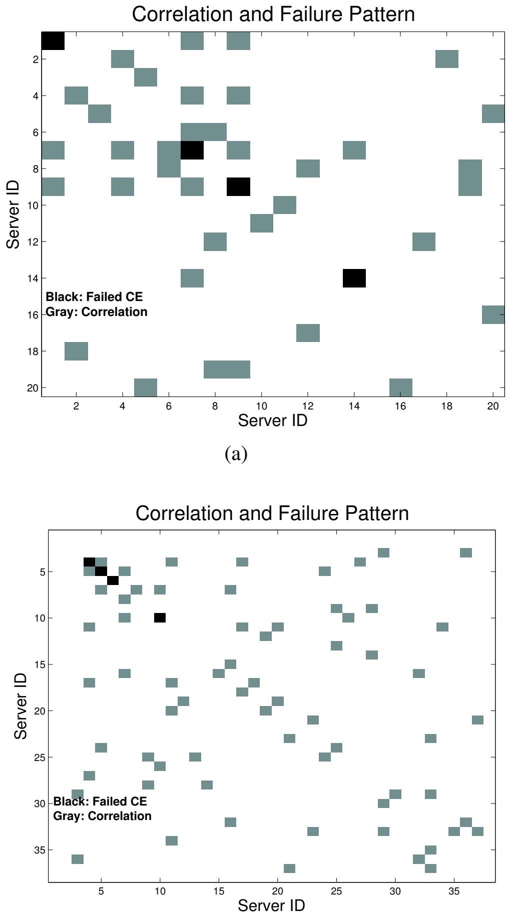
(a)

(b)

(c)

Figure 1: (a) Sample DCS composed of 20 nodes. (b) DCS interconnected by means of the AT&T IP backbone network 2Q2000 [27]. (c) DCS interconnected by means of a simplified version of the AT&T IP backbone network 2Q2000 [27].

images shown in Fig. 2. The elements in the diagonal of the matrices correspond to a sample realization from the Gibbs distributions obtained using Algorithm 1. A node, say, the ith node has become failed if the ith diagonal element is “1” (black rectangle) and is in a working state if the ith element is “0” (no rectangle). The off-diagonal elements of the matrices shown in Fig. 2 represent if there is correlation (value “1” depicted as a gray rectangle) or not (value “0” depicted using no rectangle) between the ith and jth nodes. Note that the pattern of correlated failures in Fig. 2(a) shows that four out of twenty nodes have failed. These four nodes are nodes 1, 7, 9, and 14, which clearly form a cluster of nodes in Fig. 1(a). Note also that the off-diagonal elements of the correlation matrix depicted in Fig. 2(a) confirm with a 99% confidence the correlation between the failures at the



(a)

(b)

Figure 2: Matrices showing the spatial correlation in the case of the (a) sample network with 20 nodes, and (b) the AT&T IP backbone network 2Q2000. The elements in the diagonal of the matrices correspond to a sample realization from the Gibbs distributions. A red color means a failed node. The off-diagonal elements show if there is (blue color) or not (white color) spatial correlation between the failures at the nodes.

aforementioned nodes. Similarly, Fig. 2(b) shows a sample failure pattern where the directly connected nodes 4, 5, and 10, as well as the stub node 6, have failed in a correlated manner.

Figures 3 and 4 show the effect of both the resilience parameter and the strength of interaction parameter on the average number of failed nodes for the DCSs depicted in Fig. 1(a) and (b). As expected, it can be observed from the figures that the average number of failed nodes increases as the robustness parameter decreases. In addition, as either the homogeneous or the heterogeneous strength of interaction parameter increases so it does the average number of failed nodes. This behavior suggests that failures propagate more intensely as these parameters increase, thereby reflecting the fact that failures become more correlated as the coupling between nodes increases. Also as expected, when the robustness parameter is fixed, the average number of failed nodes is larger in the case of correlated failures as compared to the case of a independent failures. Finally, note that the slopes of the plots in Figs. 3(a) to (c) are steeper than those shown in Figs. 4(a) to (c). This is attributed to the connectivity of the underlying networks associated with the DCSs. Note that the topology of the 20-node DCS is more connected than the 38-node DCS. Such a fact is clearly reflected in the Fiedler eigenvalue [28]. As a consequence of this greater connectivity, the logical coupling between nodes is naturally accentuated due to the larger number of relative connections in the 20-node DCS as compared to the connections in the 38-node DCS.

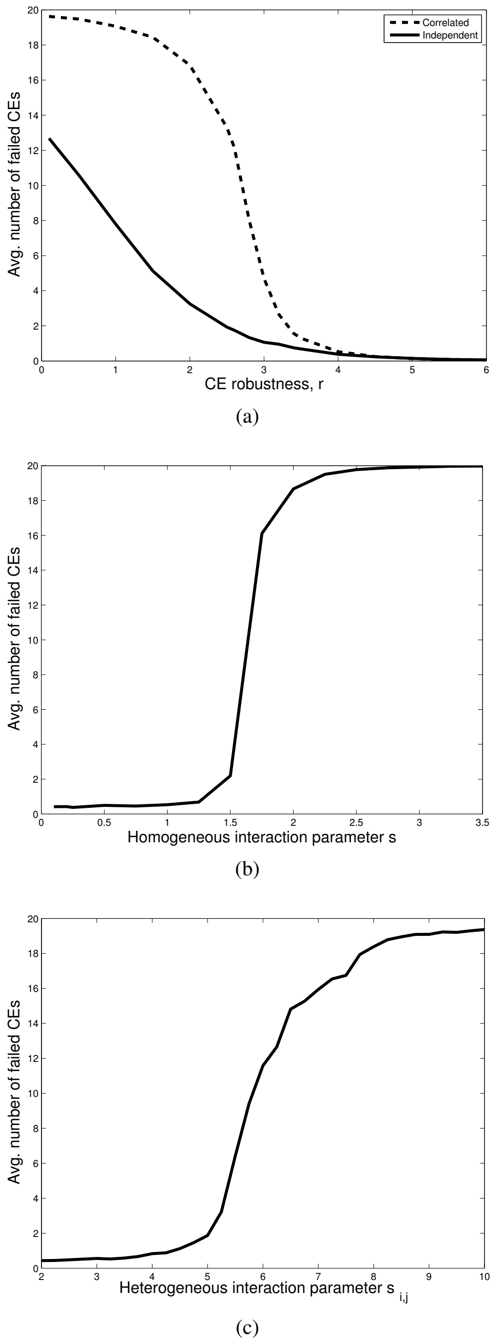
Table I: FAILURE PATTERNS IN CORRELATED AND INDEPENDENT FAILURE SCENARIOS FOR THE DCS IN FIG. 1(c).

	Probability of failure patterns		
	Correlated	Independent	Failure pattern
Clustering Effect	0.081	10^{-42}	All nodes
	0.063	10^{-35}	All except 2, 4, 10, 11
	0.030	10^{-41}	All except node 17
	0.030	10^{-41}	All except node 7
	0.030	10^{-41}	All except node 2
Inhibition Effect	10^{-5}	0.006	One node only
	4×10^{-5} to 10^{-7}	4×10^{-5}	Two nodes only
	5×10^{-5} to 10^{-9}	4×10^{-7}	Three nodes only

Table I compares the effect of correlation parameters s_L and $D_{\max}$ on some interesting failure patterns for the 17-node DCS shown in Fig. 1(c). The normalizing constant has been calculated by considering all the values in the configuration space for independent and correlated failures. With this, the probability of each specific failure pattern can be calculated from the Gibbs distribution (5). Results show that the probability of having a large fraction of the nodes failing is much higher in the correlated-failure case than in the independent-failure case. As expected from such a model, the correlation parameters $s_{v,u}$ can be used to control the degree of failed-nodes clustering or bunching. Similarly, the probability of failure patterns with very few failed nodes is much lower in the correlated-failure scenario than that corresponding to the independent case. Namely, there is a weaker "inhibition" effect in the correlated-failure scenarios compared to the independent-failure scenario.

V. Conclusion and Future Work

This paper presented a novel framework, based on MRFs and graph theory, for modeling correlated failures of computing nodes. The model abstracts the arbitrary topology of the underlying network connecting the nodes in a DCS. The developed failure model captures the spatial correlation between nodes with logical and geographical connections and captures also the percolation effect of node damage across the DCS. The model was developed by defining local conditional specifications of failure probabilities, which can

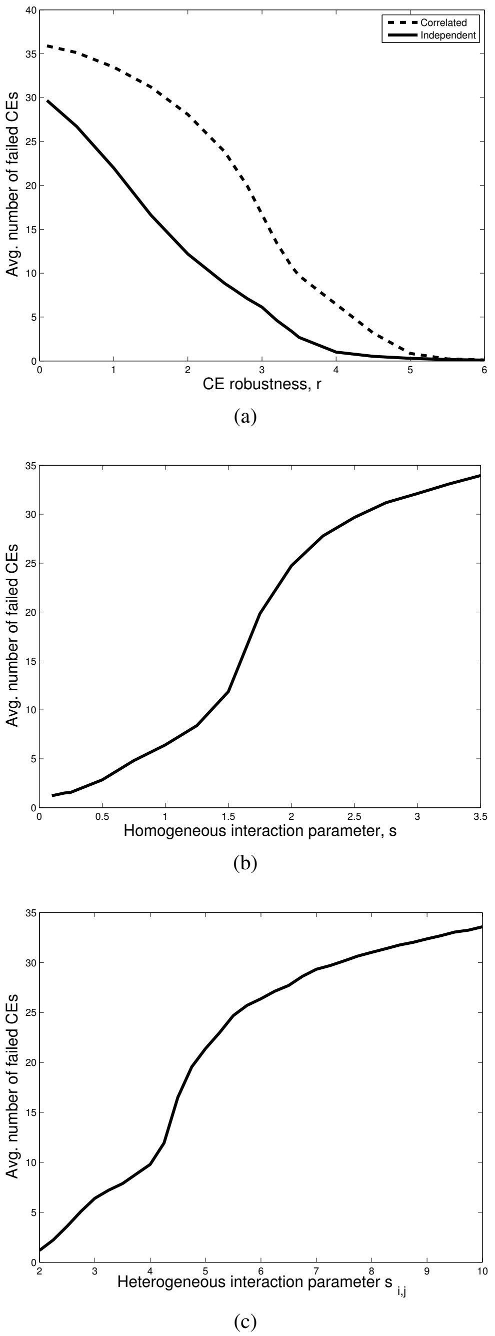


(a)

(b)

(c)

Figure 3: Average number of failed nodes versus (a) r_v parameter, (b) s_L parameter, and (c) $D_{\max}$ parameter for the DCS with 20 nodes shown in Fig. 1(a).



(a)

(b)

(c)

Figure 4: The average number of failed nodes versus (a) r_v parameter, (b) s_L parameter, and (c) $D_{\max}$ parameter for the DCS with 38 nodes shown in Fig. 1(b).

be easily specified in practice since they are related directly to both the geographical and logical relations imposed by the topology of the DCS. Key in the development of the model are the set of parameters termed as the strength of interaction between nodes, which quantifies the degree of interaction between nodes in terms of physical distances and also in terms of logical coupling.

The statistical analysis conducted on realizations obtained from the model for correlated failures has shown that the failure of a single node does propagate to other functioning nodes, and the degree of propagation depends on the intensity of the so-called inter-node strength of interaction parameter. The analysis confirms also that the average number of failures increases as the logical and geographical strength of interaction between nodes increases and, as expected, the average number of failed nodes also increases, as compared to the case of independent failures, when correlated failures affect the nodes of a DCS. Analytical results show also that the probability of having a failure pattern involving a large fraction of the nodes is considerably higher than in the case of independent failures, when correlated failures affect the system. Moreover, the strength of interaction parameters specified in the model can be used to limit the number of failed-nodes. This result is of practical interest in order to identify the vulnerabilities associated with coordinated attacks on the network infrastructure of the DCS.

As a future work, traces available on Internet from production systems will be studied and the proposed Gibbs distributions will be fitted in order to validate the proposed model.

ACKNOWLEDGMENT

The author acknowledges the support of both Fondecyt Project 11110078 and the Basal Project FB024.

REFERENCES

[1] R. Shah, B. Veeravalli, and M. Misra, “On the design of adaptive and decentralized load balancing algorithms with load estimation for computational grid environments,” *IEEE Trans. Parallel and Dist. Systems*, vol. 18, pp. 1675–1686, 2007.

[2] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains*, 2nd ed. John Wiley & Sons, Inc., 2006.

[3] V. Ravi, B. Murty, and J. Reddy, “Nonequilibrium simulated-annealing algorithm applied to reliability optimization of complex systems,” *IEEE Trans. Reliability*, vol. 46, pp. 233–239, 1997.

[4] S. Srinivasan and N. Jha, “Safety and reliability driven task allocation in distributed systems,” *IEEE TPDS*, vol. 10, pp. 238–251, 1999.

[5] D. Vidyarthi and A. Tripathi, “Maximizing reliability of a distributed computing system with task allocation using simple genetic algorithm,” *J. Syst. Arch.*, vol. 47, pp. 549–554, 2001.

[6] D. Tang and R. K. Iyer, “Analysis and modeling of correlated failures in multicomputer systems,” *IEEE Trans. Comput.*, vol. 41, no. 5, pp. 567–577, 1992.

[7] X. Chen and X. He, “Tolerating temporal correlated failures from cyclic dependency in high performance computing systems,” in *Proc. 14th IEEE Int. Conf. Parallel and Distributed Systems*, Washington, DC, USA, 2008, pp. 509–516.

[8] INRIA, “The Failure Trace Archive,” http://fta.inria.fr, 2012, [Online; accessed May-2012].

[9] M. Gallet, N. Yigitbasi, B. Javadi, D. Kondo, A. Iosup, and D. Epema, “A model for space-correlated failures in large-scale distributed systems,” in *Proc. 16th Euro-Par Conf. on Parallel processing*, 2010, pp. 88–100.

[10] P. Joshi, H. S. Gunawi, and K. Sen, “Prefail: A programmable failure-injection framework,” EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2011-30, Apr 2011. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2011/EECS-2011-30.html

[11] R. B. Kiran, K. Tati, Y. chung Cheng, S. Savage, and G. M. Voelker, “Total recall: System support for automated availability management,” in *In NSDI*, 2004, pp. 337–350.

[12] P. Yalagandula, S. Nath, H. Yu, P. B. Gibbons, and S. Seshan, “Beyond availability: Towards a deeper understanding of machine failure characteristics in large distributed systems,” in *Proc. Of Usenix Workshop On Real, Large Distributed Systems*, 2004.

[13] P. Bremaud, *Markov chains, Gibbs fields, Monte Carlo simulation, and queues*, 2nd ed. Springer-Verlag, New York, 2001.

[14] H. Weatherspoon, T. Moscovitz, and J. Kubiatowicz, “Introspective failure analysis: Avoiding correlated failures in peer-to-peer systems,” *Reliable Distributed Systems, IEEE Symposium on*, 2002.

[15] G. Jiang and G. Cybenko, “Temporal and spatial distributed event correlation for network security,” in *Proc. of American Control Conference*, Boston, MA, 2004.

[16] S. Fu and C. Z. Xu, “Quantifying temporal and spatial correlation of failure events for proactive management,” in *Proc. 26th IEEE International Symp. on Reliab. Dist. Systems*, 2007.

[17] K. Kim and N. Venkatasubramanian, “Assessing the impact of geographically correlated failures on overlay-based data dissemination,” in *GLOBECOM*, 2010, pp. 1–5.

[18] S. Neumayer, G. Zussman, R. Cohen, and E. Modiano, “Assessing the Vulnerability of the Fiber Infrastructure to Disasters,” *Networking, IEEE/ACM Transactions on*, vol. PP, no. 99, p. 1, 2010.

[19] M. Rahnamay-Naeini, J. E. Pezoa, G. Azary, N. Ghani, and M. M. Hayat, “Modeling stochastic correlated network failures and assessing their effects on reliability,” in *Proc. IEEE Int. Conf. Computer Communications Networks*, 2011.

[20] M. Bakkaloglu, J. J. Wylie, C. Wang, and G. R. Ganger, “On correlated failures in survivable storage systems,” Parallel Data Laboratory, Carnegie Mellon University, Tech. Rep., 2002.

[21] S. Nath, H. Yu, P. B. Gibbons, and S. Seshan, “Subtleties in tolerating correlated failures in wide-area storage systems,” in *Proceedings of the 3rd conference on Networked Systems Design & Implementation - Volume 3*, 2006.

[22] K. Goseva-Popstojanova and K. S. Trivedi, “Failure correlation in software reliability model,” *IEEE Trans. Reliability*, vol. 49, pp. 37–48, 2000.

[23] Y. Dai, M. Xie, and K. Poh, “Modeling and analysis of correlated software failures of multiple types,” *IEEE Trans. Reliability*, vol. 54, pp. 100–106, 2005.

[24] L. Fiondella and S. S. Gokhale, “Estimating system reliability with correlated component failures,” *International Journal of Reliability and Safety*, vol. 4, no. 2–3, pp. 188–205, 2010.

[25] S. Nath, S. Nath, H. Yu, H. Yu, P. B. Gibbons, P. B. Gibbons, S. Seshan, and S. Seshan, “Tolerating correlated failures in wide-area monitoring services,” Intel Research, Tech. Rep., 2004.

[26] J. E. Pezoa, “Theory of resource allocation for robust distributed computing,” Ph.D. dissertation, The University of New Mexico, Albuquerque, NM, USA, 2010.

[27] M. Dodge and R. Kitchin, *The Atlas of Cyberspace*. Addison Wesley, 2008.

[28] C. Godsil and G. Doyle, *Algebraic Graph Theory*. Springer Science+Bussiness, New York, NY, USA, 2006.

Traffic and Monotonic Total-Connected Random Walks of Particles

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

***Abstract*–An analytical and simulation models of random walk of particles on a closed one-dimensional lattice are considered. In these models, the particles contained in a cluster move synchronously. The problem is to find the average time interval after which only a cluster remains. This problem is solved with both analytical and simulation approaches. A simulation model is also developed that describes the movement of a particles on a ring with traffic lights. An appropriate analytical model is also developed with some different rules of functioning. The average velocity of the particle is calculated. The results obtained with the simulation and analytical model are compared. Simulations models are also described that are supposed to be developed for the traffic with traffic lights, for multi-lane case, and networks with a periodic structure on that total-connected random walks occur.**

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

I. Introduction

In [1], some analytical and simulations models in terms of random walks are considered that can be interpreted as traffic models. These mathematical models can be interpreted as cellular automata. The models of this class were introduced in [2, 3] and were investigated in a lot of works. The scheme considered in [2, 3] is similar to monotonic random walks on a lattice. The work of Yu. Belyaev and his students [4, 5] is devoted to traffic flows in the underground and contains exact results for one-dimensional random walk (not only monotonic). Appropriate references are given in [1]. Some results in this field have been found in [6–11].

In the present paper, we consider a modification of a model of random walks on a circular lattice. A sequence of adjacent particles is called a cluster. The clusters are separated one from another by empty cells. In the considered model, all the particles within a cluster move synchronously. The number of clusters can decrease and cannot increase. After some time interval with a finite average value, only a cluster remains, if the probability of the cluster movement at a time is less than 1. A problem is solved to find an average time of coming to the state with one cluster. If the lattice is small, then the problem can be solved with an analytical approach. If the number of particles is rather big, then the analytical approach is too complicated. A simulation model is developed that is useful in this case.

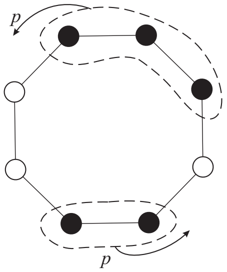


Fig. 1. Total-connected walks on a circle

We call the considered random walks totally-connected, because, in our model, every particle that was contained in a particular cluster always remains contained in the same cluster.

We also develop a simulation model that describes the movement of particles on a ring with traffic lights. An appropriate analytical model is also developed with some different rules of movement. The average velocity of the particle is calculated. The results obtained with the simulation and analytical model are compared.

Simulation models are also described that are supposed to be developed for the traffic with traffic lights, for multi-lane case, and networks with a periodic structure on which total-connected random walks occur. The flow intensity is investigated with these simulation models.

II. The average duration of the time interval after which only one cluster remains

2.1. Consider a closed sequence of cells. The number of cells is equal to n. There are m $(m < n)$ particles. No cell contains more than one particle. The clusters can move one cell forward at discrete times $0, 1, 2, \dots$ in one direction, Fig. 1. At each discrete time each cluster move one cell forward with the probability p_i, where i is the index of the particle in front of the cluster.

The behavior of the model is described by a Markov chain, [12]. The states of this chain corresponds to the configurations of the particles on the lattice.

Suppose d_i is the average duration of the time interval after which the number of clusters decreases, if at the initial time the chain state is i (the states of the chain are numerated arbitrarily); q_{ij} is the probability that, at the time when the number of clusters decreases, the chain state is j, if the ith state was initial; D_i is the average duration of the time interval after which the number of clusters becomes one. Let A_k be the set of states with k clusters and B_k be the set of states with no more than k clusters.

The problem of calculation of the values d_i and q_{ij} for $i \in A_k$ is reduced to the appropriate problems if it is supposed that the number of cells equals $n-m+k$ and there no cluster containing more than one particle.

We have

$$D_i = \sum_{j \in B_{k-1}} q_{ij} D_j + d_i, \; i \in A_k. \qquad (1)$$

Recurrent formula (1) allows to reduce the problem of calculation of D_i to the problem of calculation of d_i and q_{ij}.

In formula (1), the time interval after which the number of clusters becomes one consists of the time interval after which the number of clusters decreases, and the time interval since the end of the first time interval until the time when the number of clusters becomes one. The average value of the first time interval equals d_i. and the probability that after this interval the chain state is j equals q_{ij}. The average value of the second time interval equals D_j provided it begins itself at the state j. The average value of the total time interval equals D_i. Thus, formula (1) is valid.

In turn, the consideration of this model is reduced to the consideration of a random walk of a particle on a facet of an m-dimensional tetrahedron. Indeed, if x_i is the number of empty cells between the ith particle and the following particle, then the value of the sum $x_1 + \dots + x_m$ remains constant. The model states correspond to the facet on that this sum remains constant. The model states correspond to the facet $x_1 + \dots + x_m \le n - m$ of the tetrahendron $x_1 \ge 0, \dots, x_m \ge 0$, $x_1 + \dots + x_m \le n-m$. The particles coordinates correspond to the lengths of intervals between the particles in the original model, and each coordinate cannot decrease and increase at once more than by one. The problem is to find the average value of the duration of time interval after which the particle comes to the boundary of the facet, where at least one coordinate is equal to zero, and to find the probability that the particle comes to the boundary at a given point. Such problems are solved with an approach described in [12], and these problems are reduced to systems of linear equations. However, the number of the equations can be too big for the system could be solved in practice. Therefore, a simulation model can be useful.

2.2. Suppose $m = 2$. At the initial time there are z empty cells from the particle 2 to the particle 1 in the direction of the particles movement. At each time $0, 1, 2, \dots$ the particle 1 moves with the probability p_1 and the particle 2 moves with the probability p_2 $(0 < p_1, p_2 < 1)$.

In this case, the problem is reduced to the consideration of random walks on a segment. Denote by d_z the average duration of the interval after which the particles form a cluster. Using an approach described in [12], we get the formulas for d_z.

Proposition 1.

By the above conditions the following formulas are true

$$d_z = \frac{z(n-2-z)}{2p(1-p)}, \; p_1 = p_2 = p,$$

$$d_z = \frac{z}{(1-p_1)p_2 - p_1(1-p_2)} -$$

$$-\frac{(n-2)\left(\left(\frac{(1-p_1)p_2}{p_1(1-p_2)}\right)^z - 1\right)}{((1-p_1)p_2 - p_1(1-p_2))\left(\left(\frac{((1-p_1)p_2}{p_1(1-p_2)}\right)^{n-2} - 1\right)}, \qquad (2)$$

$$p_1 \ne p_2.$$

Denote by d the average duration of the time interval after which the particles are joined.

Proposition 2.

Suppose that all the configurations of two particles on the ring have the same probabilities $p_1 = p_2 = p$. Then

$$\lim_{n \to \infty} \frac{n^2}{12p(1-p)} \cdot \frac{1}{d} = 1,$$

i.e., the asymptotic estimation is true for big values of n

$$d = d(n) =\sim \frac{n^2}{12p(1-p)}.$$

The dependence of $\frac{n^2}{12p(1-p)} \cdot \frac{1}{d}$ on n is showed in Fig.2.

2.3. Suppose now that $m = 3$. TThe case in which the number of particles is more than 3 can be considered similarly. Suppose that the probability of the particle movement at a time is equal to p. Suppose the particle 2 follows the particle 1 in the direction of the movement. Denote by x, y and z the number of cells between the particles 1 and 2, between the particles 2 and 3, and between the particles 3 and 1, appropriately. The model is described by a Markov chain. Each state of this chain corresponds to some point (x, y, z),

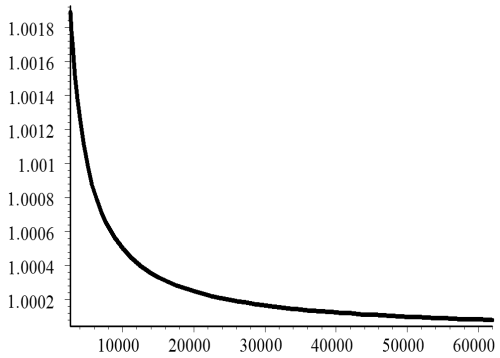


Fig. 2. Asymptotic behavior of the value of d

$x+y+z=n-2$, $x,y,z \geq 0$. Denote by $d(x,y,z)$ the average duration of the time interval after which any two particles form a cluster, if the initial state is (x,y,z), $xyz \neq 0$. Denote by $q(x,y,z;x_0,y_0,z_0)$ the probability that, at the time at that two particles form a cluster, the chain state is (x_0,y_0,z_0), if the initial state is (x,y,z), $xyz \neq 0$, $x_0y_0z_0 = 0$. The problem is reduced to the investigation of random walks on a facet of a tetrahedron.

Using an approach described in [12], we find *the following system of linear equations, which has a unique solution,*

$$3p(1-p)d(x,y,z) =$$

$$= p^2(1-p)d(x+1,y,z-1) + p^2(1-p)d(x-1,y+1,z)+$$

$$+p^2(1-p)d(x,y-1,z+1) + p(1-p)^2d(x-1,y,z+1)+$$

$$+p(1-p)^2d(x+1,y-1,z) + p(1-p)^2d(x,y+1,z-1) + 1,$$

$$xyz \neq 0;$$

$$d(x,y,z) = 0,\ xyz = 0.$$

Each equation of this system corresponds to some set (x,y,z), $x>0$, $y>0$, $z>0$, $x+y+z=n-3$.

Using the same approach, we get *the following system of linear equations, which has also a unique solution,*

$$3p(1-p)q(x,y,z;x_0,y_0,z_0) =$$

$$= p^2(1-p)q(x+1,y,z-1;x_0,y_0,z_0)+$$

$$+p^2(1-p)q(x-1,y+1,z;x_0,y_0,z_0)+$$

$$+p^2(1-p)q(x,y-1,z+1;x_0,y_0,z_0)+$$

$$+p(1-p)^2q(x-1,y,z+1;x_0,y_0,z_0)+$$

$$+p(1-p)^2q(x+1,y-1,z;x_0,y_0,z_0)+$$

$$+p(1-p)^2q(x,y+1,z-1;x_0,y_0,z_0),\ xyz \neq 0,$$

$$x_0y_0z_0 = 0;$$

$$q(x_0,y_0,z_0;x_0,y_0,z_0) = 1,\ x_0y_0z_0 = 0;$$

$$q(x,y,z;x_0,y_0,z_0) = 0,\ (x,y,z) \neq (x_0,y_0,z_0),$$

$$xyz = 0,\ x_0y_0z_0 = 0.$$

As above, each equation of this system corresponds to some set (x,y,z), $x>0$, $y>0$, $z>0$, $x+y+z=n-3$.

The proof of the fact that the solution of each of this two systems is unique uses an approach described in [12]. This proof is based on that each inner point (x,y,z) cannot be a point of maximum for $d(x,y,z)$, and, since $d(x,y,z)=0$ for the boundary points $xyz=0$, the the homogeneous system of linear equations has only the zero solution.

III. Simulation model of random walks on a circle

Models that simulate the random walks on a circle have been developed.

There is no cluster that contains more then one particle at the initial state.

The average duration of the interval after which only a cluster remains is investigated.

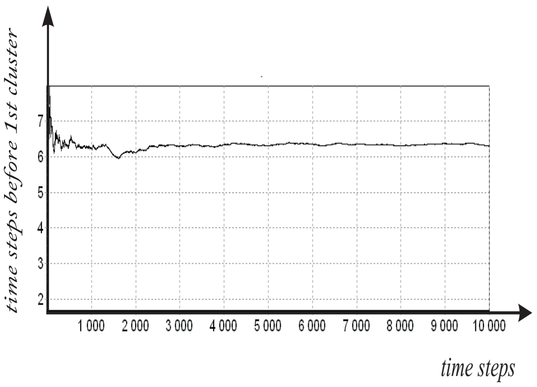


Fig. 3. The average time after which the number of clusters decreases

The dependence of the average number of time steps before the first cluster appears on the number of experiments is represented in Fig. 3. The number of experiments is represented on the x-axis, and the investigated average time is represented on the y-axis. The number of simulation experiments is equal to 10000. We suppose the number of cells equals 20, the number of particles equals 4. The probability that the particle moves at a fixed time is equal to 0.5. The initial distribution of particles is uniform, i.e., before the start of simulation, the particles are inserted into the cells one after another, and the probability that the particle is inserted to a given cell is the same for all the empty cells.

The dependence of the average number of time steps before the first cluster appears on the flow density is represented in Fig. 4. The flow density, i.e., the ratio of the number of particles to the number of cells is represented on the x-axis, and the investigated average time is represented on the y-axis. The number of simulation experiments is equal to 100000 for each density value. We suppose the number of cells equals 100. The probability that the particle moves at a fixed time is equal to 0.5. The initial distribution of particles is uniform as it described above.

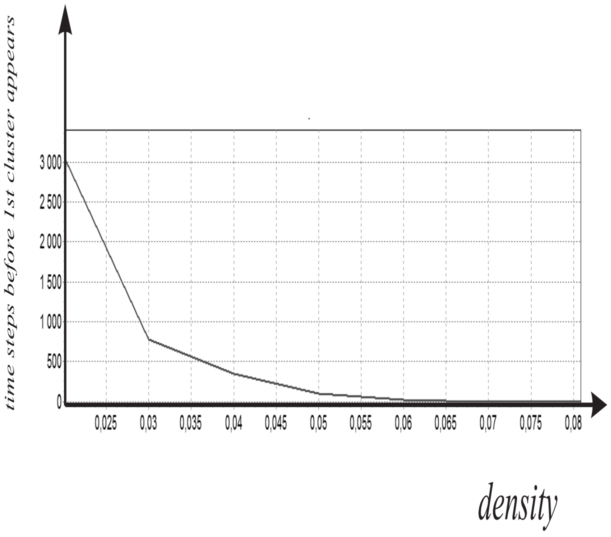


Fig. 4. The average number of time steps before the first new cluster appears

IV. Total-connected walks of a finite number of particles on a straight line

Consider also monotonic total-connected walks of particles on an infinite one-dimensional lattice.

Suppose there is a finite number of particles on a straight line. Suppose there are m clusters. The probability that the ith cluster moves at a time is equal to p_i, $i = 1, \dots, m$. (The first cluster is ahead. The $(i+1)th$ cluster follows the ith cluster, $i = 1, \dots, m-1$.) Then at each time the difference between the coordinates of the first and the last particle increases by one with the probability $p_1(1-p_m)$, decreases by one with the probability $p_m(1-p_1)$, and does not change with the probability $p_1p_m + (1-p_1)(1-p_m)$.

The problem is reduced to the results on the symmetric random walk of a particle on a straight line described in [12, 13].

In [13], a one-dimensional random walk of a particle on a lattice is considered. At each discrete time the particle with probability p moves by one position to the right and with probability q moves by one position to the left, $p + q = 1$. It is proved, [12, 13], that, if $p = q = 1/2$, then the particle returns to a given position with the probability 1 after a finite time interval, but the duration of this interval is infinite. If $p > q$, then with a positive probability the particle shifts to $+\infty$ no returning to the initial position. If $p < q$, then with a positive probability the particle shifts to $+\infty$ no returning to the initial position.

Similarly, we have in our model that, *if $p_1 = p_2 = \cdots = p_m$, then the duration of the interval after which only one cluster remains is finite with the probability 1, but the average duration of this interval is infinite. If $p_1 < p_2 < \cdots < p_m$, then the average duration of the interval after which only one cluster remains is finite. If $p_1 > p_i$ for some i, then this interval is infinite with a positive probability.*

Let us estimate the average duration of the interval after which only one cluster remains, if $p_1 < p_2 < \cdots < p_m$.

Let us consider the case of two clusters with the probabilities of the movement at the current time p_1 and p_2. Let the number of cells between the particles at the initial time be z. The problem is reduced to the consideration of a random walk of a particle that is at the point z in the initial time. The particle moves to the right by one position with the probability p, and the particle moves to the left by one position with the probability q, $p + q = 1$. If $p < q$, then the probability 1 the particle comes with the probability 1 to the position 0 after a finite time interval. The average of the average duration of this interval is equal to $z(q-p)^{-1}$.

In our case, we have to suppose $p = p_i(1-p_{i+1})/(p_{i+1}(1-p_i) + p_i(1-p_{i+1}))$, $q = p_{i+1}(1-p_i)/(p_{i+1}(1-p_i) + p_i(1-p_{i+1}))$. Taking into account that with the probability $1-p_i(1-p_{i+1}) - p_{i+1}(1-p_i)$ the particle remains at the current time at the same point, and therefore the interval for that the particle does not change its position has the average duration

$$\frac{1}{p_{i+1}(1-p_i) + p_i(1-p_{i+1})},$$

we get that in our model, the average duration d_z of the time interval after which two particles form a cluster can be calculated as

$$d_{iz} = \frac{z}{p_{i+1}(1-p_i) - p_i(1-p_{i+1})}. \tag{3}$$

Formula (3) is the limit case of (2) as n tends to ∞.

Suppose now that m is arbitrary, and z_i is the number between the ith and the $(i+1)$th particles, $i = 1, \dots, m-1$. Using (3), we have the following *the upper bound for the average duration of the interval after which only a cluster remains*

$$d < \frac{\sum\limits_{i=1}^{m-1} z_i}{p_2(1-p_1) - p_1(1-p_2)}.$$

We have taken into account that the distance between the particle 1 and the last particle decreases stochastically no less slowly than the distance between particles 1 and 2, because $p_i \geq p_2$, and hence $p_i(1-p_1) - p_1(1-p_i) \geq p_2(1-p_1) - p_1(1-p_2)$, $i = 2, \dots, m-1$.

V. Movement in the presence of an obstacle

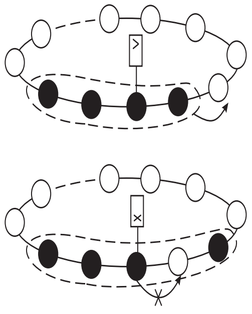


Fig. 5. Movement in the presence of an obstacle

Consider the model that is different from the model considered in Sections 2 and 3, by the fact that an obstacle exists on the circle at a given cell (see Fig. 5). This obstacle can be interpreted as the red traffic light. After some time interval the obstacle disappears, and a green phase follows. A cluster is divided if the obstacle appears within the cluster.

The appropriate simulation model is intended to be developed. The main characteristics of movement that have to be investigated in simulation experiments are the average velocity of a particle and the flow intensity. The average velocity v is the average number of the cells that a particle passes per a time unit. The flow intensity q is the average number of particles that passes through a section of the ring per a time unit. For a one-lane model $q = rv$, where r is the flow density, i.e., the ratio of the number of particles to the number of cells.

Consider a model that can describe approximately traffic with traffic lights. Suppose there is a closed sequence of cells. The number of cells is equal to n. There is one particle on the ring. If the particle is in the cell i at the time k, then at the time $k+1$ the particle with the probability a_i, $0 < a_i < 1$, is in the cell $i+1$ and with the probability $1 - a_i$ the particle remains in the cell i, $i = 1, \ldots, n-1$. If the particle is in the cell n at the time k, then at the time $k+1$ the particle with the probability a_n, $0 < a_n < 1$, is in the cell 1 and with the probability $1 - a_n$ the particle remains in the cell n.

Proposition 3.

The formula is true

$$v = \frac{n}{\sum\limits_{j=1}^{n} \frac{1}{a_j}}. \tag{4}$$

Proof. The behavior of the model is described by a Markov chain [12]. Each of the n state of this chain corresponds to the index of the cell that contains the particle. The chain states have stationary probabilities that satisfy the system of equations

$$a_1 p_1 = a_n p_n,$$

$$a_i p_i = a_{i-1} p_{i-1}, \; i = 2, \ldots, n,$$

$$p_1 + \cdots + p_n = 1.$$

This system has a unique solution

$$p_i = \frac{1/a_i}{\sum\limits_{j=1}^{n} \frac{1}{a_j}}, \; i = 1, \ldots, n. \tag{5}$$

The average number of transitions of particles is called the average velocity v of the particle, and

$$v = \sum_{i=1}^{n} p_i a_i. \tag{6}$$

Formula (4) follows from (5) and (6).
Proposition 3 has been proved.

Thus, the average velocity of the particle can be calculated with (4).

Using (4), we can estimate the average velocity of a particle in the model in which there are traffic lights at the cells, and a_i is the ratio of the duration of a green phase to the duration of the total cycle of the traffic lights located at the cell i, $i = 1, \ldots, n$. If there is no traffic lights at the cell i, then $a_i = 1$.

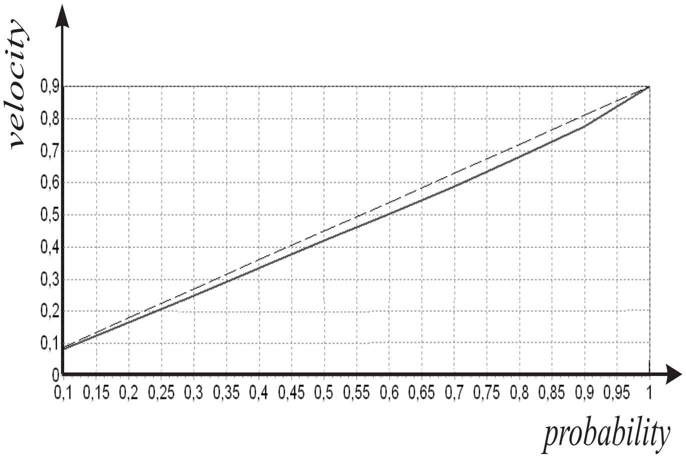


Fig. 6. Comparison of the average velocities found with the simulation and analytical models with some different rules of traffic lights functioning

In Fig. 6, the results of the comparison of the average velocities, found with the simulation and analytical models with some different rules of functioning, is represented. The analytical model is the same as described above. Suppose $n = 9$, $a_1 = p/2$, $a_i = p$, $i = 2, \ldots, n$, where p is a variable, value of which is represented on x-axis. The value of the particle velocity is represented on y-axis. In simulation model the particle moves to the next cell at each time with the probability p. There is a traffic light in the cell 1. The duration of both the green and the red phase is equal to a time unit with the probability 1. The dashed line corresponds to the formula (3) and the solid line corresponds to the simulation experiments. The number of experiments is 100000 for every value of p.

VI. The model of two-lane traffic

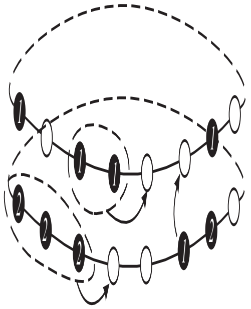


Fig. 7. A model of two-lane movement

Consider a generalization of the model of random walks on a circle.

Suppose there are two sequences of cells (two lanes) that form a ring of the dimension $2 \times n$. There are m particles on the ring, as shown in Fig. 7.

There are two types of particles. A batch of particles of the same type occupying adjacent cells of the same lane is called a cluster. The clusters of the first type particles are fast, and the clusters of the second type particles are slow. The particles of the same cluster move synchronously. All the particles of a cluster move one cell forward at each discrete time with the probability that depends on the type of the particle in front of the cluster. Clusters can be both united or divided. Two clusters of the same type are united, if they are on the same lane, and one of them catches up the other. Particles contained in a cluster can change the movement lane, and it can occur that the particles of the same cluster become occupying cells on the different lanes. A change of the movement lane can occur if a fast cluster catch up a slow cluster and the appropriate cells are empty, Fig. 8. The particles of the cluster change the movement line one at a time.

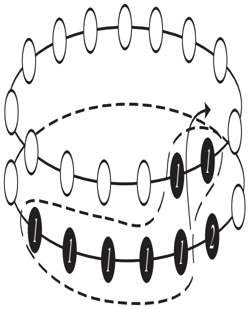


Fig. 8. Particles of a cluster change the movement lane

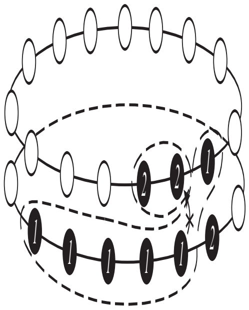


Fig. 9. Situation in that a cluster is divided

A cluster can be divided. Such a situation is showed in Fig. 9.

The main characteristics of the movement that need to be investigated in simulation experiments are the average velocity v of a particle and the flow intensity q: The follow formula is true $q = 2rv$, where q is the flow intensity, v is the average velocity, r is the flow density.

VII. Two rings model

Another generalization of the model of the random walks on a circle is the two rings model.

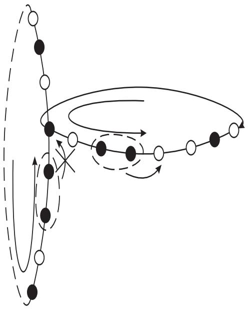


Fig. 10. Two rings model

Suppose there are two sequences of cells (two rings) that have a common cell, Fig. 10. The first ring contains n_1 cells, and the second ring contains n_2 cells. There are m_1 particles on the first ring, and there are m_2 particles on the second ring. The particles contained in the same cluster move synchronously. All particles of the cluster move one cell forward at a discrete time with the probability that can depend on the type of the particle in front of the cluster. Two clusters united if a cluster catch up an other cluster on the same circle.

If two particles can enter into the common cell, then the particle moving on the first ring has the priority. The cluster is divided if a part of the cluster have passed the common cell, and the rest of the cluster is blocked, Fig. 10.

The main characteristics that have to be investigated in simulation experiments are the average velocity of particle and the flow intensity on each ring. The steady probability that a given cell is occupied has to be also estimated.

VIII. Model of a network

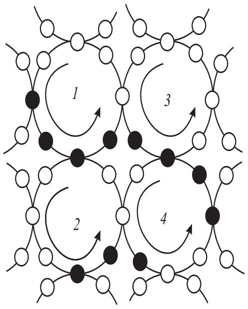


Fig. 11. A fragment of a regular network

Simulation models describing the behavior of networks have to be developed. One of the possible model structures is showed in Fig. 11. The rules of the particles movement are similar to the rules described in the previous sections. If the probability that a cluster moves at each discrete time equals 1 and the flow density is less than 1/2, then all the clusters can move with the velocity equal to 1. If the flow density on each cluster is more than 1/2, the average velocity is less than 1.

IX. Conclusion and Future Work

The analytical and simulation models of random walks of particles have been developed. The particles of the same cluster move synchronously. The results of simulation experiments were represented. The developed models can be used for traffic analysis and optimization [14]. Here are some simulation models that have to be developed:

• Total-connected movement in form of a clustering flow is observed in many cases, e.g., pedestrians, cyclists, traffic flows.

• Clusters are the limit state of the solutions of the system of non-linear ordinary differential equations in the car following model, [15].

• In the methodological sense, cluster objects simplify the investigation of the flow problem on a network.

• Traffic control can increase the number of clusters.

• Problems of clustering on networks are supposed to be investigated.

REFERENCES

[1] Buslaev, A., Tatashev, A., and Yaroshenko, A. ”Traffic and monotone random walk of particles: analytical and simulation results”, The Third International Conference on Advances in System Simulation — SIMUL 2011, October 23-29, 2011. – Barcelona, Spain. Think Mind. View article simul_2011_3_20_50018.

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

[3] Schreckenberg M., Schadschneider A., and Nagel K., Ito N., “Discrete stochastic models for traffic flow”, Phys. Rev. E., 1995, vol. 26, pp. 2939–2949.

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

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

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

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

[8] Blank, M. ”Exclusion processes with synchronous update in transport flow models”, Trudy MFTI, 2010, vol. 2, no. 4, pp. 22–30.

[9] Buslaev, A. and Tatashev, A. “Particles flow on the regular polygon”, Journal of Concrete and Applicable Mathematics (JCAAM), 2011, vol. 9, no. 4, pp. 290–303.

[10] Buslaev, A. and Tatashev, A. “Monotonic random walk on a one-dimensional lattice”, Journal of Concrete and Applicable Mathematics (JCAAM), 2012, vol. 10, no’s 1–2.

[11] Buslaev, A. and Tatashev, A. ”On exact values of monotonic random walks characteristics on lattices”, International Conference on Applied Mathematics and Approximation Theory AMAT 2012. Abstracts, Ankara, 2012.

[12] Feller, A. ”An introduction to probability theory and its applications”, New York: John Wiley, 1968.

[13] Karlin, S. ”A first course in stochastic processes”, New York and London: Academic Press, 1968.

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

[15] Buslaev, A., Gasnikov, A., and Yaschina, M. ”Selected mathematical problems of traffic flow theory”, International Journal of Computer Mathematics 2012, vol. 89, N.3, pp. 409–432.

Model-based Prediction of Complex Multimedia/Hypermedia Systems

Franco Cicirelli, Libero Nigro, Francesco Pupo
Laboratorio di Ingegneria del Software
Dipartimento di Elettronica Informatica e Sistemistica
Università della Calabria - 87036 Rende (CS) – Italy
Email: f.cicirelli@deis.unical.it, {l.nigro,f.pupo}@unical.it

Abstract–**This work develops a model-based approach to the specification and analysis of the functional/temporal behavior of complex multimedia/ hypermedia systems. The approach is centered on Time Stream Petri Nets (TSPN) as the authoring formal language and on UPPAAL as the target tool for model checking activities. A structural translation from TSPN to UPPAAL timed automata (TA) was recently defined in the form of a reusable TA library. The library was proven to be timed bisimilar to TSPN. This paper focuses on the practical aspects of the approach by showing its application to a non trivial modeling example related to a hypermedia system whose properties are predicted by exhaustive verification. Finally, conclusions are drawn with an indication of on-going and future work.**

Keywords-Model-based prediction; multimedia/ hypermedia systems; time stream Petri nets; synchronization consistency; model checking; UPPAAL.

I. INTRODUCTION

An interactive multimedia document (IMD) or hypermedia document, concerns the coordinated presentation of different types of information (audio, video, images, text, etc.) possibly associated with user interactions. The quality of the presentation of an IMD depends on the fulfillment of the temporal synchronization constraints, which are associated with the component media objects. Different approaches are described in the literature to address IMD specification and verification/validation of the synchronization consistency. Although an informal/intuitive or general purpose authoring approach can be preferable for editing an IMD, the informal specification must then the converted into some formal notation and associated tools, which can support the necessary verification activities. As an example, the SMIL language [3, 16] can be used as an authoring language. In the approach developed, e.g., in [15], from a SMIL specification some intermediate data structures are firstly generated; this makes it possible to translate the specification into the terms of process algebra of RT-LOTOS, which permits a formal analysis of synchronization consistency.

In this work, a methodology based on Petri nets [11, 13] is proposed for the specification and analysis of IMD. Petri nets have both an intuitive graphical notation and a rigorous mathematical representation for property checking. In particular, Time Stream Petri Nets (TSPN) [4, 10, 14] are chosen for the specification of complex hypermedia documents. TSPN associates temporal validity intervals to input arcs only, and a firing rule, selected in a rich set, to constrain transition firing. Although in the work described in [8-9] a TSPN specification or high level specification is translated into RT-LOTOS for verification purposes, the original contribution of this work is an exploitation of a translation of TSPN into UPPAAL [1], assisted by a library of reusable timed automata [7], which opens model checking to TSPN-based hypermedia documents. The achieved UPPAAL translation was proved to be timed bisimilar to TSPN. TSPN offers great flexibility to the modeler of general time-dependent systems. In [7], it is shown how the formalism is also well suited for modeling and property checking of real-time embedded systems.

Section II introduces the TSPN formalism and its extension HTSPN (Hierarchical TSPN) more suited to hypermedia systems modeling. Flattening problems of HTSPN to TSPN are addressed in Section III with a hypermedia example. The UPPAAL library supporting TSPN is summarized in section IV. Analysis of the synchronization consistency of the hypermedia example is reported in Section V. Finally, conclusions are presented together with an indication of on-going and future work.

II. BASIC CONCEPTS OF TIME STREAM PETRI NETS

A TSPN is a tuple $(P, T, B, F, I_{nh}, M_0, IM, SYN, MA)$ where:

- P is a finite nonempty set of *places;*
- T is a finite nonempty set of *transitions;*
- B is the backward incidence function, $B: P \times T \rightarrow \mathbf{N}$, where $\mathbf{N}$ denotes the set of natural integers;
- F is the forward incidence function, $F: P \times T \rightarrow \mathbf{N}$;
- I_{nh} is the set of inhibitor arcs, $I_{nh} \subset P \times T$ where $(p,t) \in I_{nh} \Rightarrow B(p,t) = 0$;
- M_0 is the initial marking function, $M_0 : P \rightarrow \mathbf{N}$, which associates with each place a number of tokens;
- IM is a function, which associates with each arc, incoming to a transition, an interval defining its static temporal validity interval. $IM : A \rightarrow \mathbf{Q}^+ \times (\mathbf{Q}^+ \cup \{\infty\})$, where $\mathbf{Q}^+$ represents the set of nonnegative rational values, $A = \{a = (p,t) \in P \times T \mid B(p,t) \neq 0 \vee (p,t) \in I_{nh}\}$ is the set of all incoming arcs and $IM(a) = [t_{\min}(a), t_{\max}(a)]$ is such that $t_{\min}(a) \leq t_{\max}(a)$.
- SYN is a function, which associates each transition with a firing rule: $SYN : T \rightarrow \{And,\ Weak-And,\ Or,\ Strong-Or,\ Master,\ Or-Master,\ And-Master,\ Strong-Master,\ Weak-Master,\ Pure-And\}$;
- MA is a function that associates a master arc to each transition whose firing rule requires it, $MA : T_m \rightarrow A$, where:

$T_m = \{t \in T \mid SYN(t) \in \{Master, Or-Master, And-Master, Strong-Master, Weak-Master\}\}$.

The marking of a TSPN is a function $M : P \rightarrow \mathbf{N}$, which associates each place with a number of tokens. An arc $a = (p,t) \in A$ is enabled by a marking M iff: $a \in I_{nh} \Rightarrow M(p) = 0 \wedge a \notin I_{nh} \Rightarrow M(p) \geq B(p,t)$. The set of arcs enabled by a marking M, is denoted by $enArc(M)$. The set of incoming arcs of a transition t is denoted as $A(t)$. A transition t is enabled by the marking M iff: $A(t) \subseteq enArc(M)$. The set of transitions, which are enabled by a marking M is denoted as $enabled(M)$. The set of input places of a transition constitutes its preset. The set of output places of a transition is its postset.

The state of a TSPN model is a pair (M,I) where M is a marking and I is a mapping that associates each arc enabled in M with a dynamic temporal validity interval, i.e., $I : enArc(M) \rightarrow R^+ \times (R^+ \cup \{\infty\})$, $I(a) = [x(a), y(a)]$. The initial state of a TSPN is given by (M_0, I_0), where $\forall a \in enArc(M_0)\ \ I(a) = IM(a)$. A transition t_f is said to be *fireable* at relative time θ from state (M,I) if t_f is enabled by M and $low(t_f) \leq \theta \leq \min_{t \in enabled(M)} up(t)$. For any transition t enabled by M, its dynamic time interval $[low(t), up(t)]$ is determined as follows:

$$\begin{cases}
\left[\max_{a\in A(t)}\{x(a)\}, \max\{\max_{a\in A(t)}\{x(a)\}, \min_{a\in A(t)}\{y(a)\}\}\right] & SYN(t)=And \\
\left[\max_{a\in A(t)}\{x(a)\}, \max_{a\in A(t)}\{y(a)\}\right] & SYN(t)=Weak-And \\
\left[\max_{a\in A(t)}\{x(a)\}, \max\{\max_{a\in A(t)}\{x(a)\}, y(a_m)\}\right] & SYN(t)=And-Master \\
\left[\max_{a\in A(t)}\{x(a)\}, \min_{a\in A(t)}\{y(a)\}\right] & SYN(t)=Pure-And \\
\left[\min_{a\in A(t)}\{x(a)\}, \max_{a\in A(t)}\{y(a)\}\right] & SYN(t)=Or \\
\left[\min_{a\in A(t)}\{x(a)\}, \min_{a\in A(t)}\{y(a)\}\right] & SYN(t)=Strong-Or \\
\left[\min_{a\in A(t)}\{x(a)\}, y(a_m)\right] & SYN(t)=Or-Master \\
\left[x(a_m), y(a_m)\right] & SYN(t)=Master \\
\left[x(a_m), \max\{\min_{a\in A(t)}\{y(a)\}, x(a_m)\}\right] & SYN(t)=Strong-Master \\
\left[x(a_m), \max_{a\in A(t)}\{y(a)\}\right] & SYN(t)=Weak-Master
\end{cases}$$

where $a_m = MA(t)$ if $t \in T_m$. Let t be a transition fireable from state $S = (M,I)$. The state $S' = (M', I')$ reached by firing t at relative time θ, is determined as follows:

- $\forall p \in P\ \ M'(p) = M(p) - B(t,p) + F(p,t)$
- $\forall a \in enArc(M')$

$$\begin{cases}
I'(a) = IM(a) & \text{if } a \notin enArc(M) \vee a \notin enArc(\tilde{M}) \vee a \in A(t) \\
I'(a) = \left[\max\{x(a)-\theta, 0\}, y(a)-\theta\right] & \text{otherwise}
\end{cases}$$

where $\forall p \in P\ \ \tilde{M}(p) = M(p) - B(p,t)$, i.e., $\tilde{M}$ is the intermediate marking resulting from the withdrawal sub-phase. An enabled arc is *violated* if the upper bound of its dynamic temporal validity interval is negative.

A transition t' enabled in M can lose its enabling during the atomic firing process of t either in the intermediate marking $\tilde{M}$ or in the reached marking M'. It is said *non persistent* to the firing of t. On the contrary, a *persistent* transition (which is enabled in M) keeps its enabling during the whole firing process of t. A transition is said *newly enabled* if it was not enabled in M or in $\tilde{M}$ but it is enabled in M'. It is worth noting that a valid firing interval for an enabled transition may exist also in the case the timing constraints of some of the incoming arcs are violated. Once a valid timing interval is found for a transition, it constitutes a strong constraint (as in classical Time Petri Nets [12], which associates a static timing interval to transitions) on its firing. Of course, arc violations can determine transition violation if no valid timing interval is possible for the transition. As a consequence, TPSN use a *weak synchronization model* for arcs but a *strong synchronization model* for *transitions*.

The mathematical definition of transition dynamic firing interval highlights that, in general, a synchronization rule (see also Figure 1) can be driven by

(a) the latest arriving process (*And*, *Pure–And*, *Weak–And*, *And–Master*) where the *last* arc that reaches the lower bound of its temporal interval allows the firing of the related transition

(b) the earliest arriving process (*Or*, *Strong–Or*, *Or–Master*) where the *first* arc, which reaches its lower bound permits the firing of the associated transition

(c) the arriving of a statically selected process (*Master*, *Strong–Master*, *Weak–Master*), i.e., the transition can only fire when its *master* arc reaches the lower bound of its associated temporal interval.

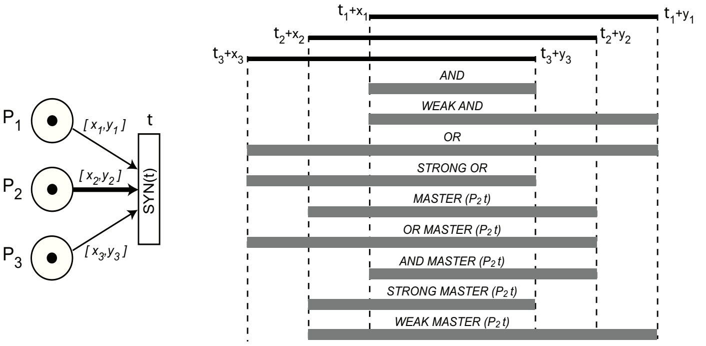


Figure 1. Basic TSPN synchronization rules

As an example of transitions, which can or cannot be violated, differences between the *And* and *Pure–And* firing rules can be pointed out. If input arcs temporal intervals overlap, the behaviour of *And* and *Pure-And* coincide: the transition can only fire in the intersection of the intervals. In the case some intervals are disjoint, at least one arc is violated when a process (input arc) reaches its lower bound. In this situation, the *And* firing rule permits *one* single synchronization point at the lower bound of the latest arriving process. In other words, under the *And* firing rule a transition can always fire, but with the *Pure–And* the firing is impossible to occur when the intersection is void.

Rules *Master* and *Or–Master* can be violated when the master arc is violated. An intriguing case concerns the *Strong–Master* rule. If at the enabling time of the transition a non master arc is already violated, there exists one single synchronization point at the lower bound of the dynamic interval of the master arc. A single point of synchronization at current time is permitted when the master arc is violated at the enabling time of the transition. In the case of an overlapping scenario of the timing intervals, a firing interval for the transition exists, which starts with the lower bound of the master arc (possibly updated to current time if it was passed) and ends with the earliest upper bound. It can easily be verified that rules *Weak–Master*, *And–Master*, *Or, Strong-Or* and *Weak-And* can never be violated.

As it is usual in Time Petri Nets [2], a TSPN multiple enabled transition is assumed to fire its enablings one at a time (*single server semantics*). After its own firing, a transition t, which is still enabled is regarded as a newly enabled one.

III. MODELING A HYPERMEDIA SYSTEM BY TSPN

In order to widen the modeling capabilities of TSPN as an authoring formal tool, in [8], it was proposed HTSPN (Hierarchical TSPN). The concept was introduced of an *abstract place* shown graphically as a dashed circle (see Figure 2). A subnet can be defined within an abstract place, which typically has an input place and a final one. The subnet in turn can have further abstract places and so forth recursively.

Abstract places enable incremental modeling by deferring to a later time the definition of the internal details of the subnet. However, an abstract place must be both *structurally* and *temporally equivalent* to the internal subnet. Structural equivalence means that the subnet could functionally replace the abstract place. Temporal equivalence means that the expected temporal behavior of the subnet must fulfill the temporal constraints expressed at the abstract place (requirement) level.

To favor the authoring of complex multimedia/hypermedia systems, the use of HTSPN can be organized into three synchronization layers (see also Figure 2). At the *link layer* the hypermedia system is designed according to the user-point of view, i.e., conceptually in a similar way to an hypertext. Here, the user can choose a link to interrupt and put forward a given presentation or can require to rewind the presentation from its beginning etc. At the *composite layer*, the hypermedia system is specified through the composition of multiple media, which are operated according to given temporal constraints. Finally, at the *atomic layer,* details concerning the playing/rendering of a multimedia scenario are furnished.

Figure 2 portrays an example hypermedia model adapted from [8], devoted to the presentation of a commercial product. An initial text is presented to the user, which invites to start the presentation by clicking on a Start button. Following a start (modeled by next1 in Figure 2 and the firing of the first Master transition), a token is generated in the place L_2 and in the abstract place Information.

Token in L_2 enables both the possibility for the user to request the Again link, which asks to interrupt current presentation and to restart it from the beginning, or to conclude the presentation by invoking the Next link (next2 arc). The following constraints exist: (a) Again cannot be issued before at least 65 time units (tu) are elapsed from the moment the token was put in L_2; (b) the Next link can only be requested at the end of multimedia presentation. The multimedia presentation is assumed to last in 150 tu. Moreover, 70 tu are allowed, after termination of multimedia presentation, during which the user can ask the Next link for concluding the presentation (see next2 arc). If 70 tu elapses, the Next link is automatically invoked (see the Weak-And transition, which feeds the End place).

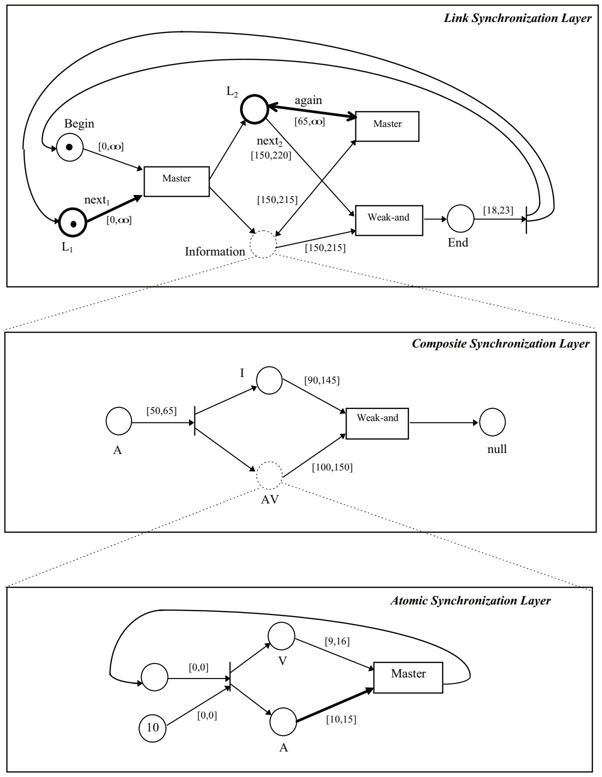


Figure 2. An HTSPN hypermedia model (from [8])

Token in Information abstract place begins the actual presentation. First an audio is played (see token in the A place in the composite layer in Figure 2), which in the worst case has a duration of 65 tu. The system forbids the user to ask the Again link if the audio content is still in progress. At the end of the audio, a token is generated in the place I of the subnet at the composite layer and in the AV abstract place. As a consequence an image is rendered with duration [90,145] tu. Token in AV starts an audio/video scenario detailed in the atomic layer. This multimedia scenario is composed of 10 audio/video objects. A video object (token in the V place) is rendered with a duration of [9,16] tu. A corresponding audio object (token in the A place) has a rendering timing constraint in [10,15] tu. Since the audio is the most important media, the master rule is used for synchronization and the input arc of the audio is defined as the master arc. As one can see from the atomic layer, inter-media synchronization (which affects skew and then *lip-synch*) was introduced just after each presentation of a couple of corresponding audio and video objects. The goal was to minimize the audio/video skew by keeping audio/video synchronized to the largest extent.

The multimedia scenario terminates when all the 10 tokens are consumed. During the presentation of the multimedia scenario, the user can select the link Again, which requires the hypermedia system to interrupt its current operations and to restart from its beginning.

Temporal intervals on the arcs exiting from Information and AV abstract places capture system requirements. The requirement of finishing AV states that the multimedia scenario should terminate within [100,150] tu, which is intuitive by considering that audio objects are most important than video objects and that each audio requires [10,15] tu to be presented. The requirement of conclusion of Information states that the underlying layers must terminate within [150,215] tu (inclusive of the duration of the audio in the A place of the composite layer).

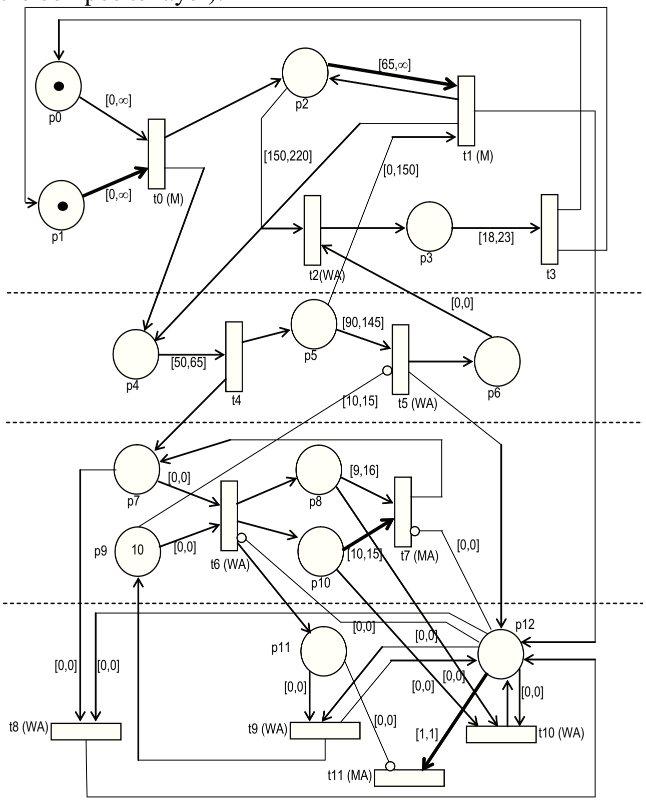


Figure 3. A flattened version of the hypermedia TSPN model

Following a termination of the multimedia scenario, a token is generated in the End place, which models some concluding remarks to the user. Such remarks have a duration in [18,23] tu. When also the remarks are finished, the system comes back to its home marking.

The conceptual model in Figure 2 specifies only hypermedia author requirements. It tacitly delegates the implementation level to correctly restart the Petri net to its home marking following the natural conclusion of the presentation or after each Again request.

In order to clarify problems that arise when an HTSPN hypermedia model like that portrayed in Figure 2 is put in concrete terms, Figure 3 shows a flattened version of the model. The unfolded model highlights dependencies between adjacent layers and, most importantly, solves problems of "dead tokens", which can be left in the model after an Again request. In addition, the flattened model is also in charge of re-installing the initial marking in all the cases it is necessary. A fourth layer, which can be named *clean-up layer*, was introduced with a minimal subnet, which eliminates no longer needed tokens and reconstitutes the model home marking.

For simplicity, in the flattened model a more uniform naming schema was followed for identifying both places and transitions. The clean-up subnet is composed by places p11 and p12 and transitions t8, t9, t10 and t11. At each termination of the multimedia scenario, or following an Again request, a token is deposited in p12, which supervises the model clean-up. Clean-up operations consumes 0 time. The token in p12 is eventually eliminated by a firing of transition t11, which uses the Master firing rule. One time unit is used for ensuring the end of the clean-up operation. Place p11 is the complementary one of the place p9. After each firing of t6 transition, a token is consumed from p9, and a new token is generated in p11. As a consequence, during the clean-up operations, t11 will fire multiple times so as to re-install the 10 initial tokens in p9. It should be noted that an Again request can occur during the presentation of a video/audio couple (one token in p8, one token in p10) or just before the beginning of the next multimedia couple of objects. Therefore, the clean-up subnet will remove no longer useful tokens in p7 or in p8 and p10.

A few inhibitor arcs were introduced in the flattened model in Figure 3. Arc p9t5 ensures the multimedia scenario naturally ends when all the 10 tokens in p9 are consumed. Arcs p12t6, p12t7 forbid prosecution of the multimedia scenario as soon as an interruption request (i.e., Again) is sensed. Finally, inhibitor arc p11t11 ensures that the p12 clean-up token can be destroyed after the home marking of p9 was re-established.

IV. An Uppaal TA Catalog Supporting TSPN

A structural translation was defined in [7], which enables a TSPN model to be equivalently transformed in a network of Uppaal timed automata [1]. The translation associates a timed automaton with each transition with a given firing rule and a certain number of input arcs. Clocks are associated with input arcs only, but they are consulted by the transition according to the chosen synchronization rule. A transition template automaton is parameterized with its unique ID, and clock and bounds of its input arcs. The realization owes to latest Uppaal version (4.0.13), which makes it possible to introduce global/local data structures and functions (in C-like syntax), which contribute to a compact definition of template processes without impairing the efficiency of state graph exploration.

Global constants A, P and T, respectively hold the number of input arcs, the number of places and the number of transitions in the model. Constants PRE and POST denote respectively the maximal cardinality of a transition preset or postset. Constant INF denotes infinity. Topology of a TSPN model is memorized into *constant* matrices Backward B:TxPRE→ArcInfo, Forward F:TxPOST→F, InpurArcs AI:A→InputArcInfo where ArcInfo and InputArcInfo are the following structures:

```
typedef struct{
    int[-1,P-1] index; //place id, -1 for a non existing arc
    int weight; //input/output arc weight
} ArcInfo;
typedef struct{
    int[0,T-1] tid; //transition id
    int[0,PRE-1] pid; //preset id
} InputArcInfo;
```

Dynamic status information of a model is stored in the marking vector M, the enabled input arc vector EA, the input arc clock vector x and the current fired transition variable TID:

```
int[0,K] M[P]={...}; //for a K-bounded a model
bool EA[A]; //initialized to all false
clock x[A]; //one clock per input arc
int[-1,T-1] TID; //transition fired ID, -1 denotes no transition
```

Transition enabling and firing are assisted by the global functions bool enabled(const int ID), void withdraw(const int ID), void deposit(const int ID), which receive the unique transition ID. Input arc statuses and associated clock variables reset are responsibility of the global function void updateArcs(const bool w), which receives a boolean parameter indicating if the update is requested by a withdraw or a deposit operation. When an input arc *a* switches from the disabled to the enabled status or following the firing of its own transition, is still enabled, its status is set in the EA[a] and its clock x[a] is reset so as to start measuring the elapsed time since the instant of arc enabling. Similarly, when *a* is found disabled its status is reset in EA[a] and its clock x[a] is reset provided the function updateArcs() is invoked during a deposit or *a* refers to a transition t different from the current fired transition held in the global TID variable. This provision allows one to check, during verification, the clock value x[a] when a fired transition is in the intermediate withdraw phase.

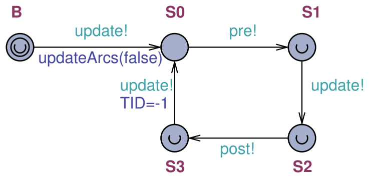


Figure 4. The Supervisor automaton

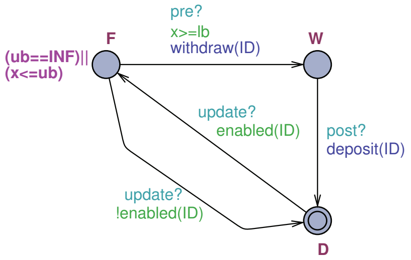


Figure 5. The Transition automaton

A TSPN model with *N* transitions is mapped onto an UPPAAL network of *N*+1 automata where each automaton corresponds to a distinct TSPN transition of the source model. The additional automaton is a supervisor [6] (see also Figure 4) whose responsibility is to allow TSPN transitions to complete their firing one at a time, by stepping through the atomic phases of transition firing.

A fundamental template is Transition (see Figure 5), which has one single input arc and can be used with any firing rule. A Transition instance receives the arc clock x and bounds lb and ub of the arc temporal validity interval.

A Transition automaton starts in the D (disabled) location. As soon as it finds itself enabled, it moves to the F (Firing) location, where it waits for the lower bound lb to be reached. Firing can be completed at any time the clock x is greater than lb but lower than or equal to ub (as stated by F invariant) if ub is finite. Would ub be infinity, the transition can stay in F an arbitrary amount of time. While in F, the transition can move immediately to D if the firing of another transition disables (for a conflict) this transition. Firing completion is mediated by the intervention of the Supervisor, using the two unicast channels pre and post and the broadcast channel update. The first phase of firing completion is for the transition to move from F to W (withdraw) location, under a pre synchronization. In the case multiple transitions are ready to complete their firing, one transition is chosen non deterministically. From W the transition eventually completes its firing by a second synchronization with the Supervisor through the post channel. Before this, the supervisor has to ask *all* the transitions to check their status following the withdraw of current firing transition. This important check is based on the broadcast channel update. A variable number of transitions (even no one transition) can possibly synchronize (update?) with the supervisor. Following an update synchronization, a transition can become enabled or being previously enabled (in the F location) it can become disabled. Urgent locations in the Supervisor ensure the firing process is terminated without passage of time. As one can see from Figure 4, the supervisor cycle includes two update broadcast synchronizations: after token withdrawl and after token deposit.

Initially, the Supervisor sends a first update synchronization so as to allow transitions that are enabled in the initial marking to switch from D to F location. The function updateArcs() is invoked after each change in the model marking, and thus after a withdrawl (within function withdraw()) or a deposit phase (within function deposit()). Initially, in order to permit input arcs to check their status, updateArcs() is invoked by the Supervisor along with the first update! synchronization.

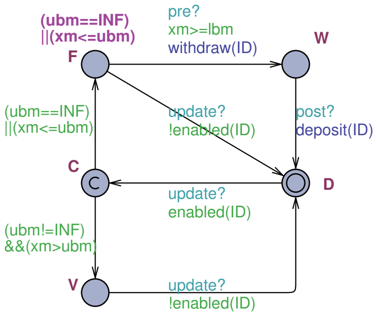


Figure 6. MasterTransition automaton

Transition template automata have names that mirror the adopted firing rule and the handled number of input arcs. In Figure 6 and Figure 7, the MasterTransition and WeakAnd2Transition templates are shown.

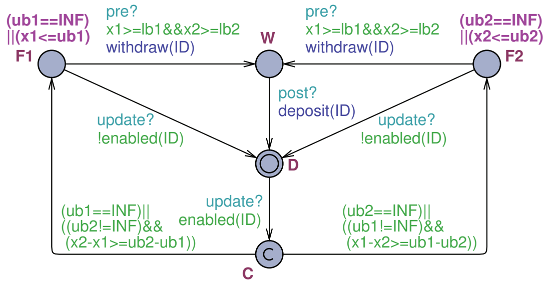


Figure 7. WeakAnd2Transition automaton

The automata in Figure 6 and Figure 7 illustrate some design principles, which were followed during library development. Each template is a decoration of the basic Transition template, according to the firing rule. A decision point C (committed location) is introduced where the automaton chooses to move to a firing location or to a V (Violation) location.

The MasterTransition is unique, whatever is the number of input arcs. It receives as parameters the transition ID and the clock xm and bounds, lbm and ubm, of the master arc. From C the automaton moves to V if the master arc is violated at the enabling time of the transition. Otherwise, it reaches the F location. It is worth to recall that from both urgent and committed locations an automaton has to exit immediately without passage of time. Committed locations, though, have greater priority than urgent locations.

In Figure 7, one can see two firing locations: F1, reached when the upper bound of first interval is found to be the maximum among the two intervals at the enabling time of transition, and F2 where the upper bound of the second interval is the maximum. From the guards of edges linking F1 or F2 to W, it is possible to see the *and* condition: only when both intervals are temporally ready, the transition can complete its firing.

Generally speaking, at the enabling time of a transition, for each input arc the two quantities can be evaluated: $lb_i – x_i$, which is the *time to start* of the relevant interval, and $ub_i – x_i$, which is the *time to finish* of the arc. In location C of Figure 7, if ub1−x1>=ub2−x2 it means that the maximum upper bound comes from the first arc. The relationship can be rewritten as x2−x1>=ub2−ub1 and so forth.

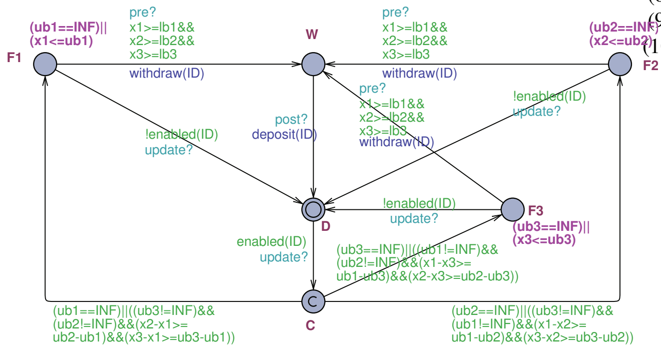


Figure 8. The WeakAnd3Transition automaton

Automata for two input arcs can easily be adapted to work with a greater number of input arcs. For example, Figure 8 shows the WeakAnd3Transition automaton. For details about all the other template automata of the developed UPPAAL library, the reader is referred to [7].

Figure 9 shows the system declaration section of the UPPAAL model corresponding to the translated flattened version of TSPN hypermedia example.

```
// Place template instantiations here.
t0=MasterTransition(T0, x[P1T0], 0, INF);
t1=MasterTransition(T1, x[P2T1], 65, INF);
t2=WeakAnd2Transition(T2, x[P6T2], 0, 0, x[P2T2], 150, 220);
t3=Transition(T3, x[P3T3], 18, 23);
t4=Transition(T4, x[P4T4], 50, 65);
t5=WeakAnd2Transition(T5, x[P9T5], 10, 15, x[P5T5], 90, 145);
t6=WeakAnd3Transition(T6, x[P9T6], 0, 0, x[P7T6], 0, 0, x[P12T6], 0, 0);
t7=MasterTransition(T7, x[P10T7], 10, 15);
t8=WeakAnd2Transition(T8, x[P7T8], 0, 0, x[P12T8], 0, 0);
t9=WeakAnd2Transition(T9, x[P11T9], 0, 0, x[P12T9], 0, 0);
t10=WeakAnd3Transition(T10, x[P10T10], 0, 0, x[P8T10],0,0,x[P12T10],0,0);
t11=MasterTransition(T11, x[P12T11], 1, 1);
// List one or more processes to be composed into a system.
system Supervisor, t0,t1,t2,t3,t4,t5,t6,t7,t8,t9,t10,t11;
```

Figure 9. System configuration of the hypermedia model in UPPAAL

V. MODEL CHECKING THE HYPERMEDIA MODEL

After translation in UPPAAL of the flattened version of TSPN model, it was possible to check its consistency synchronization. Table 1 summarizes the queries that were issued to the UPPAAL verifier for exhaustive property checking, and the gathered answers.

The actual declaration of the marking vector M was: int[0,10] M[P]={1,1,0,0,0,0,0,0,0,10,0,0,0};

Since no out-of-range assignment was signaled by UPPAAL during analysis, it was concluded that effectively the TSPN model is 10−bounded as expected.

TABLE 1. PROPERTY CHECKING OF THE HYPERMEDIA MODEL

	Query	Answer
(1)	A[] !deadlock	Property is satisfied
(2)	E<> t3.W	Property is satisfied
(3)	t3.W-->(M[0]==1 && M[1]==1 && M[4]==0 && M[5]==0 && M[6]==0 && M[7]==0 && M[8]==0 && M[9]==10 && M[10]==0 && M[11]==0 && M[12]==0)	Property is satisfied
(4)	t1.W-->(M[9]==10 && M[7]==0 && M[8]==0 && M[10]==0)	Property is satisfied
(5)	t0.W --> t3.W	Property is not satisfied
(6)	t2.W --> t3.W	Property is satisfied
(7)	A[] t5.W imply y>=150 && y<=215	Property is satisfied
(8)	A[] t7.W imply z<=150	Property is satisfied
(9)	A[] t1.W imply x[P2T1]>=65	Property is satisfied
(10)	E<> t2.W && x[P2T2]>220	Property is not satisfied

A first concern (query (1)) was checking absence of deadlocks in all the states of the model state graph (a *safety* property). Operator A[] verifies that !deadlock is invariantly true in all the states of the model. Query (2) (*existential*) asks if there is at least one state of the state graph where transition t3, which concludes the whole presentation, fires (the automaton is found in the W location). Being known that t3 actually can fires, query

(3), based on the *leads-to* operator -->, checks if it is always true that starting from a state in which t3 fires, it always follows a state in which the initial marking of the model is reinstalled. Query (4) asks, similarly, that starting from a state in which t1 fires, i.e., the Again link was clicked, it always follows a state in which the atomic layer network, which plays the multimedia scenario (see places p7, p8, p9 and p10) reaches the home marking in which the marking of p9 is 10 (the number of couple of media objects to be played) and the rest of the subnet is reset. This property in turn ensures that at any time the link Again is requested, the subnet correctly starts from its home marking. Queries (5) and (6) check *liveness* properties. In particular, query (5) asks the verifier if starting from a firing of t0 (which begins the presentation) it always follows a firing of t3 (which concludes the entire presentation). Obviously, this query has a negative response because once started the presentation can be interrupted by an Again request possibly an infinity number of times. Of course, query (6) is always true: after a conclusion of the multimedia scenario, it always follows the concluding remark is rendered. Queries (7) and (8) are examples of *bounded liveness* property checking. Toward this, two extra clocks z and y (*decoration clocks*) were added to the model. Clock y is reset at each firing of t0 or t1, which starts a new presentation, whereas clock z is reset at each firing of t4, which begins the multimedia scenario of the atomic subnet. All of this was achieved by adding a few instructions to the deposit() function. Query (7) asks if it is always true that a state in which t5 fires implies that a number of time units between 150 and 225 (including the audio presentation in place p4, which lasts in [50,65] time units) elapse (an expectation). Query (8) checks if it is always true that a state in which t7 fires implies that clock z is less than or equal to 150 time units. Query (9) checks if invariantly, i.e., in all states in which t1 fires (the Again link was requested), the arc clock x[P2T1] has always a value not less than 65. This property guarantees that, as in the requirements of the hypermedia model, the Again link cannot be issued before at least 65 time units are elapsed. Finally, query (10) verifies that in no case the arc clock x[P2T2] can be greater than 220 at the time in which t2 fires. This property ensures that, following a termination of the multimedia scenario, the Next link at most after 70 time units, that is after 220 time units, is automatically invoked.

Due to answers collected in Table 1, the TSPN hypermedia model was found correct functionally and temporally. The experimental work was carried out on a Win7, Intel Core i3, 4GB, 2.13 GHz. To give an idea of the efficiency of the achieved implementation, any query in Table 1 ends in about 5 seconds.

VI. Conclusion And Future Work

This paper proposes a methodology for modeling and analysis of multimedia/hypermedia documents, which is based on the Time Stream Petri Net formalism [10]. A TSPN model is translated into UPPAAL with the help of a developed reusable library of timed automata [7]. All of this enables synchronization consistency and temporal properties of the multimedia document to be verified through model checking.

Prosecution of the work aims to:

- Automating the translation from TSPN to UPPAAL, by completing an extension of the TPN/Designer toolbox [5] so as to graphically drawing a TSPN model and then generating the corresponding XML UPPAAL code.
- Extending the approach so as to consider some high level or general purpose authoring language like SMIL and converting an initial specification of an interactive multimedia document into the terms of TSPN for subsequent thoroughly exhaustive verification.

Acknowledgments

Authors are grateful to Christian Nigro for his contribution in the development/analysis of the hypermedia modeling example.

References

[1] G. Behrmann, A. David, and K.G. Larsen, “A tutorial on UPPAAL”, in *Formal Methods for the design of real-time systems, LNCS 3185*, Springer, pp. 200-236, 2004.

[2] B. Berthomieu and M. Diaz, "Modeling and verification of time dependent systems using Time Petri Nets”, *IEEE Trans. on Soft. Eng.*, **17**(3):259–273, 1991.

[3] S. Bouyakoub and A. Belkhir, “Towards a formal approach for modeling SMIL documents”, *IJSSST*, **10**(2):1-12, 2009.

[4] M. Boyer and M. Diaz, “Non equivalence between time Petri nets and time stream Petri nets”, in *Proc. of 8th Int. Workshop on Petri Nets and Performance Models*, pp. 198–207, 1999.

[5] L. Carullo, A. Furfaro, L. Nigro, and F. Pupo, “Modeling and simulation of complex systems using TPN DESIGNER”, *Simulation Modeling Practice and Theory*, **11**, pp. 503-532, 2003.

[6] F. Cassez and O. H. Roux, “Structural translation from time Petri nets to timed automata”, *J. of Systems and Software*, vol. 79, pp. 1456–1468, 2006.

[7] F. Cicirelli, A. Furfaro, and L. Nigro, “Model checking time-dependent system specifications using time stream Petri nets and UPPAAL”, *Applied Mathematics and Computation*, **218**(16):8160-8186, 2012.

[8] J.-P. Courtiat, Diaz M., R.C. De Olieira, and P. Senac, “Formal models for the description of timed behaviors of multimedia and hypermedia distributed systems”, *Computer Communications*, vol. 19, pp. 1134-1150, 1996.

[9] J.-P. Courtiat and R.C. De Olieira, “Proving temporal consistency in a new multimedia synchronization model”, in *Proc. ACM Multimedia,* 1996.

[10] M. Diaz and P. Sénac, “Time Stream Petri Nets, a model for multimedia streams synchronization”, in *Proc. of Multimedia Modeling (MMM’93)*, 1993.

[11] W. Liu and Y. Du, “Modeling multimedia synchronization using Petri nets”, *Inf. Technology Journal*, vol. **8**, pp. 1054-1058, 2009.

[12] P. Merlin and D.J. Farber, "Recoverability of communication protocols”, *IEEE Trans. Commun.*, **24**(9):1036-1043, 1976.

[13] T. Murata, “Petri nets: properties, analysis and applications”, in *Proc. of the IEEE*, **77**(4):541-580, 1989.

[14] P. Sénac, M. Diaz, A. Léger, and P. de Saqui-Sannes, "Modeling logical and temporal synchronization in hypermedia systems”, *IEEE J. on Selected Areas in Comm.*, **14**(1):84–103, 1996.

[15] P.N.M. Sampaio and J.-P. Courtiat, “An approach for the automatic generation of RT-LOTOS specifications from SMIL 2.0 documents”, *J. of Brazilian Computer Society*, **9**(3), 2004.

[16] SMIL on-line, http://www.w3.org/TR/smil20 [retrieved: September, 2012].

Development of Modified Ant Colony Optimization Algorithm for Compliant Mechanisms

Se-Chan Kim, Dae-Ho Chang, Kwang-Seon Yoo
Department of Mechanical Engineering
Graduate School, Hanyang University
Seoul, Republic of Korea
{sechanism, dhc, ksy}@hanyang.ac.kr

Seog-Young Han
School of Mechanical Engineering
Hanyang University
Seoul, Republic of Korea
syhan@hanyang.ac.kr

***Abstract*—A compliant mechanism is a mechanism that obtains its mobility or force from the flexibility and elastic deformation of its components. Topology for compliant mechanism is very sensitive and can be obtained very variously according to topology optimization methods and computation conditions. A modified ant colony optimization algorithm is suggested for compliant mechanisms in order to obtain a stable and robust optimal topology. The modified ant colony optimization algorithm was applied for both linear and geometrically nonlinear compliant mechanisms. Using three kinds of objective functions commonly used, optimized topologies were compared for compliant mechanisms. And it was concluded that the modified ant colony optimization algorithm can successfully be applied for linear and geometrically nonlinear compliant mechanisms, and the algorithm provides very stable and robust topologies.**

Keywords-Compliant mechanism; Modified Ant Colony Algorithm (MACO); Topology optimization; Geometrically nonlinear.

I. INTRODUCTION

Topology optimization has been applied for various linear structural problems so far [1][2][3][4][5][6]. However, when a very large load is applied or structural deformation is very large, geometrically nonlinearity may be occurred due to mechanical conditions. In order to obtain more useful and valuable optimal topology of a structure satisfying the given constraints, the above nonlinearity should be considered in analysis and design.

Buhl et al. [7] has performed stiffness designs of geometrically nonlinear structures using topology optimization based on the density method. Pedersen et al. [8] has performed linear and geometrically nonlinear topology optimizations based on the solid isotropic material with penalization (SIMP) method for large displacement compliant mechanisms. Bruns and Tortorelli [9] also carried out linear and geometrically nonlinear topology optimization for nonlinear structures and compliant mechanisms.

Recently, Kaveh [10] suggested ant colony optimization (ACO) algorithm for structural models to find the stiffest structure with a certain amount of material, based on the element's contribution to the strain energy. The results showed that ACO can be a suitable tool to handle the problem as an on-off discrete optimization. However, the field of topology optimization for compliant mechanisms is rarely new in ACO algorithms since researches on compliant mechanisms have not been done so far.

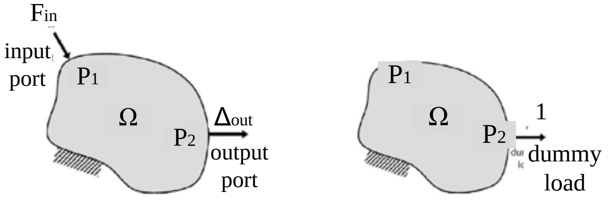
In this study, a new topology optimization algorithm based on ACO algorithm is developed for a compliant mechanism for the first time implemented with a filter scheme [11]. Distribution of material is expressed by density of each element to apply ACO algorithm. Three kinds of objective function are examined to obtain stable and robust topology, it is found that the developed topology scheme is very effective and applicable in a compliant mechanism topology optimization problems and mutual potential energy (*MPE*) / strain energy (*SE*) type of objective function was the best through the comparison of the results of the linear and geometrically nonlinear cases.

This paper is organized as follows. Formulation of a Force Compliant Mechanism in Section II. In Section III, we introduce the proposed method which is called MACO. Section IV contains Numerical Examples, and Section V, concludes the paper.

II. FORMULATION OF A FORCE COMPLIANT MECHANISM

A compliant mechanism [12] is a mechanism that obtains its mobility or force from the flexibility and elastic deformation of its components. Fig. 1(a) [13] shows design domain Ω with the input force and the desired output displacement Δ_{out}. P_1 and P_2 represent the input and output ports, respectively, and Δ_{out} represents the desired output displacement at the output port. Fig. 1(b) shows the analytical conditions of *SE* for evaluating the stiffness and *MPE* for evaluating flexibility of a compliant mechanism.

Three kinds of objective functions are usually used for compliant mechanisms. *MPE* [13] is used for maximizing displacement at the output port. One of multicriteria objective functions, *wMPE*+(1-*w*)*SE*, is used for considering



(a) Input force (b) Dummy load

Figure 1. Analysis load case

both flexibility and stiffness of compliant mechanisms with weighting factor w. A ratio [14] of MPE to SE is employed for maximizing displacement at the output port considering both flexibility and stiffness of compliant mechanisms. In this paper, the above objective functions were used, and topology optimization based on the MACO method was performed for a force compliant mechanism.

Firstly, we explain how to calculate sensitivity number for the case that *MPE/SE* is employed as the objective function. The sensitivity number can be calculated by adapting the method suggested by Ansola [13]. The finite element equation for *MPE* can be expressed using (2) as follows;

$$MPE = \Delta_{out} = \{D_2\}^T [K]\{D_1\} \tag{2}$$

where [*K*] is the generalized stiffness matrix, $\{D_1\}$ is the nodal displacement vector due to the input load and $\{D_2\}$ is the displacement vector due to the unit dummy load placed at the output port, respectively. *MPE* means that the displacement at the output port when F_{in} at the input port is applied.

The finite element equation for *SE* can be expressed using (3) as follows. *SE* means the strain energy stored when the unit dummy load applied at the output port.

$$SE = \frac{1}{2}\{D_2\}^T [K]\{D_2\} \tag{3}$$

If the i -th element is added to the previous topology, the change in *MPE* can be calculated as follows (4).

$$\Delta MPE = -\{D_1\}^T [\Delta K]\{D_2\} \tag{4}$$

Similarly, when the i -th element is added to the previous topology, the change in *SE* can be obtained as follows (5).

$$\Delta SE = -\frac{1}{2}\{D_2\}^T [\Delta K]\{D_2\} \tag{5}$$

Element addition affects the generalized stiffness matrix, and we can calculate the change in the generalized stiffness matrix due to the i -th element addition as follows (6);

$$[\Delta K] = [K'] - [K] = [K]_i \tag{6}$$

where [*K'*] is the generalized stiffness matrix after the i -th element is added, [*K*] is the generalized stiffness matrix before the element is added, and $[K]_i$ is the generalized stiffness matrix of the added element. From (4), (5), and (6), the following relations (7) can be obtained;

$$\begin{aligned} \Delta MPE &= -\{D_1\}_i^T [K]_i \{D_2\}_i \\ \Delta SE &= -\frac{1}{2}\{D_2\}_i^T [K]_i \{D_2\}_i \end{aligned} \tag{7}$$

where $\{D_1\}_i$, and $\{D_2\}_i$ represent the generalized displacement vectors due to the *i*-th element addition.

In this paper, in the case that *MPE* is employed as an objective function, the sensitivity number is calculated as follows (8).

$$\alpha_i = -\{D_1\}_i^T [K]_i \{D_2\}_i \tag{8}$$

Secondly, in the case that *wMPE*+(1-*w*)*SE* is used as an objective function, the sensitivity number is calculated as follows (9).

$$\alpha_i = w\Delta MPE + (1-w)\Delta SE \tag{9}$$

Finally, the sensitivity number for the *i*-th element can be obtained by differentiating the objective function (10).

$$\alpha_i = \frac{(\Delta MPE)(SE) - (MPE)(\Delta SE)}{(SE)^2} \tag{10}$$

The above sensitivity numbers can be used for adding, eliminating, or transforming elements in topology optimization for a force compliant mechanism.

III. Modified Ant Colony Optimization Algorithm

If ACO algorithm is applied for topology optimization of geometrically nonlinear compliant mechanisms, a critical problem can be encountered. This method can provides a stable topology in the case of a high target volume in structural topology optimization. However, in the case of a low target volume, the asymmetry of stiffness matrix becomes very severe since the topology consisted of solid elements significantly lose the symmetry of structure. It causes poor accuracy of the solution since ill-condition might be produced. In order to overcome the above weakness, it is necessary to define a design variable such as continuously distributed density in the previous studies [15] for topology optimization. Therefore, a MACO algorithm is suggested in order to remedy the weakness of the ACO algorithm.

The governing equations of the ACO algorithm [10] are briefly described as follows. Contribution of each element *i* into the overall objective of the problem, which is analog to the pheromone trail of a segment of a route, is here denoted by $\tau_i(t)$. The parameter *t* represents the time of deployment of ants which is equivalent to the cycles of iteration within the algorithm. Inspired by the procedure employed in TSP [16], and ignoring the effect of the local heuristic values, the ant decision index $a_i(t)$ can be written as (11);

$$a_i(t) = \frac{[\tau_i(t)]^\alpha}{\sum_{j=1}^{M} [\tau_j(t)]^\alpha} \tag{11}$$

where α is a parameter that controls the relative weight of the pheromone trail, M is the number of finite elements and t is an indication of the present cycle which is analogous to the t-th time of deploying our ants. Note that here the probability of an element being chosen by a typical ant is the same as the decision index as defined in (11).

After completion of a cycle of designs by all ants, each ant k deposits a quantity of pheromone $\Delta\tau_i^k$ on each element based on its relative objective function, as shown below, which is an index of the performance of the element, i.e. for a better design a larger amount of pheromone is deposited (12);

$$\Delta\tau_i^k = \frac{\left(U_i^k\right)^\lambda}{\sum_{j=1}^{M}\left(U_j^k\right)^\lambda} \tag{12}$$

where U_i^k is the objective function in each element of design and the exponent λ is a tuning parameter for improvement of performance of the algorithm and its convergence.

The amount of pheromone in each element is due to addition of new pheromone as well as evaporation which is implemented within the algorithm via the following rule (13).

$$\tau_i\left(t+1\right) = \gamma\tau_i\left(t\right) + \Delta\tau_i \tag{13}$$

where $\Delta\tau_i = \sum_{k=1}^{m}\Delta\tau_i^k$ and m is the number of ants used in each cycle. The rate of evaporation coefficient $\gamma \in \left[0,1\right]$ is applied for taking into account the pheromone decay to avoid quick convergence of the algorithm towards a suboptimal solution.

The main difference between ACO and MACO is to use a new continuous variable, which is called "fitness" in finite element analysis (FEA), instead of the positions of the ants. Fitness is defined by the ratio of the summation of the ants number passed each element to the number of inner loop. It can be expressed as (14);

$$fitness_i = \frac{\sum_{iter=1}^{N}(A_{iter})_i}{N}, \quad (A = 1\ or\ 0) \tag{14}$$

where A represents the existence of ant (existence = 1, no existence = 0). N is the number of iteration, i is the element number. Then, the calculated fitness in iteration process is used to FEA as a design variable in cycle process.

In this study, a new change of objective function α_i^{new} is defined as (15) in order to accelerate the convergence rate of the MACO;

$$\alpha_i^{new} = fitness_i \times \alpha_i \tag{15}$$

where i indicates each element. The resized α_i is applied to the global update equation (12) and the U_i in (12) is replaced by α_i for geometrically nonlinear compliant mechanism (16).

$$\Delta\tau_i^k = \frac{\left(\alpha_i^k\right)^\lambda}{\sum_{j=1}^{M}\left(\alpha_j^k\right)^\lambda} \tag{16}$$

In addition, it is very important to reduce computation time for geometrically nonlinear compliant mechanism. Convergence rate of the ACO is dependent on the ant decision index $a_i\left(t\right)$, so that the parameters of α, λ, γ, affect very much on the convergence rate. It has been suggested that convergence rate can also be accelerated by resizing pheromones newly added to the solution found by rank-based ant system or elite ants [17] [18] [19].

$$\Delta\tau_i = \left(1 - \Delta\tau_{\min}^{new}\right)\frac{\Delta\tau_i^{new}}{\Delta\tau_{\max}^{new}} + \Delta\tau_{\min}^{new} \tag{17}$$

where $\Delta\tau_i^{new} = \sum_{k=1}^{m}\Delta\tau_i^k$, $\Delta\tau_{\min}^{new}$ is 0.0001 and $\Delta\tau_{\max}^{new}$ is the maximum value of pheromone trail at each iteration. The resized $\Delta\tau_i$ provides the improved effect of acceleration rate on convergence, and overcome numerical singularity occurred on the low-density region. Also it gives the selection possibility of the elements, which have been removed because of low deposited pheromone in the previous iterations, in the following iterations. The resized $\Delta\tau_i$ is applied to the global update equation (13). The optimization process using MACO can be depicted as a flowchart shown in Fig. 2. The procedure can be outlined as follows:

1. Specify ACO control parameters (α, γ), tune the parameter λ, and assign an initial pheromone trailing value on each element.
2. Create the initial design using a sequence of random selection by spreading pheromone trailing uniformly in the design space
3. Calculate α_i for each element using compliant mechanism finite element analysis with a mesh-independency filter scheme to suppress checker-board pattern.
4. Move ants probabilistically according to (11).
5. Deposit new pheromones at the present locations where ants move using (16).
6. After completion of a cycle, resize pheromones using (17).
7. The amount of pheromone on each element is updated according to (13).
8. Steps 3 through 7 are repeated until convergence criterion is satisfied using (18).

$$error = \frac{\left|\sum_{i=1}^{N'}\left(\alpha_{c-i+1} - \alpha_{c-N'-i+1}\right)\right|}{\sum_{i=1}^{N'}\alpha_{c-i+1}} \le \varphi \qquad (18)$$

where, c is the current iteration number, φ is an allowable convergence error, and N' is the integral number which results in a stable convergence in at least ten successive iterations**.**

IV. Numerical Examples

A. Force invertng mechanism

A displacement generator in compliant mechanisms having dimensions of 40 μm × 20 μm × 7 μm is subjected to input force F_{input} = 2 mN with the spring constants $k_{input, output}$ = 1 mN/μm at input and output ports as shown in Fig. 3. Design domain is divided into 80 × 40 by four node rectangular element. The material is assumed to have Young's modulus of 100 GPa and Poisson's ratio of 0.3. The coefficients of MACO are defined as α = 1, λ = 2, ρ = 0.8. Allowable convergence error, τ is set to be 0.001. The objective is to obtain a stiffest structure under a volume constraint of 20% of the original volume.

Topologies for linear and geometrically nonlinear cases with *MPE* and *wMPE*+(1-*w*)*SE* where *w* = 0.8 as objective functions are shown in Fig. 4 and Fig. 5, respectively. And topologies for linear and nonlinear cases with *MPE/SE* are shown in Fig. 6. The displacements of the optimal topologies for three kinds of objective functions are compared in Table 1. It can be found that the displacements of *MPE* type are the largest in linear and nonlinear cases. In other cases of *MPE/SE* type for both linear and nonlinear are the smallest. Also, the joint part is a little more reinforced in the nonlinear case than the linear case.

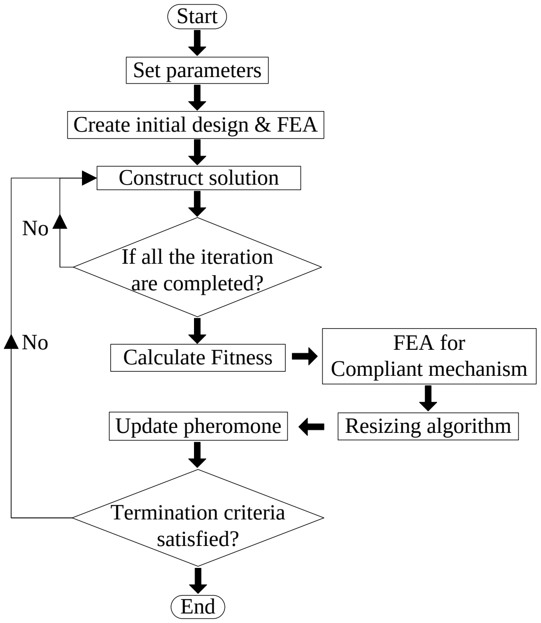

Start
Set parameters
Create initial design & FEA
Construct solution
No
If all the iteration are completed?
No
Calculate Fitness
FEA for Compliant mechanism
Update pheromone
Resizing algorithm
Termination criteria satisfied?
End

Figure 2. Flowchart of the MACO for compliant mechanisms

From the results of the example, topology at the joint parts is not connected when *MPE* implemented with the MACO is employed as an objective function. Even though topology at the joint part is connected each other when *wMPE*+(1-*w*)*SE* where *w* = 0.8 is used as objective function, there appears checkerboard pattern, and the topology may change according to weighting factor. As seen here, topology at the joint part is firmly connected each other and a stable topology can be obtained when *MPE/SE* is used as objective function. Therefore, it is concluded that *MPE/SE* is very suitable among three kinds of objective functions for designing compliant mechanisms.

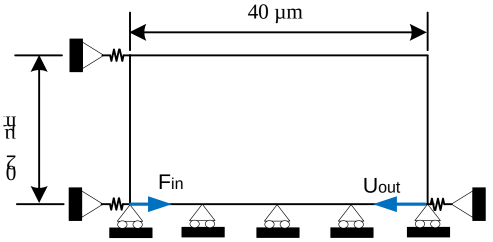
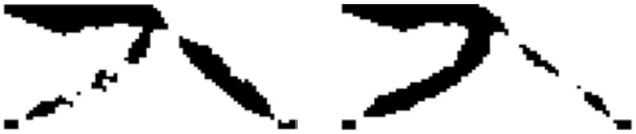
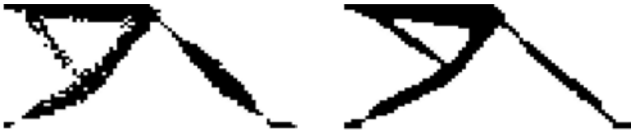


Figure 3. Design domain of a force inverter



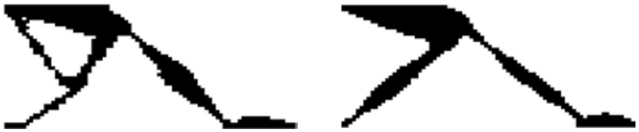
(a) Linear case (b) Geometrically nonlinear case

Figure 4. Optimal topology using *MPE*



(a) Linear case (b) Geometrically nonlinear case

Figure 5. Optimal topology using *wMPE*+(1-*w*)*SE* (*w* = 0.8)



(a) Linear case (b) Geometrically nonlinear case

Figure 6. Optimal topology using *MPE/SE*

TABLE I. Displacements of linear and nonlinear optimal topologies

Cases	***MPE***	***wMPE*+(1-*w*)*SE***	***MPE/SE***
Linear	−2.972 μm	−1.606 μm	−0.212 μm
Geometrically nonlinear	−3.067 μm	−2.426 μm	−0.587 μm

B. Gripper mechanism

A gripper mechanism having dimensions of 40 μm × 15 μm × 1 μm is subjected to input force F_{input} = 1 mN with the spring constants $k_{input, output}$ = 0.1 mN/μm at input and output ports as shown in Fig. 7. Design domain is divided into 120 × 45 by four node linear finite elements. The material is assumed to have Young's modulus of 100 GPa and Poisson's ratio of 0.3. The coefficients of MACO are defined as α = 1, λ = 2, ρ = 0.8. Allowable convergence error, τ is set to be 0.001. The objective is to obtain a stiffest structure under a volume constraint of 20% of the original volume.

Topologies for linear and geometrically nonlinear cases with *MPE* and *wMPE*+(1-*w*)*SE* where *w* = 0.8 as objective functions are shown in Fig. 8 and Fig. 9, respectively. And topologies for linear and nonlinear cases with *MPE/SE* are shown in Fig. 10. The displacements of the optimal topologies for three kinds of objective functions are compared in Table 2.

It can be found that the displacements of *MPE* type are the largest in linear and nonlinear cases. In other cases of *MPE/SE* type for both linear and nonlinear are the smallest.

From the results of the example, topology at the joint parts is not connected when *MPE* for nonlinear case implemented with the MACO is employed as an objective function. Even though topology at the joint part is connected each other when *wMPE*+(1-*w*)*SE* where *w* = 0.8 is used as objective function, there appears that topology of output part is unstable compared to the other objective function. Therefore, it is concluded that *MPE/SE* is very suitable among three kinds of objective functions for designing compliant mechanisms.

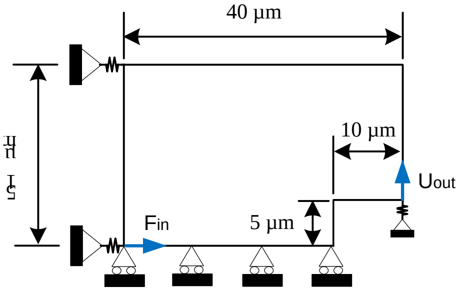
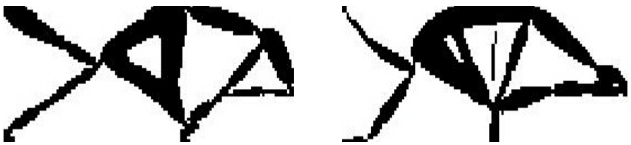
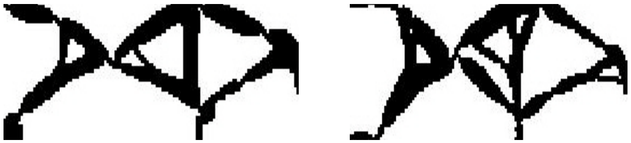


Figure 7. Design domain of a gripper mechanism



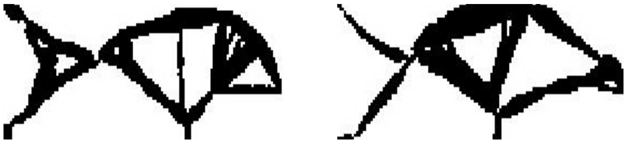
(a) Linear case (b) Geometrically nonlinear case

Figure 8. Optimal topology using *MPE*



(a) Linear case (b) Geometrically nonlinear case

Figure 9. Optimal topology using *wMPE*+(1-*w*)*SE* (*w* = 0.8)



(a) Linear case (b) Geometrically nonlinear case

Figure 10. Optimal topology using *MPE/SE*

TABLE II. DISPLACEMENTS OF LINEAR AND NONLINEAR OPTIMAL TOPOLOGIES

Cases	*MPE*	*wMPE*+(1-*w*)*SE*	*MPE/SE*
Linear	−1.922 μm	−1.703 μm	−1.426 μm
Geometrically nonlinear	−2.45 μm	−2.43 μm	−2.28 μm

V. CONCLUSION AND FUTURE WORK

A. Conclusion

In this study, the MACO algorithm has been suggested for compliant mechanism problems and a compliant mechanism using three kinds of objective functions. From the results of examples, the following conclusions are obtained.

(1) It is verified that the MACO algorithm can successfully be applied for a compliant mechanism, and provides stable and robust optimal topology.

(2) MACO algorithm is suggested for applying it for compliant mechanisms in order to obtain a stable topology since ACO algorithm might severely provide asymmetric stiffness matrix due to the characteristics of stochastic methods.

(3) It is found that *MPE/SE* considering flexibility and stiffness together is the most suitable for objective function among three kinds of objective functions for designing compliant mechanisms.

The topology optimization using the ACO could be extended to more complicated thermally actuated compliant mechanism specifications such as electro-thermal actuators subjected to non-uniform temperature fields actuated by Joule heating.

REFERENCES

[1] M. P. Bendsøe and N. Kikuchi, "Generating optimal topologies in structural design using a homogenization method," Comput. Meth. Appl. Mech. Eng., vol. 71, pp. 197-224, November 1988.

[2] H.P. Mlejek and R. Schirmacher, "An Engineer's Approach to optimal Material Distribution & Shape Finding," Comput. Meth. Appl. Mech. Eng., vol. 106, pp. 1-26, July 1993.

[3] O. M. Querin, G. P. Steven, and Y. M. Xie, “Evolutionary structural optimization (ESO) using a bidirectional algorithm,” Eng. Comput., vol. 15, pp. 1031-1048, 1998.

[4] Q. Q. Liang and G. P. Steven, Performance-based optimization of structures:Theory and applications, Spon press, Taylor and Francis Group, London, New York 2005.

[5] J. A. Sethian and A. Wiegmann, “Structural boundary design via level set and immersed interface methods,” J. Comput. Phys., vol. 163, pp. 489-528, September 2000.

[6] T. Belytschko, S. P. Xiao, and C. Parimi, “Topology optimization with implicit functions and regularization,” Int. J. Numer. Meth. Eng., vol. 57, pp. 1177-1196, June 2003.

[7] T. Buhl, C. B. W. Pedersen, and O. Sigmund, “Stiffness design of geometrically nonlinear structures using topology optimization,” Struct. Multidisc. Optim., vol. 19, pp. 93-104, April 2000.

[8] C. B. W. Pedersen, T. Buhl, and O. Sigmund, “Topology synthesis of large-displacement compliant mechanisms,” Int. J. Numer. Meth. Eng., vol. 50, pp. 2683-2705, April 2001.

[9] T. E. Burns and D. A. B. Tortorelli, “Topology optimization of non-linear elastic structures and compliant mechnisms,” Comput. Methods Appl. Mech. Engrg., vol. 190, pp. 3443-3459, March 2001.

[10] A. Kaveh, B. Hassani, S. Shojaee, and S. M. Tavakkoli, “Structural topology optimization using ant colony methodology,” Eng. Struct., vol. 30, pp. 2259-2565, September 2008.

[11] X. Huang and Y. M. Xie, “Convergent and mesh-independent solutions for the bi-directional evolutionary structural optimization method,” Finite. Elem. Anal. Des., vol. 43, pp. 1039-1049 October 2007.

[12] L. Howell, Compliant Mechanisms, Wiley, New York, 2001.

[13] R. Ansola, E. Vegueria, J. Canales, and J. A. Tarrago, “A simple evolutionary topology optimization procedure for compliant mechanism design,” Finite. Elem. Anal. Des., vol. 44, pp. 53-62 December 2007.

[14] M. Motiee, “Development of a Novel Multi-disciplinary Design Optimization Scheme For Micro Compliant Devices,” A Thesis for the degree of Ph. D, UNIVERSITY OF WATERLOO, Canada, 2008.

[15] I. Kosaka and C. C. Swan, “A symmetry reduction method for continuum structural topology optimization,” Comput. Struct., vol. 70, pp. 47-61, January 1999.

[16] M. Dorigo, “Optimization, Learning and Natural Algorithms (in Italian),” A Thesis for the degree of Ph. D, Politecnico di Milano, Italie, 1992.

[17] S. D. Shtovba, “Ant Algorithms: Theory and Applications,” Program. Comput. Softw., vol. 31, pp. 167-178, 2005.

[18] T. Stützle and H. Hoos, “Improvement on the ant-system: Introducing MAX-MIN and system,” In proceedings of the international conference on artificial neural networks and genetic algorithms, 1997.

[19] B. Bullnheimer, R. F. Hartl, and C. A. Strauss, “New rank-based version of the ant system: a computational study,” Technical Report POM-03/97, Institute of Management Science. University of Vienna, Austria, 1997.

Cooperative c-Marking Agents for the Foraging Problem

Ouarda Zedadra, Hamid Seridi
Science and Technology of Information and
Communication Laboratory (LabSTIC)
8 May 1945 University
Guelma 24000, Algeria
{zedadra_nawel1, seridihamid}@yahoo.fr

Nicolas Jouandeau
Advanced Computing Laboratory of Saint-Denis
(LIASD)
Paris 8 University
Saint Denis 93526, France
n@ai.univ-paris8.fr

***Abstract*— We consider the problem of foraging with multiple agents, in which agents must collect disseminate resources in an unknown and complex environment. So far, reactive multi-agent systems have been proposed, where agents can perform simultaneously exploration and path planning. In this work, we aim to decrease exploration and foraging time by increasing the level of cooperation between agents; to this end, we present in this paper a novel pheromone modeling in which pheromone's propagation and evaporation are managed by agents. As in c-marking agents, our agents are provided with very limited perceptions, and they can mark their environment. Simulation results demonstrate that the proposed model outperforms the c-marking agent-based systems in a foraging mission.**

Keywords- reactive agents; foraging task; digital pheromone; APF construction

I. INTRODUCTION

Foraging is a task that lends to multi-robots systems that can beat single robot systems in such a task. On the other side, the possible profit of multi-robots systems is conditioned by the level of cooperation [2]. Swarm intelligence is the study of collective complex and intelligent behaviors observed in natural systems where global swarm behaviors emerge as a result of local interactions between agents and global interactions between agents and environment [3]. Foraging is, therefore, a benchmark problem within swarm robotics [4]. A particularly interesting situation problem is when foraging robots have no a priori information about the locations of objects in unknown and complex environment. As wider searching spaces need more scalable and reliable solutions, distributed cooperative multi-robots systems are much adopted to achieve foraging missions.

Synthetic pheromones are one of the most popular swarm techniques that provide interesting solutions to problems such foraging [5]; most of these solutions create local minima that can lead the multi-robots system to fail. In [1], a pheromone is modeled as a static piece dropped and picked up by agents. The cooperation between agents is managed with the c-marking agents' algorithm. The pheromone has no propagation properties, that is minimizing the level of cooperation between agents. This hypothesis is very promising to achieve rapidly tasks such as foraging [6].

This paper presents a novel pheromone modeling that aims at increasing the level of cooperation between agents to achieve rapidly the foraging task. To this end, we present in this paper, a new behavioral model and an extension version of the c-marking agents' algorithm. This new behavioral model handles specific situations such as the presence of two resources in neighboring cells. Through simulation tests, the system is compared with the original one [1] in terms of the number of iterations required for achieving the foraging task.

The rest of paper is organized as follows. In Section 2, we discuss related works. A new pheromone modeling, behavioral model and extended algorithm (cooperative c-marking agent algorithm) are given in Section 3. Section 4 describes the simulation environment and an experimental comparison between the original c-marking and c-marking enhanced algorithm. Section 5 concludes our research.

II. RELATED WORK (PHEROMONE BASED TECHNIQUES FOR FORAGING)

Foraging is a benchmark problem for robotics, especially for multi robot systems [2]. It is the act of searching for any objects and collecting them at a storage point which is called base. Ostergaard et al. define it as "a two-step repetitive process in which (1) robots search a designated region of space for certain objects, and (2) once found these objects are brought to a goal region using some form of navigation" [7].

A wide range of approaches has been adopted to suggest solutions to the foraging problem in unknown environments. Most of them focus on examples of multi-robot foraging from within the field of swarm robotics. Three strategies for cooperation very known in this field are: information sharing [8], physical cooperation, which can be a cooperative grabbing [9][10], or a cooperative transport [11][12][13]. In multi-robot foraging it is well know that overall performance does not increase with increasing team size [14][15][16], division of labor in ant colonies has been well studied and there was a proposition of threshold model [17][18], some other works concentrate on individual adaptation and division of labor in ants that allow a swarm of robots to self-organize [19][20][21]. Pheromone based techniques inspired from ants are used for foraging with robots [22][23][24], where agents drop a quantity of pheromones in their environment in order to build gradients from sources to the base. This approach has some drawbacks, such as the computation of propagation and evaporation dynamics, and each agent needs specific mechanisms or materials that allow him to get back home. Panait and Luke [5] and Resnick [24] propose the use of a second pheromone diffusion from the base in order to avoid this last problem. In the same time, this solution can create new local minima.

An original approach has been proposed in [1] that allows agents to build optimal paths for foraging using only reactive agents, which have limited information about their environment. To keep track of the sources found and to build trails between sources and the base, agents drop a quantity of pheromones in their environment.

In this paper, we present a novel extension of the c-marking agents' algorithm, in order to increase the level of cooperation between agents.

III. COOPERATIVE C-MARKING AGENTS

The proposed multi agents system has the same properties of agents defined in [1]; the system is defined as a set of objects, which are static obstacles, sources and other agents, sources have fixed positions and agents are moving in the environment to achieve their own task:

A. *The pheromone model*

Two kinds of the pheromone's model are used in most of the works cited above. The first one integrates the management of the two modules propagation and evaporation in the pheromone, which is a complex task that causes in some cases building of local minima, the second one use the pheromone as a piece that does not propagate in the environment, and that can be picked up by agents when all is finished.

The proposed pheromones' model combines between the two properties cited before, thus the pheromone is considered as a piece that can be dropped and picked up by the agent, it has propagation and evaporation properties that are managed by the agent. The agent will creates a maximal trail (deposits a diffusible pheromone), if the quantity of resources is important for more attraction and recruitment of agents to the trails. If the quantity of resources is (or becomes) less than a fixed minimum quantity; the agent creates a minimal trail (deposits a non-diffusible pheromone) to avoid the attraction of other agents. This new modeling is shown by figure 1.

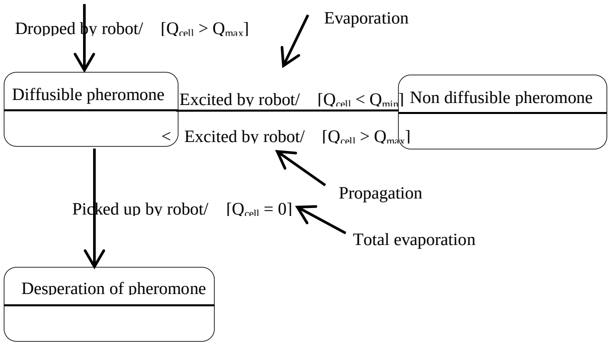

Dropped by robot/ [$Q_{cell} > Q_{max}$]
Evaporation
Diffusible pheromone
Excited by robot/ [$Q_{cell} < Q_{min}$]
Non diffusible pheromone
<
Excited by robot/ [$Q_{cell} > Q_{max}$]
Propagation
Picked up by robot/ [$Q_{cell} = 0$]
Total evaporation
Desperation of pheromone

Figure 1. Pheromone modeling

The fact that the pheromone is managed by the agents (dropped, picked, propagated and evaporated); there will be no doubt that a non-operational trails steel existing between the base and an exhausted source.

B. *The environment model*

The environment is modeled as a squared grid with variable size that have resources in multiple locations, these locations are scattered randomly, and they are unknown by the agents; each location has a given quantity of resources. Cells in the environment can:

- Be an obstacle (grey color);
- Contain a resource (green color) with a limited quantity Q max
- Be the base (red color), positioned in all simulations in the environment's center, and form the start point of all the agents;
- Contain an agent

C. *The agent model*

Agents have limited information about their environment; they occupy a cell, and each agent can:

- Move from a cell to another, which is not an obstacle in the four direction;
- Read and write values in the current cell;
- Perceive and read the values of the four neighboring cells, so he can detect resources, and he can load a quantity of resources according to Qte_{max}.

Increasing the level of cooperation between agents, and dealing with specific situations such as the two resource neighboring cells, can decrease dramatically the time of foraging. To achieve those goals, we address the new agent's behavior given by Figure 2 (Cooperative c-marking agents) and the enhanced algorithm corresponding to cooperative c-marking agents is given by the algorithm (extension of the c-marking agents) bellow:

Algorithm 1: Cooperative c-marking agents

```
SEARCH & CLIMB (Simonin & al, 2010)
If a resource is detected in a neighboring cell THEN
      move into that cell and execute LOADING
ELSE IF neighboring cells are colored and different from the
previous position THEN  move to highest-valued such cell,
ELSE execute EXPLORATION & APF CONSTRUCTION

LOADING
Pick up a quantity $Qte_{max}$ of resource ;
IF the cell is not exhausted of resources THEN
  IF $Q_{cell}$ > = $Q_{max}$  THEN
    IF the cell is colored THEN execute RETURN TO BASE
    ELSE execute RETURN & COLOR MAX TRAIL
   ELSE IF $Q_{cell}$  = $Q_{min}$  THEN
        IF the cell is colored THEN execute RETURN &
                                     ERASE MAX TRAIL          (1)
        ELSE execute RETURN & MIN TRAIL
    ELSE IF $Q_{cell}$ < $Q_{min}$  THEN
      IF the cell is colored  THEN execute RETURN TO
```

```
                                        BASE
        ELSE execute RETURN & COLOR MIN TRAIL

ELSE IF the cell is exhausted THEN
      IF (resource found in neighboring cells) and (exist trail)
              THEN
               Color current cell with trail color
              IF  $Q_{cell}$ > = $Q_{max}$  THEN  color the 4 neighboring
                  cells with light gray color and execute
                    RETURN TO BASE
             ELSE IF $Q_{cell}$  <=  $Q_{min}$  THEN execute RETURN
                                                  TO BASE
   ELSE IF (resource found in neighboring cells) and (not
       exist trail) THEN execute (1)
     ELSE IF the cell is colored THEN execute RETURN &
                           ERASE MIN TRAIL
     ELSE execute RETURN TO BASE ;

RETURN & COLOR MAX TRAIL
IF the base is reached THEN
            unload resource and execute SEARCH & CLIMB
ELSE move to a new neighboring cell with the least value
   Color the current cell in a trail color -dark gray color- (it
   will be propagates to the 4 neighboring cells with another
   color trail –light gray color-);

RETURN & COLOR MIN TRAIL (Simonin & al, 2010)
IF the base is reached THEN
             unload resource and execute SEACH & CLIMB
ELSE  move to a new neighboring cell with the least value
 Color the that cell in a trail color;

RETURN & ERASE MAX TRAIL
IF the agent is located at base THEN UNLOAD resource &
execute SEARCH & CLIMB
ELSE Move to a neighboring colored cell with the least value
Erase the color of the 4 neighboring cells to the default color;

RETURN & ERASE MIN TRAIL
IF the base is reached THEN UNLOAD resource & execute
SEARCH & CLIMB
ELSE IF the number of colored neighboring cells >=2 THEN
execute RETURN TO BASE
Else IF there is a colored cell with least value THEN
     move to it & erase the color to the default one.

RETURN TO BASE
If the base is reached THEN UNLOAD resource and execute
SEARCH & CLIMB
ELSE IF there is a neighboring colored cell with min value THEN
move to it and UPDATE VALUE
ELSE move to a cell with min value & UPDATE VALUE
```

In the foraging system shown in Figure 2, each robot search resource in a pseudo random walk, and then it brings the discovered resources to the base. In its way, the robot creates paths to keep track of resources found, and to attract more robots to the discovered resource cell. Paths are of two kinds; max trails, which are formed by a diffusible digital pheromones (i.e. these are larger paths, and they are diffusible to the four neighboring cells), and min trails, which are formed by a non-diffusible pheromones. The robot follows the following rules according to the quantity of resources:

- If the quantity of resources is greater than a quantity max and there is no trail the robot drops diffusible pheromones (creation of max trails), else he climbs the existing trail;
- If the quantity is equal to a quantity min and there is a trail then the agent will climb it and erase the max trail to avoid the attraction of other robots, else if there is no trail, the robot will return and color min trail;
- If the quantity is less than a quantity min, and there is a trail then the robot climbs it else it creates a min trail.
- If the base is reached, the robot unloads resources and chooses between climb an existing trail and searching for a resource.

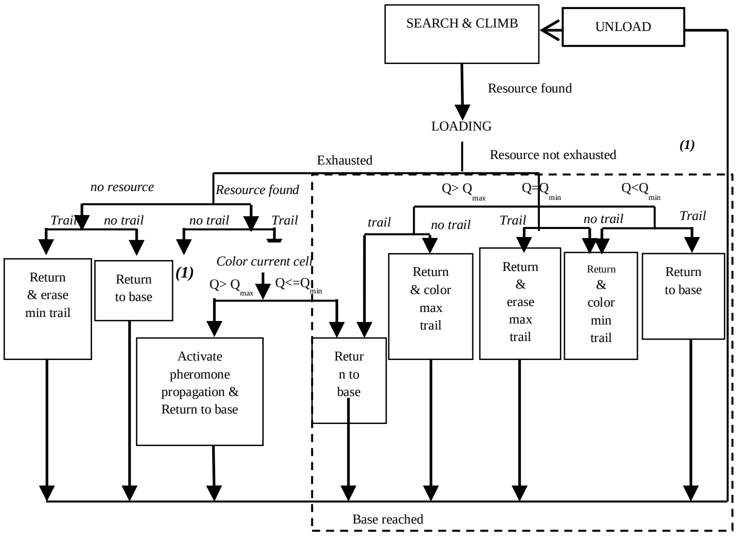


Figure 2. Cooperative c-marking agent behavior

IV. SIMULATION RESULTS AND COMPARAISON

According to the simulation results, we compared the proposed system with the original foraging system for verifying its effectiveness. The c-marking agent system offers a small level of cooperation between agents, whereas the ameliorated system offers a great level of cooperation, which decreases the foraging time.

We tested with square environments (grid with variable size); obstacles and resource locations are disseminated randomly in the environment, and all the agents start from the center (base). Several setups were used to test the model:

Setup 1 is defined as follows:

- An environment with 40 X 40 cells, 30% obstacles and 20 cells are resource locations; each resource contains 1000 units of resources.
- Each agent can load a maximum of 100 units.

As in [1], we define time as the number of iteration required to discover and exhaust all the resources in the environment. We evaluate the performance of the two models in different configurations (number of agents, size of the environment).

A. Influence of the number of agents on performance

Using the setup 1, and varying the number of agents from 5 to 160 agents, we obtained the results illustrated by Figure 3 and Table 1. Experiments show that increasing the number of agents decreases the time of foraging. This is due to the great level of cooperation.

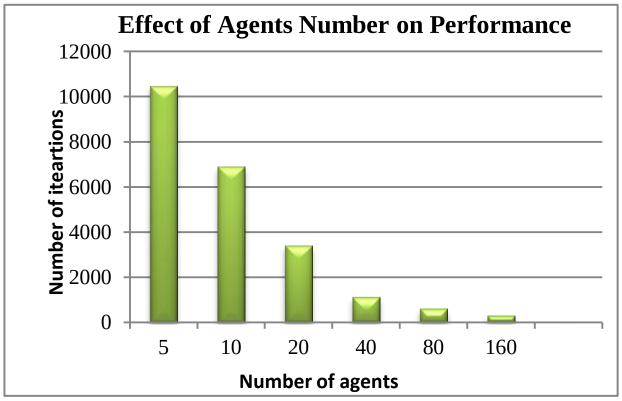


Figure3. Performance of Cooperative c-marking agents in setup 1

TABLE 1 INFLUENCE OF NUMBER OF AGENTS ON THE PERFORMANCE OF COOPERATIVE C-MARKING AGENTS

Number of agents	5	10	20	40	80
Iteration number	10476	6917	3403	1125	609

B. Influence of the environment size on performance

In this case, the number of agents is fixed and the environment size is variable. We used the setup 2 to see how the size of the environment affects the performance of the proposed model.

Setup 2 is defined by:

- Environments contain 5% obstacle density and 20 cells are resource locations; each resource contains 2000 units of resources.
- The number of agents is 50. Each agent can transport a maximum of 100 units.

Table 2 and Figure 4 show the performance of the algorithm for environments of varying sizes ranging from 12x12 to 100x100.

The results show that the foraging time decreases with increasing the size of the environment. The solution becomes ineffective due to the increase in exploration time. The problem of dead connected trails creates local minima that can lead to freeze a great number of agents in the local minima vicinity.

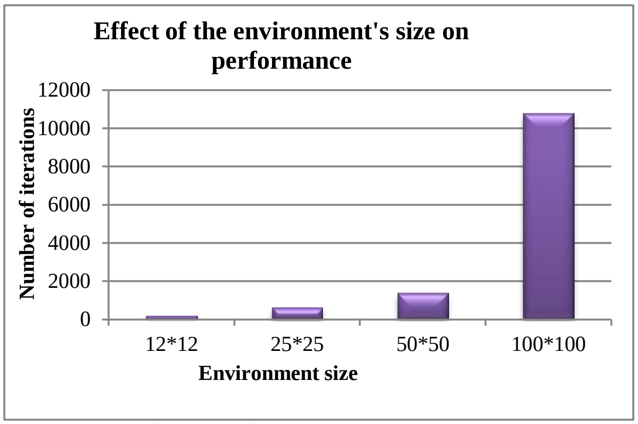


Figure 4. Performance of Cooperative c-marking agents in setup2

TABLE 2 INFLUENCE OF THE ENVIRONMENT SIZE ON THE PERFORMANCE OF COOPERATIVE C-MARKING AGENTS

Environment size	12*12	25*25	50*50	100*100
Iteration number	*192*	652	1395	10777

C. Influence of the obstacles on performance

Obstacles are disseminated in a random way in the environment. Such situation allows us to test the robustness of the algorithm to obstacles.

Setup 3 is defined by:

- Environment size is 41 X 41 cells, 20 cells are resource locations; each resource contains 1000 units of resources;
- Number of agents is 10 and each one can transport a maximum of 100 units;

The obstacle percentage is varying from 15 to 30 % of the environment surface. Results are shown in Table 3 and in Figure 5, which demonstrate that the performances do not depend on the density of obstacles. The algorithm offers an interesting level of robustness to obstacles.

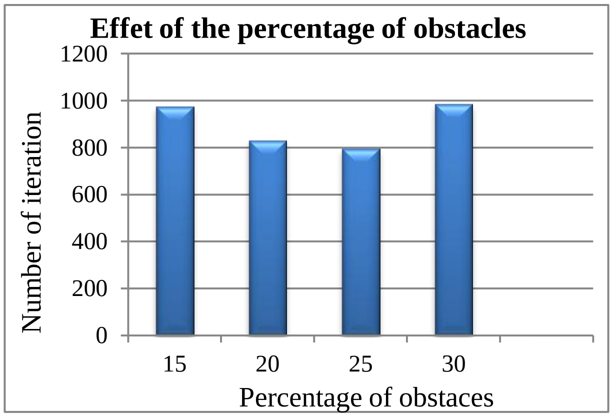


Figure 5. Performance of Cooperative c-marking agents in setup3

TABLE 3 INFLUENCE OF THE PERCENTAGE OF OBSTACLES ON THE PERFORMANCE OF COOPERATIVE C-MARKING AGENTS

Percentage of obstacles	15	20	25	30
Iteration number	976	831	796	985

D. *Comparaison with c-marking agents model*

Simulating the ant model requires more environment management mechanisms in which propagation of the pheromones represents a high computational cost. The main advantage of marking agents is their abilities of creating quick paths to the base during the exploration phase [1]. In the proposed model, the first problem is avoided by giving a new modeling for the digital pheromone, and the local minima problem is also avoided. Due to this last, agents are able to go back home easily.

Figure 6 presents a comparison between the two models. It shows that the proposed model gives more efficiency in time than the c-marking agents in case of varying the number of agents. This is due specifically to the great level of cooperation which decreases the exploration time. Figure 7 gives a comparison between the two models by varying the environment's size. Results show that this model gives a less efficiency in time than the c-marking agents' model. We think that this ineffectiveness of results is due to a problem that appears during the simulations, there is a possibility that two or more trail are connected (have common cells in some part of the trail); or crossed ones, when the agent in trail 1 exhaust the resource, it will erase the trail and this will cause the erase of the common portion of trail; the agents in the other trail have no way to continue to the base, or to the resource, because they look for colored cells , with this phenomena a great number of agents will be trapped and the simulation is continued with just those agents which are not trapped and this will increase the number of iteration .

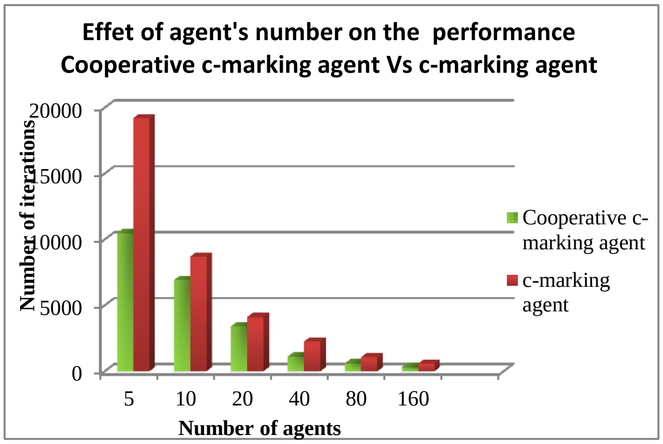


Figure 6. Effect of the number of agents
Cooperative c-marking agents Vs c-marking agents

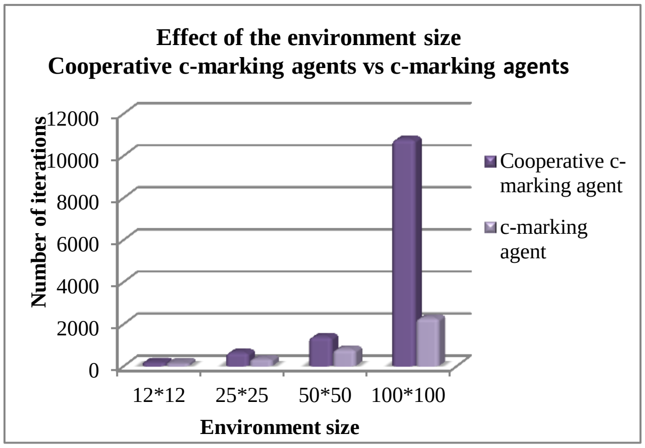


Figure 7. Effect of the environment size
Cooperative c-marking agents Vs c-marking agents

V. CONCLUSION AND FUTURE WORK

A multi agent model simulation and a new version of the c-marking agents' algorithm to increase the cooperation between agents and to decrease the time of foraging have been presented. Some other problems such as neighboring resources is solved with our new model and results indicates that the use of the new pheromone modeling give more efficiency in time than the original one. In perspective, we think that robot's behavior can be enhanced by introducing both new exploration approaches and solutions to problems such as trail erasing and APF fast convergence.

REFERENCES

[1] O. Simonin, F. Charpillet, and E. Thierry. Collective construction of numerical potential fields for the foraging problem. ACM TAAS, Transactions on Autonomous and Adaptive Systems. In press (25 pages), 2010.

[2] A. Winfield. Foraging robots. In Meyers, R., ed. Encyclopedia of Complexity and Systems Science. Springer, New York, 2009, pp. 3682-3700.

[3] E. Bonabeau, M. Dorigo, and G. Theraulaz. Swarm Intelligence-From Natural to Artificial Systems. Oxford University. Press, USA, 1999.

[4] M. Dorigo and M. Birattari. Swarm Intelligence. Scholarpedia, 2007.

[5] L. Panait and S. Luke. A pheromone-based utility model for collaborative foraging. In Proc AAMAS'04. ACM, 36-43 , 2004

[6] A. Bautin, O. simonin, and F. Charpillet. Stratégie d'exploration multi-robot fondée sur les champs de potentiels artificiels, Journée Francophones sur les Systèmes Multi-Agents (JFSMA), 2011.

[7] E. H. Østergaard, G. S. Sukhatme, and M. J. Mataric. Emergent bucket brigading: A simple mechanism for improving performance in multi robot constrained-space foraging tasks. In proc. Int. Conf. On Autonomous Agents, Montreal, Canada, May 2001.

[8] M. J. Mataric and M. J. Marjanovic. Synthesizing complex behaviors by composing simple primitives. In Proc. Self-Organization and Life, From Simple Rules to Global Complexity, European Conference on Artificial Life (ECAL-93), Brussels, pages 698-707, May 1993.

[9] A. J. Ijspeert, A. Martinoli, A. Billard, and L. M. Gambardella. Collaboration through the exploration of local interactions in autonomous collective robotics: The stick pulling experiment. Autonoumous robots, 2001.

[10] A. Martinoli, K. Easton, and W. Agassounon. Modeling swarm robotic systems: A case study in collaborative distributed manipulation. In Journal of Robotics research, Special Issue on Experimental Robotics, 2004.

[11] L. E. Paker. ALLIANCE: an architecture for fault tolerant, cooperative control of heterogeneous mobile robots. In Proc. IEEE/ RSJ International Conference on Intelligent Robots and Systems, 1994.

[12] R. Grob, E. Tuci, M. Dorigo, M. Bonani, and F. Mondada. Object transport by modular robots that self-assemble. In Proc. IEEE International Conference on Robotics and Automation, 2006.

[13] M. Dorigo, E. Tuci, R. Grob, V. Trianni, T. H. Labella, S. Nouyan, and C. Ampatzis. The SWARM-BOT project. In Erol Sahin and William Spears, editors, Swarm Robotics Workshop: State-of-the-art survey, number 3342 in Lecture Notes in Computer Science, Berlin Heidelberg, Springer Verlag, 2005.

[14] T. Balch and R. C. Arkin. Communication in reactive multiagent robotic systems. Autonomous Robots. 1994.

[15] D. Goldberg and M. J. Mataric. Interference as a tool for designing and evaluating multi robot controllers. In Proc. 14th National Conference on Artificial intelligence (AAAI-97). MIT Press, 1997.

[16] K. Lerman. Mathematical model of foraging in a group of robots inspired by ants foraging behavior. ACM Transactions on Autonomous and Adaptive Systems, 2006.

[17] E. Bonabeau, C. Theraulaz, and J. L. Deneubourg. Quantitative study of the fixed threshold model for the regulation of division of labor in insect societies. Proceedings of the RoyalSociety of London, Series B Biological Sciences, 1996.

[18] E. Bonabeau, G. Theraulaz, and J.L. Deneubourg. Fixed response thresholds and the regulation of division of labour in insect societies. Bulletin of Mathematical Biology, 1998.

[19] M. Krieger and J.B. Billeter. The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. Jour. of Robotics & Autonomous Systems, 2000.

[20] J. L. Deneubourg, S. Goss, J. M. Pasteels, D. Fresneau, and J. P. Lachaud. Self-organization mechanisms in ant societies (ii): learning in foraging and division of labour. Experientia Suppl, 1987.

[21] T. H. Labella, M. Dorigo, and J. L. Deneubourg. Division of labour in a group of robots inspired by ants foraging behaviour. ACM Transactions on Autonomous and Adaptive Systems, 2006.

[22] J. L. Deneubourg, S. Aron, S. Goss, J. M. Pasteels, and G. Duerrinck. Random behavior, amplification processes and number of participants: How they contribute to the foraging properties of ants. Physica, 1986

[23] B. Holldobler and E. Wilson. The Ants. Harvard University

[24] M. Resnick. Turtles, Termites, and Traffic Jams: Explorations in Massively parallel Microworlds. Cambridge, MIT Press, 1994.